mathematics

Special Issue Reprint

Statistical Data Modeling and Machine Learning with Applications, 3rd Edition

Edited by
Snezhana Gocheva-Ilieva

mdpi.com/journal/mathematics

MDPI

Statistical Data Modeling and Machine Learning with Applications, 3rd Edition

Statistical Data Modeling and Machine Learning with Applications, 3rd Edition

Guest Editor

Snezhana Gocheva-Ilieva

MDPI

Basel • Beijing • Wuhan • Barcelona • Belgrade • Novi Sad • Cluj • Manchester

Guest Editor
Snezhana Gocheva-Ilieva
Department of Mathematical
Analysis, Faculty of
Mathematics and Informatics
University of Plovdiv
Paisii Hilendarski
Plovdiv
Bulgaria

Editorial Office
MDPI AG
Grosspeteranlage 5
4052 Basel, Switzerland

This is a reprint of the Special Issue, published open access by the journal *Mathematics* (ISSN 2227-7390), freely accessible at: https://www.mdpi.com/journal/mathematics/special_issues/Stat_Data_Mach_Learn.

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

Lastname, A.A.; Lastname, B.B. Article Title. *Journal Name* **Year**, *Volume Number*, Page Range.

ISBN 978-3-7258-9395-9 (Hbk)
ISBN 978-3-7258-9396-6 (PDF)
https://doi.org/10.3390/books978-3-7258-9396-6

© 2026 by the authors. Articles in this reprint are Open Access and distributed under the Creative Commons Attribution (CC BY) license. The reprint as a whole is distributed by MDPI under the terms and conditions of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) license (https://creativecommons.org/licenses/by-nc-nd/4.0/).

Contents

About the Editor . . . . . **ix**

Snezhana Gocheva-Ilieva
Preface to the Special Issue "Statistical Data Modeling and Machine Learning with Applications, 3rd Edition"
Reprinted from: *Mathematics* **2026**, *14*, 3172, https://doi.org/10.3390/math14173172 . . . . . **1**

Georgios Athanasiou, Josep Lluis Arcos and Jesus Cerquides
Enhancing Medical Image Segmentation: Ground Truth Optimization through Evaluating Uncertainty in Expert Annotations
Reprinted from: *Mathematics* **2023**, *11*, 3771, https://doi.org/10.3390/math11173771 . . . . . **8**

Michaël Hillen, Ivan De Boi, Thomas De Kerf, Seppe Sels, Edgar Cardenas De La Hoz, Jona Gladines, et al.
Enhanced Checkerboard Detection Using Gaussian Processes
Reprinted from: *Mathematics* **2023**, *11*, 4568, https://doi.org/10.3390/math11224568 . . . . . **23**

Todorka Glushkova, Vanya Ivanova and Boyan Zlatanov
Beyond Traditional Assessment: A Fuzzy Logic-Infused Hybrid Approach to Equitable Proficiency Evaluation via Online Practice Tests
Reprinted from: *Mathematics* **2024**, *12*, 371, https://doi.org/10.3390/math12030371 . . . . . **36**

Boris V. Malozyomov, Nikita V. Martyushev, Svetlana N. Sorokova, Egor A. Efremenkov, Denis V. Valuev and Mengxu Qi
Analysis of a Predictive Mathematical Model of Weather Changes Based on Neural Networks
Reprinted from: *Mathematics* **2024**, *12*, 480, https://doi.org/10.3390/math12030480 . . . . . **50**

Oksana Mandrikova and Bogdana Mandrikova
Hybrid Model of Natural Time Series with Neural Network Component and Adaptive Nonlinear Scheme: Application for Anomaly Detection
Reprinted from: *Mathematics* **2024**, *12*, 1079, https://doi.org/10.3390/math12071079 . . . . . **67**

Nhat-Duc Hoang, Van-Duc Tran and Xuan-Linh Tran
Predicting Compressive Strength of High-Performance Concrete Using Hybridization of Nature-Inspired Metaheuristic and Gradient Boosting Machine
Reprinted from: *Mathematics* **2024**, *12*, 1267, https://doi.org/10.3390/math12081267 . . . . . **82**

Luca Di Persio, Nicola Fraccarolo, Andrea Veronese
Wind Energy Production in Italy: A Forecasting Approach Based on Fractional Brownian Motion and Generative Adversarial Networks
Reprinted from: *Mathematics* **2024**, *12*, 2105, https://doi.org/10.3390/math12132105 . . . . . **106**

Pengjiao Wang, Qiuliang Long, Hu Zhang, Xu Chen, Ran Yu and Fengqi Guo
Forecasting and Multilevel Early Warning of Wind Speed Using an Adaptive Kernel Estimator and Optimized Gated Recurrent Units
Reprinted from: *Mathematics* **2024**, *12*, 2581, https://doi.org/10.3390/math12162581 . . . . . **122**

Rangan Gupta and Christian Pierdzioch
Multi-Task Forecasting of the Realized Volatilities of Agricultural Commodity Prices
Reprinted from: *Mathematics* **2024**, *12*, 2952, https://doi.org/10.3390/math12182952 . . . . . **140**

Shengkun Xie
Analyzing the Influence of Telematics-Based Pricing Strategies on Traditional Rating Factors in Auto Insurance Rate Regulation
Reprinted from: *Mathematics* **2024**, *12*, 3150, https://doi.org/10.3390/math12193150 . . . . . **166**

Vladica S. Stojanović, Tanja Jovanović Spasojević, Radica Bojičić, Brankica Pažun and Zlatko Langović
Cauchy–Logistic Unit Distribution: Properties and Application in Modeling Data Extremes
Reprinted from: *Mathematics* **2025**, *13*, 255, https://doi.org/10.3390/math13020255 . . . . . **189**

Mohammed Alruqimi and Luca Di Persio
Probabilistic Forecasting of Crude Oil Prices Using Conditional Generative Adversarial Network Model with Lévy Process
Reprinted from: *Mathematics* **2025**, *13*, 307, https://doi.org/10.3390/math13020307 . . . . . **213**

Wenyi Wang, Haifei Liu, Lin Jiang and Lei Wang
Prediction of Lithium Carbonate Prices in China Applying a VMD–SSA–LSTM Combined Model
Reprinted from: *Mathematics* **2025**, *13*, 613, https://doi.org/10.3390/math13040613 . . . . . **229**

Ricardo Caetano and José Manuel Oliveira and Patrícia Ramos
Transformer-Based Models for Probabilistic Time Series Forecasting with Explanatory Variables
Reprinted from: *Mathematics* **2025**, *13*, 814, https://doi.org/10.3390/math13050814 . . . . . **246**

Snežana Tadić, Aida Kalem, Mladen Krstić, Nermin Čabrić, Adisa Medić and Miloš Veljović
Benchmarking Analysis of Railway Infrastructure Managers: A Hybrid Principal Component Analysis (PCA), Grey Best–Worst Method (G-BWM), and Assurance Region Data Envelopment Analysis (AR-DEA) Model
Reprinted from: *Mathematics* **2025**, *13*, 830, https://doi.org/10.3390/math13050830 . . . . . **275**

Kostadin Yotov, Emil Hadzhikolev and Stanka Hadzhikoleva
Modeling the Relationship Between Radon Anomalies and Seismic Activity Using Artificial Neural Networks and Statistical Methods
Reprinted from: *Mathematics* **2025**, *13*, 1075, https://doi.org/10.3390/math13071075 . . . . . **303**

Somlak Utudee, Pharunyou Chanthorn, and Sompop Moonchai
Dual Kriging with a Nonlinear Hybrid Gaussian RBF–Polynomial Trend: The Theory and Application to $PM_{2.5}$ Estimation in Northern Thailand
Reprinted from: *Mathematics* **2025**, *13*, 2811, https://doi.org/10.3390/math13172811 . . . . . **330**

Vasileios Alevizakos
A Nonparametric Double Homogeneously Weighted Moving Average Signed-Rank Control Chart for Monitoring Location Parameter
Reprinted from: *Mathematics* **2025**, *13*, 3027, https://doi.org/10.3390/math13183027 . . . . . **354**

Ekaterina V. Orlova
Dynamic Modeling and Structural Equation Analysis of Team Innovativeness Under the Influence of Social Capital and Conflict Mediation
Reprinted from: *Mathematics* **2025**, *13*, 3301, https://doi.org/10.3390/math13203301 . . . . . **389**

Paulino José García-Nieto, Esperanza García-Gonzalo, Luis Alfonso Menéndez-García, Laura Álvarez-de-Prado, Marta Menéndez-Fernández and Antonio Bernardo-Sánchez
Interpretable Machine Learning Models for Estimating Electric Energy Consumption in Steel Industries
Reprinted from: *Mathematics* **2025**, *13*, 3364, https://doi.org/10.3390/math13213364 . . . . . **410**

Michael Politis, Nicholas Christakis, Zoi Dorothea Pana, Dimitris Drikakis
Integrating Machine Learning and Scenario Modelling for Robust Population Forecasting Under Crisis and Data Scarcity
Reprinted from: *Mathematics* **2025**, *13*, 4024, https://doi.org/10.3390/math13244024 . . . . . **432**

Shengkun Xie and Chinwendu Onungwe
Modeling and Forecasting U.S. Outbound Travel Demand Across Regions Using Time Series Model and Machine Learning: A Comparative Study
Reprinted from: *Mathematics* **2026**, *14*, 758, https://doi.org/10.3390/math14050758 . . . . . **462**

Daniel Álvarez-Ruiz, Alberto Sánchez-Alzola and Andrés Pastor-Fernández
Geospatial Clustering of GNSS Stations Using Unsupervised Learning: A Statistical Framework to Enhance Deformation Analysis for Environmental Risk Management
Reprinted from: *Mathematics* **2026**, *14*, 855, https://doi.org/10.3390/math14050855 . . . . . **482**

About the Editor

Snezhana Gocheva-Ilieva

Snezhana Gocheva-Ilieva, D.Sc., Ph.D., is currently a Guest Professor in the Department of Mathematical Analysis at the University of Plovdiv Paisii Hilendarski, Bulgaria. She received her M.Sc. and B.Sc. degrees in Computational Mathematics from Sofia University "St. Kliment Ohridski" and her Ph.D. in Physics and Mathematics from Taras Shevchenko National University of Kyiv. She received her Doctor of Sciences degree in Mathematics in 2016. From 1985 to 2011, Prof. Gocheva-Ilieva was an Associate Professor at the University of Plovdiv Paisii Hilendarski and has been a Full Professor here since 2011. She is a member of the Union of Bulgarian Mathematicians and the American Mathematical Society and is also a reviewer for many international journals in mathematics, statistics, environmental science, engineering sciences, and related fields. Her primary research interests include modeling and simulations of plasma and laser physics and engineering; modeling in environmental science; theories and applications of finite difference methods for ordinary and partial differential equations; singular problems; applied computational statistics; applications of predictive data mining techniques; and machine learning. Her secondary interests fall into scientific computing, ICT, programming, and innovative technologies in mathematics education.

mathematics

MDPI

Editorial

Preface to the Special Issue "Statistical Data Modeling and Machine Learning with Applications, 3rd Edition"

Snezhana Gocheva-Ilieva

Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 24 Tzar Asen St, 4000 Plovdiv, Bulgaria; snow@uni-plovdiv.bg

MSC: 60E05; 60G15; 60A86; 62M45; 62G08; 62H12; 68T05; 68T07

1. Introduction

Interest in statistical modeling of complex systems and large-scale phenomena using data about them continues to actively excite the scientific community. On the one hand, the quantity and quality of data are increasing, and on the other hand, the methods, algorithms, and software tools for processing them are constantly being enhanced and refined. A key element in this process is strengthening the influence of mathematics and statistics in modern research. This contributes significantly to greater reliability and accuracy in established trends and scientific facts. One of the most difficult challenges in this area is achieving the most accurate and objectively correct forecasts possible for static datasets, time series, and spatiotemporal data.

This Special Issue (SI) is the third consecutive issue of the SI "Statistical Data Modeling and Machine Learning with Applications" in the journal *Mathematics*, for which I had the pleasure of serving as a guest editor [1]. It is classified under section "D1: Probability and Statistics". The research objectives spanned a wide range of areas, including probability and statistical theory and applications, statistical modeling, time-series analysis and forecasting, machine learning (ML) algorithms, hybrid modeling, and applications.

This SI includes 23 scientific articles by 81 authors from 16 countries: Belgium, Bosnia and Herzegovina, Bulgaria, Canada, China, Cyprus, Germany, Greece, Italy, Portugal, Russia, Serbia, South Africa, Spain, Thailand, and Vietnam. As of August 2026, these contributions have received more than 85,000 views and approximately 215 independent citations.

In the field of probability and statistics, new results have been obtained on various aspects of stochastic processes and stochastic analysis, Gaussian processes, Gaussian process regression (kriging), stochastic approximation, probabilistic models, probability distributions, and other statistical topics.

The preferred approach is to build hybrid models that address diverse problems by applying statistical and advanced data analysis methods across environmental science, economics, engineering, industry, the social sciences, and other fields. These models combine classical statistical methods with advances in artificial intelligence (AI) and ML. They propose innovative methods and algorithms based on neural networks and related approaches to inference from stochastic processes, fuzzy logic, probabilistic time series forecasting, signal processing, and more.

2. Overview of the Contributions

This section provides a brief overview of all published articles. The articles are reviewed in the order of their publication.

In the first article of the SI, Athanasiou et al. use coupled convolutional neural network (CNN) algorithms in combination with statistical methods for medical image segmentation (Contribution 1). For the first time, a model is proposed to compute and optimize ground-truth reliability from confusion matrices derived from uncertain areas, using maximum-likelihood estimation. This approach is applied to a small real-world dataset with high agreement among annotating experts.

In the field of computer vision, various probabilistic models are being developed to improve low-quality images affected by unclear angles, low resolution, and other factors, such as camera imperfections or defects. In this field, Hillen et al. proposed a new algorithm and a checkerboard detection pipeline that can be applied after standard checkerboard detection (Contribution 2). For this purpose, a machine learning technique based on Gaussian processes is used to transform the checkerboard's local coordinates into pixel coordinates. As a result, mapping and prediction of image coordinates for undetected or hidden angles, smoothing, and overall image enhancement are facilitated.

A new methodology and model for assessing students' foreign language knowledge in a cyber–physical educational environment, based on fuzzy theory, are presented in (Contribution 3). The modeling of processes and communication among the environment's intelligent components is performed using the formal semantics of the calculus of context-aware ambients (CCAs).

An application for weather time-series analysis and forecasting using a neural network (NN) ensemble is presented by Malozyomov et al. in (Contribution 4). This technique is implemented for classification by combining the results of several independent NN models, each trained and evaluated on the same historical observational dataset. The following NN methods are used: differential-polynomial NN (D-PNN), radial basis function network (RBFN), and multilayer perceptron network (MLPN). To build essentially diverse individual models in the final ensemble, constructive algorithms are applied, using different NN topologies, hyperparameters, and various mathematical methods for minimizing the root mean square error. The resulting models achieve prediction accuracy of up to 91% compared to the measured data.

In the next contribution, Mandrikova et al. address the problem of modeling non-stationary time series using data from neutron- monitor ground stations that record cosmic-ray intensities in near-Earth space (Contribution 5). The data exhibit a complex structure, including anomalies, correlated noise, and other features. The authors constructed a hybrid model that combines an autoencoder NN with a nonlinear adaptive approximation scheme and a wavelet basis selection criterion. The key results focus on modeling time episodes with anomalies. To this end, a likelihood ratio test was used to optimize estimation of the anomalous component in the time series.

In the following paper (Contribution 6), the task of predicting the compressive strength (CS) of high-performance concrete (HPC) mixes is addressed using several independent determinants, including cement content, fly ash content, silica fume content, fine-aggregate content, coarse-aggregate content, water content, age, and others. For this purpose, a high-performance ML regression model is built and evaluated, based on a combination of two methods—the ensemble method, Extreme Gradient Boosting (XGBoost), and the nature-inspired metaheuristic algorithm Differential Flower Pollination (DFP). The resulting hybrid XGBoost-DFP model demonstrates excellent predictive performance for the CS of HPC, compared with ANN and Support Vector Machine methods.

https://doi.org/10.3390/math14173172

Di Persio et al. (Contribution 7) developed a machine learning model for forecasting time series of wind energy production. A stochastic approach combining stochastic differential equations with dynamic and generative adversarial networks was applied in two scenarios—one with Brownian motion and one with fractional Brownian motion using an experimentally estimated Hurst exponent. The model provides a one-week forward prediction horizon.

In (Contribution 8), Wang et al. model the wind speed time series as a function of air temperature, humidity, and the first-order finite difference of wind speed. An advanced hybrid approach that uses a probabilistic method for multilevel prediction is proposed. Two main methods are combined—the optimized gated recurrent unit (GRU), a type of recurrent neural network (RNN) designed to predict dynamic data, and the adaptive kernel density estimator (AKDE), which determines the bandwidth locally based on the probability density of data points. The constructed model allows calculation of the conditional probability of the actual wind speed and, in particular, estimation of the exceedance of critical values, resulting in more accurate probabilistic forecasts that support early warning systems.

Gupta and Pierdzioch study the application of multi-task forecasting stacking algorithms to realized volatility in agricultural commodity prices, using empirical analyses of daily time-series data (Contribution 9). Various stacking models are built and analyzed using the heterogeneous autoregressive (HAR) method as the base learner and different meta-learners such as Lasso, elastic net, and ridge regression. In addition, modified stacking algorithms with alternative combinations of estimators are used, including a Lasso–Lasso estimator, a Lasso–Random Forests, a ridge–ridge estimator, and others. The results demonstrate high model accuracy and effective accounting for various effects in the data, particularly spillover effects.

In article (Contribution 10), Xie presents a novel pricing strategy for insurance markets that incorporates usage-based insurance. This study analyzes how telematics variables, such as annual percentage driven, total miles driven, and driving patterns, influence the distributional behavior of conventional rating factors used to predict auto insurance risk in rate regulation. To this end, non-negative sparse principal component analysis (NSPCA) is used to reduce data dimensionality and to construct predictive pricing models using generalized linear models (GLMs).

In general statistics, a novel two-parameter stochastic distribution for a single random variable with markedly elevated values at the ends of the unit interval was studied in (Contribution 11). The distribution is called the Cauchy–logistic unit distribution. It is obtained by transforming the Cauchy distribution via a generalized logistic mapping onto the unit interval. A variety of properties of this distribution are established, and its application to modeling data extremes is presented.

The aim of the paper (Contribution 12) is to forecast the complex dynamics and high volatility of crude oil prices, which are influenced by external factors. As an alternative to classical time-series methods such as ARIMA or long short-term memory (LSTM) networks, the authors built a hybrid model that integrates a conditional generative adversarial network with a Lévy–Merton jump-diffusion process. The models demonstrate excellent forecasting performance on historical data for real-world oil prices.

To predict lithium carbonate prices in China, Wang et al. propose an efficient approach that combines variational mode decomposition, sparrow search, and LSTM algorithms (Contribution 13). The model incorporates prices of related materials used in the lithium battery industry, as well as macro-environmental indicators.

https://doi.org/10.3390/math14173172

Transformer-based deep learning and probabilistic forecasting for retail time series are presented in (Contribution 14). In particular, the Vanilla Transformer, Informer, Autoformer, ETSformer, NSTransformer, and Reformer architectures are used. Adequate models are constructed using multiple explanatory variables, including selling prices, socioeconomic factors, and other variables.

Tadic et al. developed a comprehensive approach to analyzing and optimizing railway infrastructure management in (Contribution 15). The proposed hybrid model integrates principal component analysis (PCA) to reduce data dimensionality and group variables, the Gray Best–Worst method to extract essential information from expert evaluations, and Assurance Region Data Envelopment Analysis (AR-DEA) to assess the relative efficiency of railway infrastructure managers under realistic conditions.

To determine the influence of meteorological variables on outdoor radon concentrations, and in particular to identify potential radon anomalies associated with earthquakes, Yotov et al. built various artificial neural network models (Contribution 16). An essential part of the study was predicting radon concentrations in time-series segments with no recorded earthquakes and detecting anomalies in segments with recorded earthquakes. The models were trained on data from three observation stations in Bulgaria.

Several methods and techniques are combined in (Contribution 17) to achieve more accurate $PM_{2.5}$ pollution prediction. Dual kriging (DK) models are built for spatial interpolation of data using Gaussian radial basis functions, k-means clustering, genetic algorithms, and other methods. The non-singularity of the coefficient matrix in the dual kriging system is theoretically proven. Results from $PM_{2.5}$ models for Northern Thailand are presented.

A new design for a nonparametric signed-rank control chart (CC) was proposed by Alevizakos, assuming that the distribution is unknown but continuous and symmetric (Contribution 18). The CC detects shifts in the location parameter of the distribution. Simulations were conducted using the Monte Carlo method. Comparisons with several well-known distributions are presented to demonstrate the effectiveness of the proposed CC.

Orlova (Contribution 19) uses multivariate statistical analysis, confirmatory factor analysis, and structural equation modeling to model complex relationships in the social sphere. Project teamwork is examined along three main parameters—innovativeness, social capital, and conflicts—for two groups of participants—innovators and adaptors. Various factors are considered to determine the level of equilibrium.

Modeling steelworks' energy consumption is the main topic of the paper (Contribution 20). For this purpose, the Multivariate Adaptive Regression Splines (MARS) method and the following regularization regression methods, including ridge regression, elastic net, and the Least Absolute Shrinkage and Selection Operator (Lasso), were used. The hyperparameters of the models were tuned using the Whale Optimization algorithm (WOA).

A specific study on population dynamics forecasting was carried out in (Contribution 21). To reduce the influence of large extreme values (point outliers), the authors used Huber-M regression, and to account for the greater influence of more recent data compared to older ones, they used a variant of the Exponential Moving Average method. By taking into account different demographic factors across various scenarios using the standard cohort-component methodology, 15-year forecasts were obtained from data from three European countries.

The application of statistics, ML, and deep learning methods to time series in the social sciences is presented in (Contribution 22). Six models for tourism data were built and compared. For this purpose, the Autoregressive Integrated Moving Average (ARIMA)

https://doi.org/10.3390/math14173172

method, the ML ensemble methods XGBoost and Random Forests, and two deep learning methods—LSTM and the Gated Recurrent Unit (GRU)—were used.

The SI concludes with article (Contribution 23), in which the authors examine the characteristics of Global Navigation Satellite System (GNSS) time series at a large number of stations. The modeling was carried out using unsupervised learning tools, including linear harmonic regression, analysis of noise levels, calculation of basic data characteristics such as periodicity and velocities, clustering of stations, and application of linear discriminant analysis.

3. Conclusions

The general overview of the results shows a wide thematic diversity and a range of approaches, methods, algorithms, frameworks, and applications. It should be noted that all contributions share a common foundation—the theory of probability and mathematical statistics. To varying degrees, these foundations are involved in the creation, evaluation, and validation of the built models. Depending on the methods used, the models and their results are classified into three main areas, namely: probability theory and stochastic processes, statistics, and computer science. They fully align with the proposed SI topics (see [1]). One can note the authors' most preferred topics, including computational statistics, dimensionality reduction, parameter estimation, nonparametric statistical modeling, supervised learning (classification, regression), clustering, financial statistics, time series analysis and forecasting, ensemble methods, neural networks, deep learning, hybrid models, and data analysis.

On this basis, we can conclude that the goal of this third edition of SI "Statistical Data Modeling and Machine Learning with Applications" has been successfully achieved. The results presented may be useful to scientists working in the suggested fields, doctoral candidates, students, and anyone interested in studying the latest mathematical approaches to modeling complex real-world phenomena.

Finally, I would like to express my sincere thanks to all the authors who contributed to the realization of this SI for their hard work on their research and the presentation of their articles. I also express my deep respect and gratitude to all the reviewers, whose help and patience substantially improved the scientific quality of the results. In addition, I owe thanks to all the editors at the journal *Mathematics* for their highly professional work and assistance throughout the compilation of this SI.

Funding: This study was supported by the National Center for High-performance and Distributed Computing (NCHDC), part of the National Roadmap of RIs, funded by MES-Bulgaria.

Conflicts of Interest: The author declares no conflicts of interest. The funders had no role in the design of the study, the writing of the manuscript, or the decision to publish the results.

List of Contributions

1. Athanasiou, G.; Arcos, J.L.; Cerquides, J. Enhancing medical image segmentation: Ground truth optimization through evaluating uncertainty in expert annotations. *Mathematics* **2023**, *11*, 3771. https://doi.org/10.3390/math11173771.
2. Hillen, M.; De Boi, I.; De Kerf, T.; Sels, S.; Cardenas De La Hoz, E.; Gladines, J.; Steenackers, G.; Penne, R.; Vanlanduit, S. Enhanced checkerboard detection using Gaussian processes. *Mathematics* **2023**, *11*, 4568. https://doi.org/10.3390/math11224568.
3. Glushkova, T.; Ivanova, V.; Zlatanov, B. Beyond traditional assessment: A fuzzy logic-infused hybrid approach to equitable proficiency evaluation via online practice tests. *Mathematics* **2024**, *12*, 371. https://doi.org/10.3390/math12030371.

4. Malozyomov, B.V.; Martyushev, N.V.; Sorokova, S.N.; Efremenkov, E.A.; Valuev, D.V.; Qi, M. Analysis of a predictive mathematical model of weather changes based on neural networks. *Mathematics* **2024**, *12*, 480. https://doi.org/10.3390/math12030480.
5. Mandrikova, O.; Mandrikova, B. Hybrid model of natural time series with Neural Network component and adaptive nonlinear scheme: Application for anomaly detection. *Mathematics* **2024**, *12*, 1079. https://doi.org/10.3390/math12071079.
6. Hoang, N.-D.; Tran, V.-D.; Tran, X.-L. Predicting compressive strength of high-performance concrete using hybridization of nature-inspired metaheuristic and gradient boosting machine. *Mathematics* **2024**, *12*, 1267. https://doi.org/10.3390/math12081267.
7. Di Persio, L.; Fraccarolo, N.; Veronese, A. Wind energy production in Italy: A forecasting approach based on fractional brownian motion and generative adversarial networks. *Mathematics* **2024**, *12*, 2105. https://doi.org/10.3390/math12132105.
8. Wang, P.; Long, Q.; Zhang, H.; Chen, X.; Yu, R.; Guo, F. Forecasting and multilevel early warning of wind speed using an adaptive kernel estimator and optimized gated recurrent units. *Mathematics* **2024**, *12*, 2581. https://doi.org/10.3390/math12162581.
9. Gupta, R.; Pierdzioch, C. Multi-task forecasting of the realized volatilities of agricultural commodity prices. *Mathematics* **2024**, *12*, 2952. https://doi.org/10.3390/math12182952.
10. Xie, S. Analyzing the influence of telematics-based pricing strategies on traditional rating factors in auto insurance rate regulation. *Mathematics* **2024**, *12*, 3150. https://doi.org/10.3390/math12193150.
11. Stojanović, V.S.; Spasojević, T.; Bojičić, R.; Pažun, B.; Langović, Z. Cauchy–logistic unit distribution: Properties and application in modeling data extremes. *Mathematics* **2025**, *13*, 255. https://doi.org/10.3390/math13020255.
12. Alruqimi, M.; Di Persio, L. Probabilistic forecasting of crude oil prices using conditional generative adversarial network model with Lévy process. *Mathematics* **2025**, *13*, 307. https://doi.org/10.3390/math13020307.
13. Wang, W.; Liu, H.; Jiang, L.; Wang, L. Prediction of lithium carbonate prices in China applying a VMD–SSA–LSTM combined model. *Mathematics* **2025**, *13*, 613. https://doi.org/10.3390/math13040613.
14. Caetano, R.; Oliveira, J.M.; Ramos, P. Transformer-based models for probabilistic time series forecasting with explanatory variables. *Mathematics* **2025**, *13*, 814. https://doi.org/10.3390/math13050814.
15. Tadić, S.; Kalem, A.; Krstić, M.; Čabrić, N.; Medić, A.; Veljović, M. Benchmarking analysis of railway infrastructure managers: A hybrid principal component analysis (PCA), grey best–worst method (G-BWM), and assurance region data envelopment analysis (AR-DEA) model. *Mathematics* **2025**, *13*, 830. https://doi.org/10.3390/math13050830.
16. Yotov, K.; Hadzhikolev, E.; Hadzhikoleva, S. Modeling the relationship between radon anomalies and seismic activity using artificial neural networks and statistical methods. *Mathematics* **2025**, *13*, 1075. https://doi.org/10.3390/math13071075.
17. Utudee, S.; Chanthorn, P.; Moonchai, S. Dual kriging with a nonlinear hybrid gaussian RBF–polynomial trend: The theory and application to PM2.5 estimation in Northern Thailand. *Mathematics* **2025**, *13*, 2811. https://doi.org/10.3390/math13172811.
18. Alevizakos, V. A nonparametric double homogeneously weighted moving average signed-rank control chart for monitoring location parameter. *Mathematics* **2025**, *13*, 3027. https://doi.org/10.3390/math13183027.
19. Orlova, E.V. Dynamic modeling and structural equation analysis of team innovativeness under the influence of social capital and conflict mediation. *Mathematics* **2025**, *13*, 3301. https://doi.org/10.3390/math13203301.
20. García-Nieto, P.J.; García-Gonzalo, E.; Menéndez-García, L.A.; Álvarez-de-Prado, L.; Menéndez-Fernández, M.; Bernardo-Sánchez, A. Interpretable machine learning models for estimating electric energy consumption in steel industries. *Mathematics* **2025**, *13*, 3364. https://doi.org/10.3390/math13213364.
21. Politis, M.; Christakis, N.; Pana, Z.D.; Drikakis, D. Integrating machine learning and scenario modelling for robust population forecasting under crisis and data scarcity. *Mathematics* **2025**, *13*, 4024. https://doi.org/10.3390/math13244024.

22. Xie, S.; Onungwe, C. Modeling and forecasting U.S. outbound travel demand across regions using time series model and machine learning: A comparative study. *Mathematics* **2026**, *14*, 758. https://doi.org/10.3390/math14050758.
23. Álvarez-Ruiz, D.; Sánchez-Alzola, A.; Pastor-Fernández, A. Geospatial clustering of GNSS stations using unsupervised learning: A statistical framework to enhance deformation analysis for environmental risk management. *Mathematics* **2026**, *14*, 855. https://doi.org/10.3390/math14050855.

Reference

1. Special Issue Statistical Data Modeling and Machine Learning with Applications, 3rd Edition. Available online: https://www.mdpi.com/si/mathematics/Stat_Data_Mach_Learn (accessed on 23 August 2026).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Enhancing Medical Image Segmentation: Ground Truth Optimization through Evaluating Uncertainty in Expert Annotations

Georgios Athanasiou *, Josep Lluis Arcos * and Jesus Cerquides *

Artificial Intelligence Research Institute (IIIA), Spanish National Research Council (CSIC), Campus Autonomous University of Barcelona (UAB), 08193 Barcelona, Spain

* Correspondence: gathanasiou@iiia.csic.es (G.A.); arcos@iiia.csic.es (J.L.A.); cerquide@iiia.csic.es (J.C.)

Abstract: The surge of supervised learning methods for segmentation lately has underscored the critical role of label quality in predicting performance. This issue is prevalent in the domain of medical imaging, where high annotation costs and inter-observer variability pose significant challenges. Acquiring labels commonly involves multiple experts providing their interpretations of the "true" segmentation labels, each influenced by their individual biases. The blind acceptance of these noisy labels as the ground truth restricts the potential effectiveness of segmentation algorithms. Here, we apply coupled convolutional neural network approaches to a small-sized real-world dataset of bovine cumulus oocyte complexes. This is the first time these methods have been applied to a real-world annotation medical dataset, since they were previously tested only on artificially generated labels of medical and non-medical datasets. This dataset is crucial for healthy embryo development. Its application revealed an important challenge: the inability to effectively learn distinct confusion matrices for each expert due to large areas of agreement. In response, we propose a novel method that focuses on areas of high uncertainty. This approach allows us to understand the individual characteristics better, extract their behavior, and use this insight to create a more sophisticated ground truth using maximum likelihood. These findings contribute to the ongoing discussion of leveraging machine learning algorithms for medical image segmentation, particularly in scenarios involving multiple human annotators.

Keywords: deep learning; convolutional neural networks; medical data; assisted reproductive technology; image segmentation; optimization; consensus segmentation

MSC: 68T45

1. Introduction

The task of segmenting anatomical structures in medical images is known to be challenged by significant inter-reader variability, which can impact the performance of subsequent supervised machine learning models. This issue is particularly pressing in the medical and biological domains, where labeled data are often limited due to the high costs associated with expert annotations. Across diverse fields and applications, differing biases and levels of expertise among annotators lead to considerable variations in segmentation annotations of structures in medical images [1]. For instance, in the realm of assisted reproduction, expert annotators are primarily responsible for the segmentation of relevant structures, both in human and other mammalian data. For this kind of tasks, the UNet architecture [2] has been used extensively in several medical and biological fields [3]. Despite their expertise, disagreements and inconsistencies in annotations are not uncommon, adding another layer of complexity to the already challenging task. As a result, despite the proliferation of digital medical imaging data over the past couple of decades, access to data with curated labels that can be readily used for machine learning remains

relatively scarce [4]. This underlines the pressing need for advanced methodologies capable of robust learning from such noisy annotations.

Various pre-processing techniques are routinely employed to reconcile inter-reader variations and curate segmentation annotations by merging labels from multiple experts. A particularly common and straightforward approach utilizes the majority-vote approach [5], whereby the ground truth is derived from the most frequently occurring opinion among the experts. This method has been effectively applied in various contexts, including Assisted Reproductive Technology (ART) for COC, as demonstrated by Athanasiou et al. [6]. Furthermore, an advanced variant of this method, which considers class similarities, has demonstrated effectiveness in the aggregation of brain tumor segmentation labels [1]. However, a primary limitation of these approaches is the assumption that all experts hold equal reliability. To circumvent this, Warfield et al. [7] proposed a label fusion technique, known as STAPLE, which explicitly models the individual reliability of each expert and leverages this information to 'weight' their opinions during the label aggregation process. Due to its consistent superiority over traditional majority-vote pre-processing across various applications, STAPLE has emerged as the preferred label fusion method.

Recently, STAPLE has seen numerous enhancements, particularly in the context of multi-atlas segmentation problems [8]. These enhancements particularly shine when image registration transfers segments from labeled images ("atlases") to a new scan. A notable improvement, known as STEPS, was introduced by Cardoso et al. [9]. This approach took things a step further by incorporating the local morphological similarity between atlases and target images. Numerous extensions of this approach, such as in the work of Asman et al. [10], have since been explored. However, a shared limitation across these label fusion methods is the absence of a mechanism allowing information integration across different training images. This restriction essentially limits the applicability of these methods to cases where each image comes with a substantial number of annotations from multiple experts. In practice, obtaining such many annotations can be prohibitively expensive. Moreover, these methods often utilize relatively simplistic functions to model the relationship between observed noisy annotations, true labels, and expert reliability. As a result, they may fall short of capturing the complex characteristics of human annotators.

Several studies suggest solutions to these challenges by devising models that execute an end-to-end supervised segmentation process. This process aims to simultaneously determine the reliability of multiple human annotators and the true segmentation labels using only noisy labels. The proposed architectures [11–14] involve two integrated CNNs. The first one calculates the true segmentation probabilities. In contrast, the second characterizes individual annotators (e.g., tendencies to over-segment, mix-ups between different classes, etc.) by computing the pixel-wise confusion matrices (CMs) on a per-image basis. These models purport to outperform methods such as STAPLE and its variants by employing deep neural networks to unravel the complex mappings from the input images to the behaviors of the annotators and to the actual segmentation labels. Moreover, the CNNs' parameters are "global variables" optimized over multiple image samples. This feature empowers the model to disentangle the true labels from the annotators' errors robustly, drawing on correlations between analogous image samples. This robustness holds even when the number of available annotations per image is minimal (e.g., a single annotation per image). However, these techniques have not been tested on actual medical data yet, where challenges such as the absence of clear-cut differences among experts may occur, unlike in the artificially created data used in their previous studies. Furthermore, the feasibility of these architectures on small-sized datasets, with their unique challenges, has not been investigated yet. In this study, we implement the methods suggested in the literature [12] on a small-sized dataset of COC, annotated by three experts. This marks the first instance where such techniques are employed with real-world medical data and masks from multiple experts, and not synthetic masks as in the previously mentioned work. Our findings reveal that while these techniques have potential, they may not be entirely effective in this context, and several challenges emerge, suggesting room for refinement

in these proposals. Venturing a step further, we turn our attention towards areas of high uncertainty, as the majority of the areas exhibit considerable agreement among experts. By concentrating on these areas with high predictive uncertainty, we succeed in crafting a unique confusion matrix for each expert. This matrix serves as a signature of each expert's annotation style within the field of Assisted Reproductive Technology.

We also establish a similar 'identity' for the final Deep Learning model and compare the behaviors of the model and human experts. Ultimately, we use these 'identities' within the uncertain areas to derive a new ground truth. This novel ground truth proves to be more reliable than the commonly employed majority-vote or STAPLE methods in the literature, thereby providing a promising direction for future work in this domain.

The structure of this paper is as follows: Following the introduction, we delve into related work in the field in Section 2. This is followed by a detailed description of the methodology we adopted in Section 3. Subsequently, we present our findings and engage in a thoughtful discussion about the implications and significance of our results in Section 4. Finally, in Section 5, we draw conclusions from our study and suggest possible directions for future work.

2. Related Work

Image segmentation in medical data has witnessed numerous advancements due to the introduction and utilization of the UNet architecture [2]. There are numerous cases in the literature demonstrating the effectiveness of utilizing the UNet [3]. Assessing the efficacy of image segmentation techniques is a longstanding challenge. Interactive segmentation by expert annotators is a commonly accepted practice, yet it introduces variability between different experts. A variety of methods has been put forth to estimate annotator skills and establish ground truth labels. These approaches can be broadly categorized into two groups: (i) two-stage approaches [7,15–17] and (ii) simultaneous approaches [12,13,18–20].

In two-stage approaches, label aggregation and supervised model training occur independently. Initially, noisy labels are consolidated using a probabilistic model that incorporates unknown variables—annotator skills and ground truth labels. A machine learning model is then trained on ground truth labels to perform the task. Unfortunately, these methods often neglect raw input information during label aggregation, impacting estimated true labels.

The STAPLE algorithm was introduced by Warfield et al. [15] in 2002. It characterizes each annotator w through a confusion matrix $\theta_w \in \mathbb{R}^{L\times L}$, where L is the number of classes, and $\theta_{w,c',c}$ denotes the probability that expert w labels a pixel as c' given c in consensus. STAPLE employs Expectation Maximization [7] for maximum likelihood consensus segmentation. Enhancements followed, including Spatial STAPLE [16] and Local MAP STAPLE [17]. Non-Local STAPLE (NLS) [10] proposed by Asman and Landman addresses intensity inclusion in consensus computation. MOJITO [21] minimizes Fréchet variance to counter STAPLE sensitivity. Carass et al. [22] used STAPLE for a challenge, revealing experts' superiority over algorithms.

Simultaneous approaches combine supervised model prediction with noisy label handling for enhanced accuracy. Employing expectation maximization, these methods require ample labels. However, real-life label collection constraints limit their practicality.

This category simultaneously curates labels and train models, achieving synergy. Despite favoring basic classification, these approaches improve predictive power and sample efficiency over the first category. Yet, structured prediction tasks with high-dimensional outputs lack attention. Zhang et al. [12] introduced a supervised segmentation method that estimated annotator reliability and true labels using noisy data. While they evaluate their methodology in several synthetic scenarios, it has not been tested on real medical data with masks from multiple annotators. Zhang et al. [13] refined this approach with crowdsourcing. Liu et al. [18] shifted their focus to learning dynamics—a concept previously studied in classification problems but scarcely explored in the realm of image segmentation. They propose the ADELE model, but the results provided are not grounded

in the core issue of multiple annotators with varying annotation methods. Other methods were proposed [19,20,23,24], but none handle real medical data from multiple experts.

Our work represents the first attempt at implementing a simultaneous approach to this tangible problem, using data directly provided by three distinct experts in the field.

3. Materials and Methods

3.1. Dataset

Images from 100 COCs were captured both at the immature stage (0 h of maturation) and at the mature stage (22 h post maturation). This was accomplished using an Olympus stereomicroscope coupled with a ToupCam camera and facilitated by ToupView software (version 3.7.13270.20181102). All images were acquired with a consistent magnification of 56×, centering a single oocyte's zona pellucida in the middle of the field of focus. Images were stored as .png files with a resolution of 2592 × 1944 pixels in RGB format. A total of 100 paired images of COCs were presented to three experts to annotate the COC area, using the ImageJ (version 1.53j) software. Finally, a total of 200 images and 200 × 3 masks were collected. The images and the masks were downscaled to reach the final dimensions of 192 × 240 × 1. The method of majority-vote [5] was used to compute a GT and use it for evaluation purposes.

An example of the data provided by the Department of Internal Medicine, Reproduction and Population Medicine, Ghent University (https://www.ugent.be/di/irp/en, accessed on 9 November 2021), is available in Figure 1.

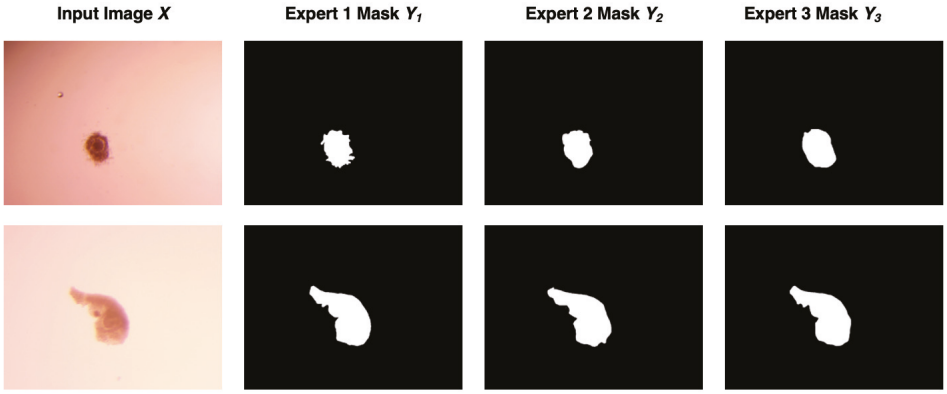


Figure 1. A sample of the dataset. The first column contains a COC in both immature and mature stages. The subsequent three columns represent the corresponding masks provided by different experts. It is obvious that there is no complete agreement among the experts for each case.

3.2. Architecture

Our proposed approach has been deployed utilizing the PyTorch framework. The network dedicated to segmentation tasks employs the UNet structure, featuring a depth of five layers. The number of channels in these layers scales up progressively, starting from 32, then to 64, 128, 256, and finally reaching 512. The annotating network uses a Softmax [25] function for the simpler case of having a global CM, or, for the more complicated local CM case, a double convolutional block, with a convolution function followed by a ReLU function (Figure 2). The dataset, consisting of 200 images, was divided into three subsets: 80% for the training set, 10% for the validation set, and 10% for the test set. It is trained for 100 epochs, with a learning rate starting at 1×10^{-3} and going down to 1×10^{-6} progressively. The training batch size was set to 16, and the optimizer to Adam [26].

3.2.1. Segmentation Network

The segmentation model is undertaken by a UNet architecture [2]. It is parameterized by θ and outputs a predicted probability distribution denoted as $p_\theta(\mathbf{X}) \in [0,1]^{WxHxL}$, with W being the width of the image, H the height of the image, and L being the number

of classes. This distribution represents the likelihood of a pixel belonging to a specific class. For this application, there are only 2 classes ($L = 2$) since there is only the COC area and the background. UNet, with its distinctive U-shaped layout, has become the network of choice for medical image segmentation since this architecture is adept at extracting both low-level and high-level features, thus enhancing the segmentation accuracy. It has been also used previously for similar tasks with COC [6]. Additionally, UNet demonstrates superior performance with smaller datasets, making it the best choice for the current problem. The UNet is available on the left side of Figure 2, linked to the following regularization network.

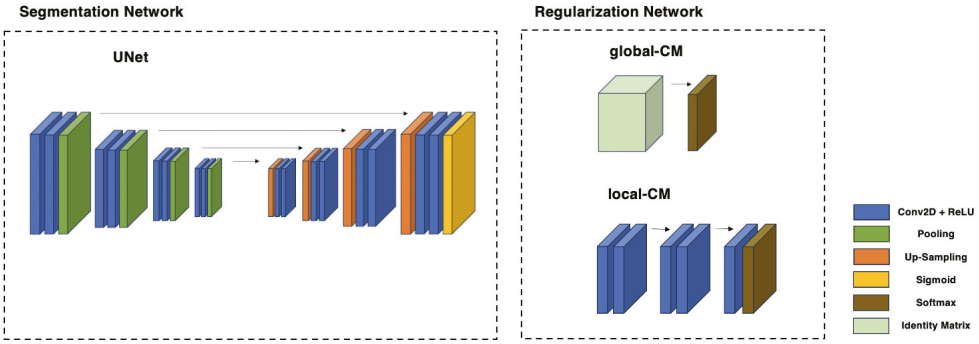


Figure 2. The structure of the segmentation network, which consists of a UNet architecture parameterized by θ and the regularization networks parameterized by ϕ. The UNet has a depth of 5 layers, with the number of channels moving progressively from 32 to 64, 128, 256, and finally, to 512. The maxpooling layer has a padding and stride of 2, while the upsampling layer has a kernel size and a stride of 2. The regularization networks contain a simple network to compute the global confusion matrices and a CNN to compute the local confusion matrices.

3.2.2. Regularization Network

Following [12], the regularization network (Figure 2) is parameterized by ϕ, and its primary goal is to recover the true segmentation probability distribution. This is achievable by emulating the distinct behavior of the three annotators (Figure 3). The term 'behavior' refers to each annotator's unique inclination toward specific "false" segmentation patterns, such as the tendency to assume pixels to be part of the COC area and not of the background. These behaviors are captured using confusion matrices (CMs). Assuming that the true label is previously known, then the CM is constructed as follows:

$$a_{ij}^{(m)}(\mathbf{x}) = p(y^{(m)} = i \mid y_{GT} = j, \mathbf{x}) \tag{1}$$

where $a_{ij}^{(m)}$ stands for the element in the (i, j) cell of the CM for expert m. In an image of size WxH, for every pixel $w \in W$, $h \in H$, the CM could be the same among all pixels (global-CM case), or it could change for each pixel (w, h) of each image $\mathbf{x}$ (local-CM case). Consequently, in global CM cases, there is only one CM representing the whole image, while in local CM cases, there is a CM for each pixel. y_{GT} refers to the GT mask used for this process, and $y^{(m)}$ refers to the mask provided by annotator m. The GT here is not observable, so the model is disentangled to reach its own GT, and the CM is not able to be computed in a straightforward way.

Thus, a regularization network is implemented, which serves to model the confusion matrices. Each of these matrices is the output from a CNN that takes the COC image $\mathbf{x}$ as its input. The CNN is characterized by $\phi(m)$ and shares the same architecture for every CM branch, while it is independently optimized following random initialization. The probability distribution prediction, $p_\phi(\mathbf{x})$, is provided by the segmentation network, as can be seen in the schemas (Figures 3 and 4). By performing element-wise multiplication of

the CMs and this predicted probability distribution, we derive the estimated probability distributions corresponding to each annotator:

$$p_{\theta,\phi}^{(m)}(\mathbf{x}) = A_{\phi}^{(m)}(\mathbf{x}) \cdot p_{\theta}(\mathbf{x}) \tag{2}$$

where $A_{\phi}^{(m)}$ is the total of the CMs with $m \in \{1, 2, 3\}$, and '$\cdot$' operator denotes the element-wise matrix multiplication in (w, h), where needed.

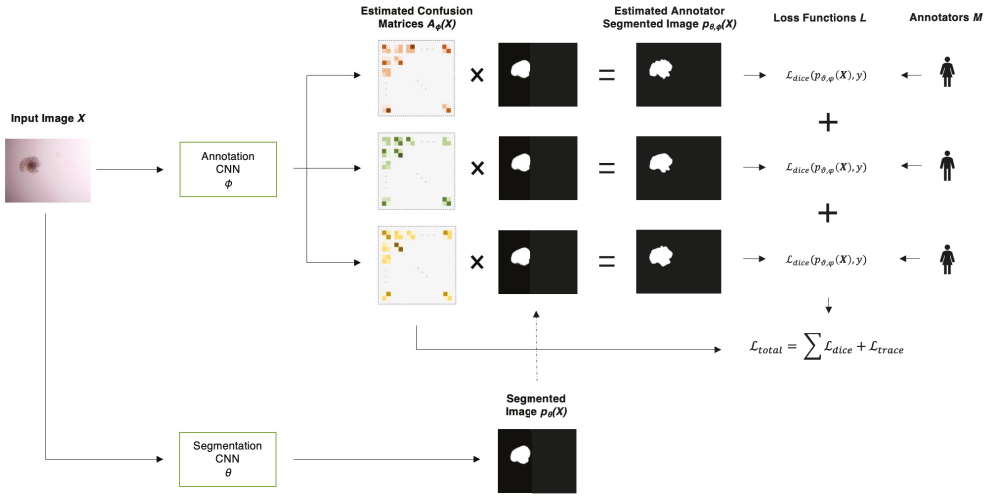


Figure 3. The architecture consists of two components: (a) a segmentation network, characterized by the parameter θ, which produces a probability distribution p_θ for segmentation; and (b) a regularization module consisting of a CNN, parameterized by ϕ, which utilizes the input image to generate three pixel-wise confusion matrices A_ϕ at the local (pixel) level. During the training process, the parameters (θ, ϕ) are learned simultaneously by optimizing the overall loss function.

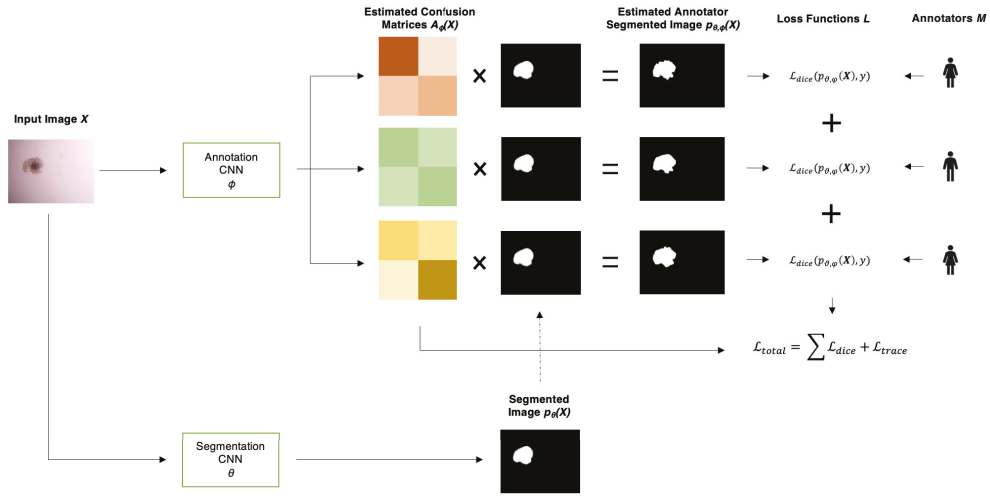


Figure 4. The architecture consists of two components: (a) a segmentation network, characterized by the parameter θ, which produces a probability distribution p_θ for segmentation, and (b) a regularization module consisting of a CNN, parameterized by ϕ, which utilizes the input image to generate three image-wise confusion matrices A_ϕ at the global (image) level. During the training process, the parameters (θ, ϕ) are learned simultaneously by optimizing the overall loss function.

In essence, the creation of the ideal confusion matrix and the corresponding estimated probability distribution is accomplished through a joint optimization process involving both the segmentation network and the regularization network.

3.3. Loss Function and Evaluation Metrics

In this section, we provide a detailed explanation of how we perform a combined optimization of the segmentation network parameters denoted as θ and the annotator network parameters denoted as ϕ. We also discuss the evaluation metrics employed in our study.

To assess the performance of the proposed segmentation, the dice coefficient [27] is selected. The dice coefficient quantifies the spatial concurrence between two regions, providing a range from 0 to 1, where 0 implies no overlap whatsoever, while 1 indicates complete agreement. The corresponding equation is given below (3):

$$dice(f, \mathbf{x}, y) = \frac{2\sum_{ij} f(\mathbf{x})_{ij} y_{ij}}{\sum_{ij} f(\mathbf{x})_{ij} + \sum_{ij} y_{ij}} \tag{3}$$

with y being the ground truth, $\mathbf{x}$ being the input image, and $f(\mathbf{x})$ being the prediction of the model.

To ensure that the estimated probability distribution, incorporating annotator behaviors, closely aligns with the actual annotations, we utilize dice loss. Dice loss for binary segmentation is formulated as follows:

$$\mathcal{L}_{dice}(f, \mathbf{x}, y) = 1 - dice(f, \mathbf{x}, y) \tag{4}$$

Following this, we integrate a regularization term based on the trace norm theorem [11]. The theorem implies that, given a dominant trace of the confusion matrix, the operation of (2) will aim to minimize this trace. This minimization facilitates the alignment of the estimated confusion matrices of individual annotators with the true values. The optimization of the CM's trace is accomplished via the loss function:

$$\mathcal{L}_{trace}(f, \mathbf{x}, y) = \sum_{m=1}^{3} tr(A_{\phi}^{(m)}(\mathbf{x})) \tag{5}$$

with $tr(A)$ denoting the trace of A.

Essentially, minimizing the trace promotes the highest degree of unreliability in the estimated annotators, whereas minimizing the dice loss secures fidelity to the observed annotator inputs. The overall loss function is formulated by integrating both the dice loss and the trace loss:

$$\mathcal{L}_{total}^{\theta,\phi} = \mathcal{L}_{dice}^{\theta} + \gamma \mathcal{L}_{trace}^{\phi} \tag{6}$$

where γ corresponds to a regularization term. To establish the optimal γ, several options are tested, followed by a quantitative comparison from γ = 0.0 to γ = 1.0 with a step of 0.1, while some options are above 1.0 to examine the case of assigning higher importance to the trace loss. By minimizing the combined loss function, the model learns the parameters θ and ϕ.

3.4. Methodology

In our study, we structure our approach into three key stages. Initially, we examine the methods outlined in the existing literature that are focused on training a supervised segmentation model using labels generated by several human experts (Section 3.4.1). This is particularly examined in the context of a limited dataset comprising COC images and associated binary masks annotated by three distinct experts. The primary objective here is to extract the true segmentation exclusively from this collection of noisy labels. Due to some poor results, we switch our focus toward areas of high uncertainty (Section 3.4.2). We further refine our analysis by extracting distinctive annotating behavior profiles for each expert. In the third and final stage (Section 3.4.3), we aim to optimize the ground truth. We underscore the areas of high uncertainty, thus augmenting the precision and reliability of the segmentation model.

The code is available on the following GitHub page: maximum-likelihood-gt (accessed on 19 July 2023).

3.4.1. Jointly Learning

In this research, we implement the suggested methodology [12] comprising of a pair of convolutional neural networks (CNNs). The unique function of these CNNs is to predict the confusion matrix for each expert, which is subsequently used for segmenting the area of the COC, eliminating the need for prior GT.

- **Coupled CNN training:** The initial approach was to implement a local CM model where each image pixel was assigned its individual CM. Given the high dimensionality involved with assigning a CM to each pixel—effectively a $H \times W \times 2 \times 2$ dimensional problem—we decided to revise our strategy, with H being the height of the image, W being the width of it, and a 2×2 confusion matrix for each pixel. (Figure 3)
  To manage the high dimensionality, we adopted a different approach, reducing the multiple CMs to a single one (global CM) and aiming at capturing the behavior across the entire image. This effectively condensed the problem from a $H \times W \times 2 \times 2$ dimension to a more manageable 2×2 CM. (Figure 4)
- **Coupled CNN training with transfer learning:** In order to assist the initial models, examining further options was necessitated. We adopted the method proposed by Athanasiou et al. [6] and trained a segmenting CNN model to segment the COC area proficiently. Subsequently, the weights of the model were saved to serve as a starting point for disentangling the process from the ground truth, negating the need to train both CNNs simultaneously and offering a promising starting point for the training.
  Upon revisiting the approach, two primary concepts stood out. The first entailed training with the pre-trained weights, allowing the models to optimize the weights for both CNNs. The second concept involved training with the pre-trained weights, freezing the segmentation CNN, and enabling the models to train the annotating CNN, thus learning the CMs.

3.4.2. Confusion Matrices on Uncertainty

To derive the confusion matrices in areas of uncertainty, a pre-trained deep learning model that has achieved a high dice score (Athanasiou et al. [6]) for segmenting the COC area is employed. This model is used to pinpoint areas of uncertainty by setting a threshold at 0.05. In this context, every pixel that falls between [0.05–0.95] is considered uncertain. Conversely, any pixel outside this range is deemed a certain segmentation where there should be agreement among experts.

The pixels of interest for each image are identified and directly compared with the GT. In this instance, the GT is defined as the majority-voted masks created by combining the masks provided by experts. By comparing the expert annotations with the majority-vote annotation solely on uncertain pixels, confusion matrices are constructed.

Specifically, for each pixel in the uncertainty range, if an expert annotates it as '1' and the GT indicates '0', it is recorded as a False Positive in the confusion matrix. If the expert marks it as '0' and the GT indicates '1', it is a False Negative. Conversely, if the expert's annotation aligns with the GT, it is recorded as a True Positive or True Negative. This process is applied to all pixels in the uncertain areas of an image.

Following this, we normalize across the rows of actual values to obtain a confusion matrix for each annotator per image. To attain a total confusion matrix for each annotator, we compute the mean of their confusion matrices across all images.

3.4.3. Maximum Likelihood Ground Truth

Once the confusion matrices for each annotator are estimated, it is possible to estimate the ground truth by maximum likelihood. Next, we show how to put together the labels provided by each annotator and their confusion matrices to compute the Maximum Likelihood Ground Truth. Assume that for a pixel, the set of annotators is M, and for each

annotator, $m \in M$ receives a label $y^{(m)}$. We have a CM, $A^{(m)} = (a_{ij}^{(m)})$, where $a_{ij}^{(m)}$ stands for the probability that annotator m labels the pixel as i when the real label for that pixel is j. Also, assume that we have a prior $\rho = (\rho_1, \ldots, \rho_L)$, where ρ_j describes how likely it is that this pixel is from class j, before receiving any annotation of this pixel. We want to compute $p(y_{GT} = j|y^{(M)}, A^{(M)}, \rho)$, which is the probability that the pixel is of class j given all the available information ($y^{(M)} = \{y^{(m)}|m \in M\}$ and $A^{(M)} = \{A^{(m)}|m \in M\}$). Following the results presented in [28] (Equation (2.5)), we can compute it as

$$p(y_{GT} = j|y^{(M)}, A^{(M)}, \rho) = \frac{\alpha_j \rho_j}{\sum_{l \in L} \alpha_l \rho_l}, \tag{7}$$

where $\alpha_j = \prod_{m \in M} a_{y^{(m)}j}^{(m)}$. For a straightforward application, we suggest defining ρ_j as the frequency of label j in the set of annotations.

4. Results and Discussion

In the following section, we present and discuss the outcomes derived from the application of the methodology outlined in the previous sections. These results correspond to (i) the performance of the dual-CNN framework as depicted in existing literature, along with the modifications that we introduced, (ii) the extraction and understanding of the unique annotating behavior profile exhibited by each expert, and (iii) the advancements made in developing a more sophisticated ground truth, focused on the uncertain areas. This analysis aims to shed light on the effectiveness of our multi-faceted approach for addressing the challenges inherent in the segmentation task.

4.1. Performance Coupled CNNs

The performance of the dual-CNN structure did not meet the expectations formed by its previous applications on artificial data. This observation held true across all different iterations, irrespective of the use of global or local CMs and regardless of the use of transfer learning. Multiple factors contributed to this discordance between the expected potential of the methodology and the ultimate results achieved.

Two key challenges became evident during our study. Firstly, the nature of real-world medical problems, such as the present COC investigation, inherently restricts the size of available data. In our scenario, the dataset was composed of a maximum of 200 COC images. Secondly, experts in the medical fields typically exhibit high accuracy in identifying the regions of COCs, thereby diminishing the degree of disagreement amongst them. This relatively high consensus among experts further complicates the task of segmenting and distinguishing various expert behaviors.

Eventually, neither the local CM's nor the global CM's strategy allowed the dual-CNN model to learn the segmentation of COC areas successfully. In both cases, the model began to diverge after a few epochs. Various attempts were made to tackle this issue, including adjusting multiple hyperparameters and the exploration of different γ values to manage the trace loss better. Despite these efforts, the performance remained poor, underscoring the challenge of adapting the model to this specific task.

In a preceding study by Athanasiou et al. [6], transfer learning emerged as an indispensable tool to achieve high performance in COC area segmentation. The first approach, despite starting from an advantageous position in training, could not sustain high performance for both the local CM and global CM implementations. The second approach involved freezing the segmenting CNN to ensure its high performance, shifting the focus towards learning the CMs on the annotating CNN.

While this model could learn three different CMs, the values were strikingly similar across the experts. The True Positive (TP) and True Negative (TN) parts were extremely close to 1.0, and the False Positive (FP) and False Negative (FN) parts were extremely close to 0.0. Upon further examination, we discovered that this was to be expected, as there was

substantial agreement among the experts on the majority of the pixels forming either part of the image background or the COC itself.

The same poor results were achieved even when a pre-trained network (by Athanasiou et al. [6]) was used as the starting point for this approach. Despite its advantageous initial standing, the network could not maintain high performance when applied to either the local CM or global CM variants. In the latter approach, the segmentation CNN was 'frozen' to maintain its high performance, thus shifting the emphasis towards learning the CMs on the annotation CNN. Although this model was able to discern three different CMs, their values were strikingly alike across all the experts, with True Positive (TP) and True Negative (TN) values approximating 1.0 and False Positive (FP) and False Negative (FN) values near 0.0.

As it is already assumed, the high degree of consensus among experts on most of the pixels led to this poor outcome. Finally, we accepted that the proposed approach still needs improvements to fulfill the idea of their creators and alter our focus on the CMs.

4.2. Performance on CMs—Learning

In this phase of our work, our primary aim is to learn the unique CMs of each expert in areas of high uncertainty (Figure 5). Given that the experts largely agree in their assessments, the only viable approach to characterize their distinct annotating "behaviors" lies in these contentious areas. Initially, the areas of high uncertainty are pinpointed. This is achieved through the use of our deep learning model, which discerns the zones where there is an elevated level of indecision regarding pixel classification, as described in Section 3.4. Subsequently, we compare each pixel of every annotator's mask to the majority-vote ground truth. This process is repeated five times, and ultimately, we synthesize a mean confusion matrix for each annotator drawn from the complete dataset. The resulting matrices, which shed light on the unique 'identity' of each expert, are illustrated in Figure 6.

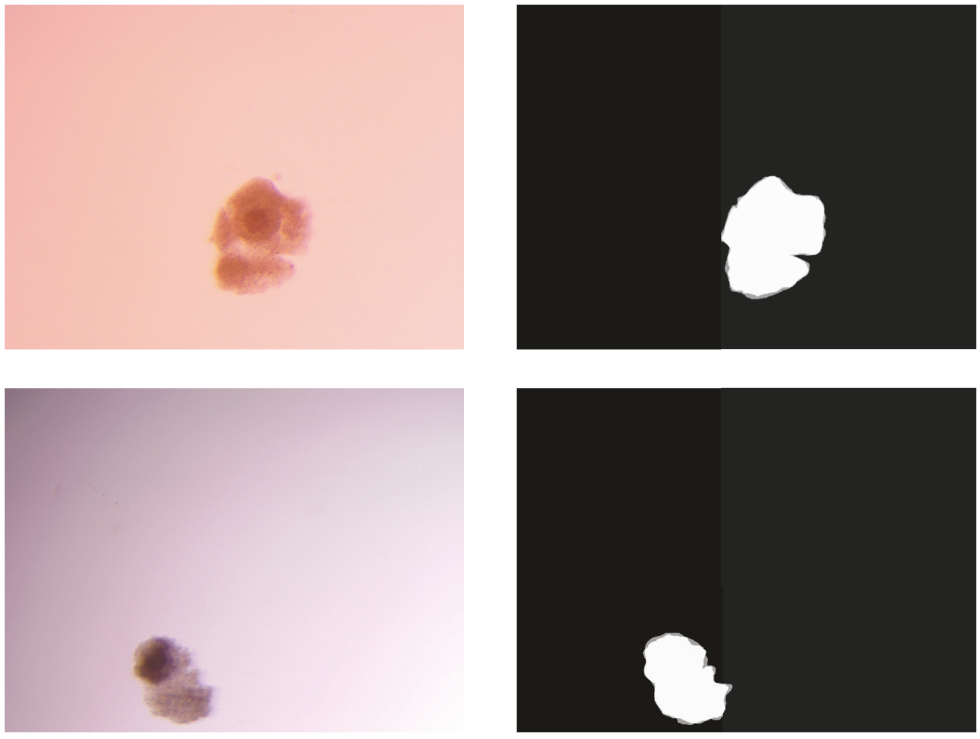


Figure 5. Visualization of uncertainty regions in the segmentation process: On the left-hand side, a sample of the original microscopy images of COC is shown. On the right-hand side, the uncertainty regions corresponding to the sample on the left-hand side are displayed. As is evident, areas of high uncertainty are concentrated along the borders of the cumulus oocyte complexes.

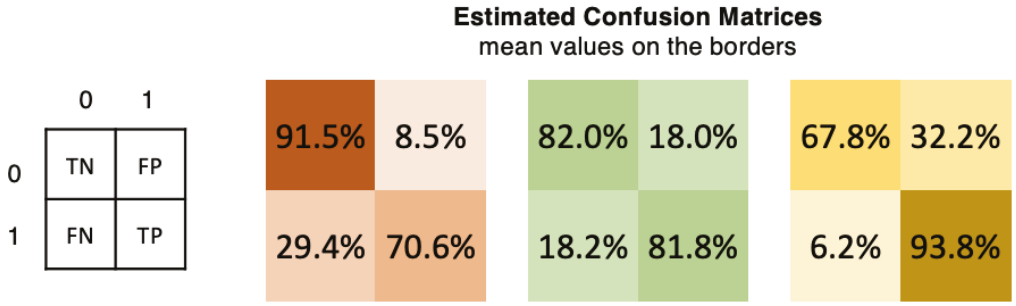


Figure 6. Visualisation of the confusion matrices for each annotator, focusing on the areas of high uncertainty. This representation shows a clear behavior of each expert on the most difficult areas to identify.

At this point, the approach of each expert towards the ambiguous areas—those with the highest likelihood of disagreement—has become clear. The first expert scores significantly high on the True Negative metric but averages on the True Positive one, indicating a greater tendency to categorize a pixel as part of the background and stricter criteria for identifying a pixel as part of the COC. Conversely, the third expert shows a somewhat opposing approach. He appears more lenient in categorizing pixels as part of the COC, even at the risk of falsely including background pixels in the COC region, as evidenced by his high True Positive score and average True Negative score. Lastly, the second expert sits in between these two extremes, exhibiting a more balanced approach. He demonstrates relatively high accuracy in correctly identifying pixels as part of the COC when they indeed are, and as parts of the background when they are not.

4.3. Ground Truth

Using the maximum likelihood approach as described in Section 3.4.3, a new and more sophisticated method for computing the ground truth is proposed, incorporating the personal annotations of experts. An illustrative example of this novel ground truth computation is depicted in Figure 7.

The first approach utilizes a majority vote strategy (Figure 7a). In cases where three experts are involved, an uncertainty level of only 0.67 is achieved when two out of the three experts agree while the third expert disagrees. The second approach (Figure 7b), however, leverages the previously derived confusion matrices from the annotation processes to gain insight into the annotators' behaviors. This knowledge is then utilized to enhance the certainty level for each pixel beyond 0.67, encompassing a range of 0.67–1. Consequently, the final ground truth determination is fortified, as it has the capacity to identify erroneous pixels that may have been produced by the majority vote method. Figure 7c exhibits the disparities in the areas of the image where the majority vote and maximum likelihood methods diverge.

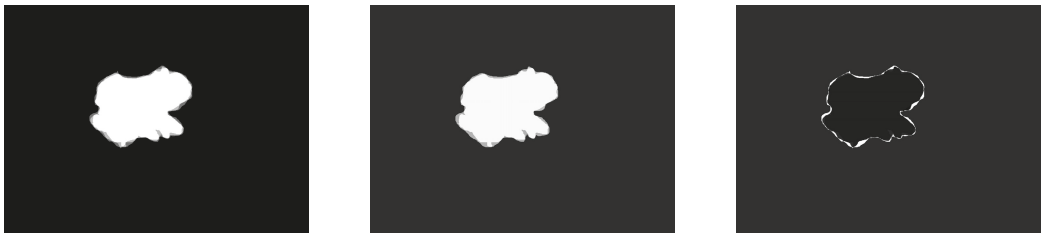
The maximum likelihood method's qualitative superiority over previous approaches is readily apparent. What makes our method stand out is how it tackles existing limitations. We have found that current methods fall short when applied to real-world annotation medical data from assisted reproduction. They were efficient with synthetic data or annotations by non-experts, but in real cases, they need improvement.

Our method is designed to address limitations that have been identified in current practices. It has come to light that these methods fall short when dealing with actual medical data from assisted reproduction, a significant departure from their performance with synthetic data or annotations by non-experts (crowdsourcing). A distinctive feature of our approach is its focus on creating customized profiles for each expert based on their annotations. These profiles, derived using confusion matrices, target areas where experts exhibit discrepancies or uncertainty. This capability is achieved by working with real-world annotation data that accurately capture genuine expert disagreement.

Furthermore, our approach allows us to discern the distinct annotation behaviors of each expert. This step holds importance as it enables the reuse of these behavioral patterns for future annotations, thereby sidestepping the need for repetitive experimentation. This not only conserves time and resources, but also bolsters model training accuracy. Notably, our approach does away with the prerequisite of an odd number of experts, which is a constraint imposed by prevailing methods.

A notable practical advantage of our methodology is its seamless integration into clinical and laboratory environments. This integration permits the preservation and application of personalized annotation profiles, streamlining the process of generating augmented results without the intricacies of sourcing and amalgamating numerous annotations.

In conclusion, our innovative approach revolutionizes the computation of ground truth. It actively addresses shortcomings in current methodologies while delivering tangible benefits in terms of precision and operational efficiency in medical image analysis.



(**a**) The average mask and areas of disagreement.

(**b**) The maximum likelihood mask.

(**c**) The area of disagreement between (**a**,**b**).

Figure 7. A comparison between the majority-vote ground truth and the maximum likelihood ground truth, which focuses on the areas of uncertainty. In (**a**), there is the majority-vote mask, with a gray zone on the borders, for the pixels of disagreement. In (**b**), there is the maximum likelihood mask, which can vary within the range of (0.0–1.0), since it is calculated using the confusion matrix identity of each expert. In (**c**), there is the zone of disagreement and alterations between case (**a**) and case (**b**).

5. Conclusions

The focus of the study was an exploration of multi-annotator segmentation using a COC dataset. This research consisted of three main steps: the application of a coupled CNN architecture to real medical data as proposed in the literature, a focus on uncertain areas to extract individual expert annotating profiles, and an attempt to optimize the ground truth by emphasizing areas of high uncertainty.

A primary finding of this investigation was the distinct disparity between results obtained from artificial data that have been used in previous works, and those from real-life medical datasets. Significantly, strategies that showed considerable success with artificial data proved unsatisfactory when applied to real-life medical data. This underscored the inherent complexities and unique characteristics of real-world annotation medical data, serving as a reminder that solutions developed in idealized environments may not necessarily translate directly to practical applications. Efforts to enhance the model's performance using different strategies, such as transfer learning and variations in CMs, did not yield any notable improvements.

Another obstacle encountered was the difficulty in achieving a consensus among the experts. Although high agreement among experts is generally desirable, it presented a hurdle in the quest to improve the coupled CNN's segmentation model using the individuals' CMs, since minimal knowledge could be obtained in a highly agreeable environment. This suggests that models designed for scenarios with significant annotation disagreement may struggle when the discrepancies are minimal.

In response to these challenges, the focus shifted toward the regions of uncertainty within the data. An opportunity was found to delve deeper into understanding the individual behaviors of each expert, specifically in the context of these ambiguous areas. This introspection proved to be a fruitful endeavor, unveiling unique tendencies and preferences for each expert.

With this newfound understanding of annotator behaviors, a transition was made to redefine the ground truth (GT) from a more informed perspective. This was not an attempt to divorce the GT from expert annotations, but rather, it was a sophisticated endeavor to better integrate expert preferences into the construction of the GT. A novel GT was proposed and formulated based on maximum likelihood and with a specific focus on areas of uncertainty identified through the utilization of the deep learning model. This new approach surpasses the simplistic majority-vote strategy, providing a more sophisticated reflection of expert knowledge in the resulting GT.

The proposed method offers distinct advantages, primarily centered on addressing multi-labeling challenges by focusing on uncertain areas. Additionally, they create personalized annotating profiles for experts, enhancing and assessing future tasks. Moreover, these methods hold practical value in real-world settings, saving time and resources while overcoming odd-expert-number limitations for accurate ground truth determination.

In conclusion, this research illuminated the shortcomings of transferring methodologies developed from artificial data to real-world applications. However, through this exploration, the value of focusing on uncertain areas and individual expert behaviors was revealed. These insights fostered a more sophisticated construction of the ground truth, highlighting the potential for future research in multi-annotator segmentation.

Author Contributions: Conceptualization, G.A., J.L.A. and J.C.; methodology, G.A. and J.C.; software, G.A.; validation, G.A.; formal analysis, G.A. and J.C.; investigation, G.A. and J.L.A.; resources, G.A.; data curation, G.A.; writing—original draft preparation, G.A.; writing—review and editing, G.A. and J.C.; visualization, G.A.; supervision, J.L.A. and J.C. All authors have read and agreed to the published version of the manuscript.

Funding: G.A. is supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 860960 and by project CI-SUSTAIN (PID2019-104156GB-I00), funded by the Spanish Ministry of Science and Innovation. G.A. is a Ph.D. student at the doctoral program for computer science at the Universitat Autonoma de Barcelona. J.C. is supported by the project Humane-AI-net, which has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 952026.

Data Availability Statement: The data were provided by Ghent University, and we are not allowed to share them.

Acknowledgments: We would like to thank Annelies Raes and Ann Van Soom from the Department of Internal Medicine, Reproduction and Population Medicine, Faculty of Veterinary Medicine, Ghent University, Merelbeke, Belgium, for providing us with the data to perform our research.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CM	Confusion Matrix
CMs	Confusion Matrices
GT	Ground Truth
DL	Deep Learning
COC	Cumulus Oocyte Complex
COCs	Cumulus Oocyte Complexes
ART	Assisted Reproductive Technology

References

1. Menze, B.H.; Jakab, A.; Bauer, S.; Kalpathy-Cramer, J.; Farahani, K.; Kirby, J.; Burren, Y.; Porz, N.; Slotboom, J.; Wiest, R.; et al. The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). *IEEE Trans. Med. Imaging* **2015**, *34*, 1993–2024. [CrossRef]
2. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv* **2015**, arXiv:1505.04597.
3. Azad, R.; Aghdam, E.K.; Rauland, A.; Jia, Y.; Avval, A.H.; Bozorgpour, A.; Karimijafarbigloo, S.; Cohen, J.P.; Adeli, E.; Merhof, D. Medical Image Segmentation Review: The success of U-Net. *arXiv* **2022**, arXiv:2211.14830.
4. Harvey, H.; Glocker, B. A Standardised Approach for Preparing Imaging Data for Machine Learning Tasks in Radiology. In *Artificial Intelligence in Medical Imaging*; Springer: Cham, Switzerland, 2019.
5. Nguyen, N.T.T.; Le, P.B. Topological Voting Method for Image Segmentation. *J. Imaging* **2022**, *8*, 16. [CrossRef]
6. Athanasiou, G.; Cerquides, J.; Arcos, J.L. Detecting the Area of Bovine Cumulus Oocyte Complexes Using Deep Learning and Semantic Segmentation. In Proceedings of the CCIA 2022: 24th International Conference of the Catalan Association for Artificial Intelligence, Sitges, Spain, 19–21 October 2022; pp. 249–258. [CrossRef]
7. Warfield, S.K.; Zou, K.H.; Wells, W.M. Simultaneous truth and performance level estimation (STAPLE): An algorithm for the validation of image segmentation. *IEEE Trans. Med. Imaging* **2004**, *23*, 903–921. [CrossRef] [PubMed]
8. Iglesias, J.E.; Sabuncu, M.R.; Leemput, K.V. A unified framework for cross-modality multi-atlas segmentation of brain MRI. *Med. Image Anal.* **2013**, *17*, 1181–1191. [CrossRef]
9. Cardoso, M.J.; Leung, K.; Modat, M.; Keihaninejad, S.; Cash, D.; Barnes, J.; Fox, N.C.; Ourselin, S.; for the Alzheimer's Disease Neuroimaging Initiative. STEPS: Similarity and Truth Estimation for Propagated Segmentations and its application to hippocampal segmentation and brain parcelation. *Med. Image Anal.* **2013**, *17*, 671–684. [CrossRef] [PubMed]
10. Asman, A.J.; Landman, B.A. Non-local statistical label fusion for multi-atlas segmentation. *Med. Image Anal.* **2013**, *17*, 194–208. [CrossRef] [PubMed]
11. Tanno, R.; Saeedi, A.; Sankaranarayanan, S.; Alexander, D.C.; Silberman, N. Learning from noisy labels by regularized estimation of annotator confusion. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; pp. 11244–11253.
12. Zhang, L.; Tanno, R.; Xu, M.C.; Jin, C.; Jacob, J.; Ciccarelli, O.; Barkhof, F.; Alexander, D.C. Disentangling human error from the ground truth in segmentation of medical images. In Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS'20, Red Hook, NY, USA, 6–12 December 2020; pp. 15750–15762.
13. Zhang, J.; Zheng, Y.; Hou, W.; Jiao, W. Leveraging non-expert crowdsourcing to segment the optic cup and disc of multicolor fundus images. *Biomed. Opt. Express* **2022**, *13*, 3967–3982.
14. Hashmi, A.A.; Agafonov, A.; Zhumabayeva, A.; Yaqub, M.; Takáč, M. In Quest of Ground Truth: Learning Confident Models and Estimating Uncertainty in the Presence of Annotator Noise. *arXiv* **2023**, arXiv:2301.00524.
15. Warfield, S.K.; Zou, K.H.; Wells, W.M. Validation of image segmentation and expert quality with an expectation-maximization algorithm. In Proceedings of the Fifth International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Part I, Tokyo, Japan, 25–28 September 2002.
16. Asman, A.J.; Landman, B.A. Formulating spatially varying performance in the statistical fusion framework. *IEEE Trans. Med. Imaging* **2012**, *31*, 1326–1336. [CrossRef] [PubMed]
17. Commowick, O.; Akhondi-Asl, A.; Warfield, S.K. Estimating a reference standard segmentation with spatially varying performance parameters: Local MAP STAPLE. *IEEE Trans. Med. Imaging* **2012**, *31*, 1593–1606. [CrossRef] [PubMed]
18. Liu, S.; Liu, K.; Zhu, W.; Shen, Y.; Fernandez-Granda, C. Adaptive Early-Learning Correction for Segmentation From Noisy Annotations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2606–2616.
19. Wang, C.; Gao, Y.; Fan, C.; Hu, J.; Lam, T.L.; Lane, N.D.; Bianchi-Berthouze, N. AgreementLearning: An End-to-End Framework for Learning with Multiple Annotators without Groundtruth. *arXiv* **2021**, arXiv:2109.03596.
20. Rottmann, M.; Reese, M. Automated Detection of Label Errors in Semantic Segmentation Datasets via Deep Learning and Uncertainty Quantification. *arXiv* **2023**, arXiv:2207.06104.
21. Hamzaoui, D.; Montagne, S.; Renard-Penna, R.; Ayache, N.; Delingette, H. MOrphologically-Aware Jaccard-Based ITerative Optimization (MOJITO) for Consensus Segmentation. In *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging*; Sudre, C.H., Baumgartner, C.F., Dalca, A., Qin, C., Tanno, R., Van Leemput, K., Wells, W.M., III, Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2022; pp. 3–13. [CrossRef]
22. Carass, A.; Roy, S.; Jog, A.; Cuzzocreo, J.L.; Magrath, E.; Gherman, A.; Button, J.; Nguyen, J.; Prados, F.; Sudre, C.H.; et al. Longitudinal multiple sclerosis lesion segmentation: Resource and challenge. *NeuroImage* **2017**, *148*, 77–102. [CrossRef] [PubMed]
23. Guo, X.; Lu, S.; Yang, Y.; Shi, P.; Ye, C.; Xiang, Y.; Ma, T. Modeling Annotator Variation and Annotator Preference for Multiple Annotations Medical Image Segmentation. In Proceedings of the 2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Las Vegas, NV, USA, 6–8 December 2022; pp. 977–984. [CrossRef]
24. Prados, F.; Ashburner, J.; Blaiotta, C.; Brosch, T.; Carballido-Gamio, J.; Cardoso, M.J.; Conrad, B.N.; Datta, E.; Dávid, G.; Leener, B.D.; et al. Spinal cord grey matter segmentation challenge. *NeuroImage* **2017**, *152*, 312–329. [CrossRef] [PubMed]
25. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
26. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.

27. Dice, L.R. Measures of the Amount of Ecologic Association between Species. *Ecology* **1945**, *26*, 297–302. [CrossRef]
28. Dawid, A.P.; Skene, A.M. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *J. R. Stat. Soc. Ser. C (Appl. Stat.)* **1979**, *28*, 20–28. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Enhanced Checkerboard Detection Using Gaussian Processes

Michaël Hillen *, Ivan De Boi, Thomas De Kerf, Seppe Sels, Edgar Cardenas De La Hoz, Jona Gladines, Gunther Steenackers, Rudi Penne and Steve Vanlanduit

InViLab, University of Antwerp, Groenenborgerlaan 171, B-2020 Antwerp, Belgium;
ivan.deboi@uantwerpen.be (I.D.B.); thomas.dekerf@uantwerpen.be (T.D.K.); seppe.sels@uantwerpen.be (S.S.);
edgar.cardenas@uantwerpen.be (E.C.D.L.H.); jona.gladines@uantwerpen.be (J.G.);
gunther.steenackers@uantwerpen.be (G.S.); rudi.penne@uantwerpen.be (R.P.);
steve.vanlanduit@uantwerpen.be (S.V.)
* Correspondence: michael.hillen@uantwerpen.be

Abstract: Accurate checkerboard detection is of vital importance for computer vision applications, and a variety of checkerboard detectors have been developed in the past decades. While some detectors are able to handle partially occluded checkerboards, they fail when a large occlusion completely divides the checkerboard. We propose a new checkerboard detection pipeline for occluded checkerboards that has a robust performance under varying levels of noise, blurring, and distortion, and for a variety of imaging modalities. This pipeline consists of a checkerboard detector and checkerboard enhancement with Gaussian processes (GP). By learning a mapping from local board coordinates to image pixel coordinates via a Gaussian process, we can fill in occluded corners, expand the board beyond the image borders, allocate detected corners that do not fit an initial grid, and remove noise on the detected corner locations. We show that our method can improve the performance of other publicly available state-of-the-art checkerboard detectors, both in terms of accuracy and the number of corners detected. Our code and datasets are made publicly available. The checkerboard detector pipeline is contained within our Python checkerboard detection library, called PyCBD. The pipeline itself is modular and easy to adapt to different use cases.

Keywords: checkerboard detection; Gaussian processes; occlusions

MSC: 60G15; 68T10

1. Introduction

Checkerboard detection is a fundamental tool in computer vision applications such as camera calibration [1–6], projector-camera systems [7,8], simultaneous localisation and mapping (SLAM) [9], and robotics in general [10–12]. This topic is of such high importance that it has received a large amount of attention from the community over the past decades and a large variety of detection methods have been developed.

Generally, checkerboard detection involves corner detection, corner refinement, and checkerboard structure recovery [2,3,13]. The locations of the inner corners of a checkerboard pattern (X-type corners) in an image are found with a corner detector. In many cases, the additional refinement of the found corner locations is necessary to achieve sub-pixel accurate locations. The structure of the checkerboard is recovered by utilising the locations and the topology of the detected corners, and the detected corners are mapped to local checkerboard coordinates.

A wide variety of automatic checkerboard detectors already exist. OpenCV's [14] standard checkerboard detector, findChessboardCorners, is based on the work of [15]. An improved version of this detector is proposed in [16], which is better suited for blurred and distorted images. More recently, a new detector called findChessboardCornersSB was added to OpenCV [17], which is more robust to all sorts of noise and runs faster on large

images compared to findChessboardCorners. The major drawbacks of the previously mentioned detectors are that they require the size of the checkerboard, and a thick white edge around the checkerboard is necessary for optimal performance. A checkerboard detector that does not require the size of the checkerboard was introduced in [13]. Their structure recovery algorithm cannot cope with occlusions, which is particularly problematic when an occlusion occurs near the centre of the checkerboard. OCPAD [18], which is a successor to ROCHADE [19], and the newer version of OCamCalib [20] are both able to detect occluded checkerboards. Convolutional neural networks (CNNs) are becoming more ubiquitous for a large variety of detection tasks, including checkerboard corner detection [2–4,21–23]. Both [21] and [3] have developed CNN-based checkerboard detectors. Chen et al. improved upon the structure recovery algorithm by Geiger et al. so it can be used to detect occluded checkerboards. At the time of writing, both OpenCV implementations, OCamCalib, and the detector by Geiger et al. are publicly available. Zhang et al. share their code and data, but the trained weights for the CNN are omitted.

Many real world applications that rely on checkerboard detection suffer from low quality images, drastically impacting the corner detection performance. This could be due to low resolution, artefacts in the lenses of the camera used such as scratches or faults in the glass, contamination on the lens itself, heavily warped images from fisheye lenses, etc. An example where these factors accumulate is provided in Section 4.3. The image was taken by a camera used in endoscopy. Besides the fisheye warping and the contamination in the lens, we can also see some corners being missed due to the specular reflection. The result is a checkerboard that is only partly detected.

In this work, we address the above mentioned problems by proposing a new checkerboard detection pipeline with additional checkerboard enhancement using Gaussian processes (GP) [24], which can be performed after standard checkerboard detection. This machine learning technique is used to learn a mapping from local board coordinates to pixel coordinates, which can be used to predict the image coordinates for undetected or occluded corners and to smooth out the checkerboard corner image locations.

All of our code and the used datasets are publicly available on GitHub. The checkerboard detector and enhancement algorithms are contained within our Python checkerboard detection library, called PyCBD (https://github.com/InViLabUAntwerp/PyCBD, accessed on 30 October 2023) . This library is modular and easy to adapt to different use cases.

The rest of the paper is structured as follows: Section 2 describes the methodology implemented in the enhancement using Gaussian processes. Section 3 explains our experimental setup. We provide the results in Section 4 and discuss them in Section 5. Finally, we formulate our conclusions in Section 6.

2. Methodology

The complete pipeline is summarised in Figure 1. In this section, we will explain the Gaussian process enhancement.

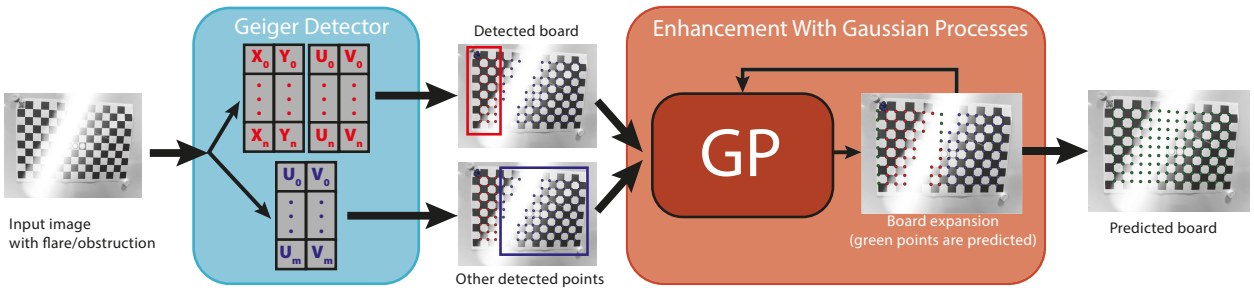


Figure 1. The proposed processing pipeline for enhancing a detected checkerboard using GP.

2.1. Gaussian Processes

As a probabilistic machine learning technique, Gaussian processes (GP) can be used to perform non-linear regression [24]. In our work, GPs are used to learn a function or mapping from local board coordinates to pixel coordinates on an image.

Let $\mathcal{D} = \{X, \mathbf{y}\}$ be a dataset of n observations, where $X = \begin{bmatrix}\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\end{bmatrix}^T$ is an $n \times d$ matrix of n input vectors of dimension d and $\mathbf{y} = \begin{bmatrix}y_1, y_2, ..., y_n\end{bmatrix}^T$ is a vector of continuous-valued scalar outputs. These data points are also called training points. Regression aims to find a mapping $f : \mathbb{R}^d \to \mathbb{R}$,

$$y = f(\mathbf{x}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2), \tag{1}$$

with ϵ being identically distributed observation noise. We implement a Gaussian process to realise this mapping.

By definition, a Gaussian process can be defined as a continuous collection of random variables, any finite subset of which is normally distributed as a continuous multivariate distribution. In our work, the data point inputs are the local xy-coordinates of corners in the checkerboard and the outputs are the uv pixel coordinates of those corners. Those inputs are jointly Gaussian, which means they are not independent but co-vary. The uv-coordinates of one corner provide some information about a corner nearby. This amount of information decreases with distance.

A Gaussian process is fully defined by its mean $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$, in which $\mathbf{x}$ and $\mathbf{x}'$ are inputs. It is denoted as $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$. We can always normalise the data to zero mean [24]. The covariance function on the other hand shows us how much the data points influence each other as a function of their relative distance. Thus, a GP yields a distribution over functions that have a joint normal distribution,

$$\begin{bmatrix}\mathbf{f} \\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{m}_X \\ \boldsymbol{m}_{X_*}\end{bmatrix}, \begin{bmatrix}K_{X,X} & K_{X,X_*} \\ K_{X_*,X} & K_{X_*,X_*,}\end{bmatrix}\right) \tag{2}$$

where X and X_* are the input vectors of the n observed training points and n_* the unobserved test points, respectively. The mean values for X and X_* are given by $\boldsymbol{m}_X$ and $\boldsymbol{m}_{X_*}$. The covariance matrices $K_{X,X}$, K_{X_*,X_*}, $K_{X_*,X}$, and K_{X,X_*} are constructed by evaluating k at their respective pairs of data points. In practice, we do not have access to the latent function values directly, but are dependent on the noisy observations $\mathbf{y}$.

The conditional predictive posterior distribution of the GP can be written as:

$$\mathbf{f}_* | X, X_*, \mathbf{y}, \boldsymbol{\theta}, \sigma_\epsilon^2 \sim \mathcal{N}(\mathbb{E}(\mathbf{f}_*), \mathbb{V}(\mathbf{f}_*)) \tag{3}$$

$$\mathbb{E}(\mathbf{f}_*) = \boldsymbol{m}_{X_*} + K_{X_*,X}\left[K_{X,X} + \sigma_\epsilon^2 I\right]^{-1}\mathbf{f} \tag{4}$$

$$\mathbb{V}(\mathbf{f}_*) = K_{X_*,X_*} - K_{X_*,X}\left[K_{X,X} + \sigma_\epsilon^2 I\right]^{-1}K_{X,X_*}. \tag{5}$$

There exists a large variety of covariance functions, also called kernels [25]. A common choice is the squared exponential kernel, which we also implement in our work. It is infinitely differentiable and thus yields smooth functions. This is a reasonable assumption to make, as our checkerboards do not show discontinuities. It has the form:

$$k_{SE}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{|\mathbf{x} - \mathbf{x}'|^2}{2l^2}\right), \tag{6}$$

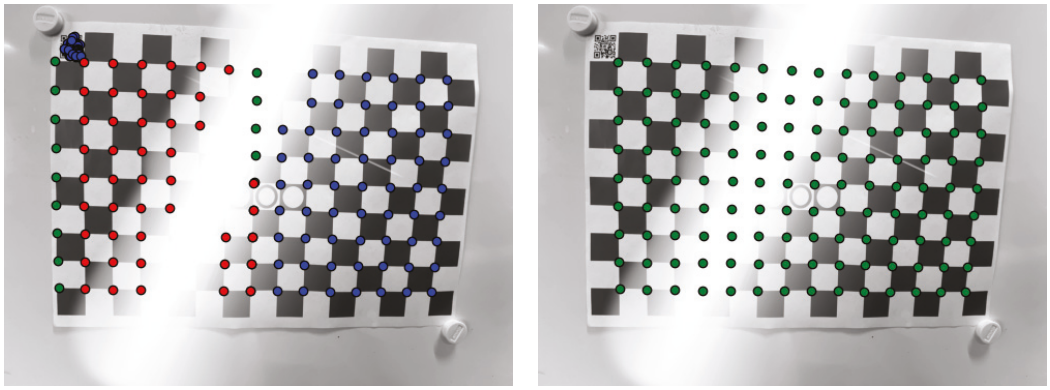
where σ_f^2 is a height-scale factor and l the length-scale that determines the radius of influence of the training points. For the squared exponential kernel, the hyperparameters are σ_f^2 and l. The values for the hyperparameters, which we combine in the symbol $\boldsymbol{\theta}$, are learned by using a gradient based optimisation algorithm to maximise the log marginal likelihood,

$$\log p(\mathbf{y}|\theta, X) \propto -\frac{1}{2}\left[\mathbf{y}^T\left[K_{X,X} + \sigma_\epsilon^2 I\right]\mathbf{y} + \log|K_{X,X} + \sigma_\epsilon^2 I)|\right], \tag{7}$$

which is a combination of a data fit term and a complexity penalty and thus automatically incorporates Occam's Razor [24]. This guards the GP against overfitting. In our experiments, we use L-BFGS, a quasi-Newton method.

2.2. Allocation of Detected Corners

Many checkerboard detectors perform checkerboard corner detection followed by a structure recovery algorithm that allocates these detected corners to positions in a grid, based on their location and topology. It is possible that not all detected corners are allocated to this grid because they do not fit according to the structure recovery algorithm. These detected corners could either be false positives, i.e., detected corners that are not checkerboard corners, or checkerboard corners that are somehow missed by the structure recovery step. The latter, for instance, could be the result of a large flare across the image, splitting the checkerboard into two distinct islands of ordered corners. An example of this is given in Figure 2.



(**a**) (**b**)

Figure 2. (**a**) Visualisation of iteration 4: red are corners for which a local coordinate is found, blue are the remaining corners without a local coordinate and green are the predicted corner locations. Notice how the original corner detection also detected corners in the QR code in the upper left corner of the image. As these corner coordinates do not correspond to any predicted values of the GP, they are deemed as false positives and filtered out. (**b**) The entire re-predicted checkerboard is in green.

In this section, we describe an algorithm to expand a partially detected checkerboard with detected corners that are part of the checkerboard pattern, but were not assigned to the grid by the structure recovery algorithm. By learning the map from local board coordinates to pixel coordinates, we can make predictions for missing grid points, which in turn can be used to add unassigned corners to the grid. This section relates to the last (red) box of our method in Figure 1. Our Algorithm is given in pseudo-code in Algorithm 1.

Our algorithm takes as input a partially detected checkerboard, which has two sets of coordinates for each corner: a set of image coordinates (boardUV) and a set of local grid coordinates (boardXY), and the image coordinates of the unassigned detected corners (cornersUV), which do not have local grid coordinates. The aim of the algorithm is to find the corresponding local grid coordinates (xy) for any remaining corners, filter out corners that are not part of the checkerboard, and find occluded corners. Each of these three sets could be empty. This is not known at the start.

The central idea behind our algorithm is the mapping from local grid coordinates (xy) of points on a checkerboard to pixel coordinates (uv) of points in a given image. This mapping is learned via Gaussian processes. We implemented two distinct GPs in parallel to perform regression from xy-coordinates to both the u-coordinates and the v-coordinates of detected corners. They both take a 2D point as input and each produce a scalar value.

In Algorithm 1, the image coordinates of the corners are called cornerUV. The image and local coordinates of the corners that are already on a (possibly impartial) checkerboard are called boardUV and boardXY, respectively.

Algorithm 1 Algorithm to allocate all detected corners.

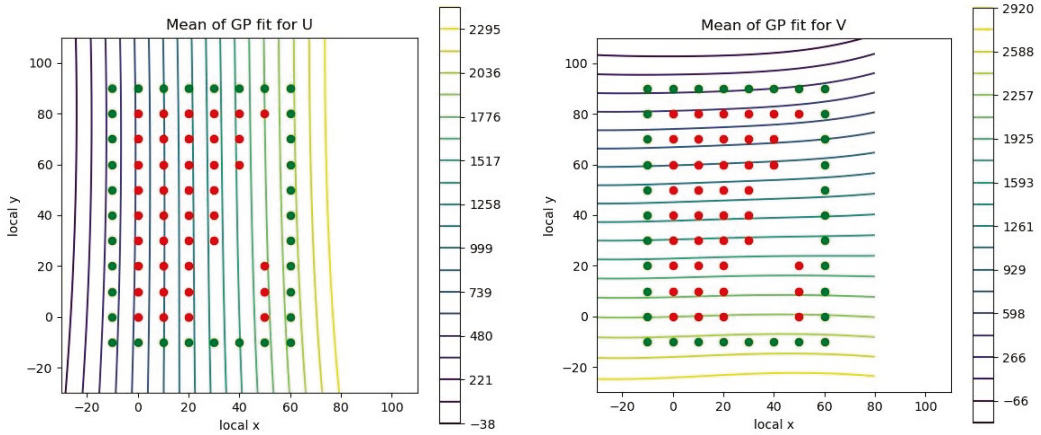
Require: *boardXY*, *boardUV*, *cornerUV*
  maxNrOfIterations = 10
  currIteration = 1
  while *true* **do**
    $GPs \leftarrow boardXY, boardUV$
    Expand local grid of corners with *newXY*
    Predict *newUV* for given *newXY*
    Match *newUV* with existing *cornerUV*
    Augment *boardXY*, *boardUV*
    currIteration++
    if *cornerUV* = *boardUV* **or**
     currIteration > maxNrOfIterations **or**
     no *newUV* in image
     then
      break ▷ leave loop
    end if
  end while
  No more corners to be allocated
  return *boardUV* ▷ possibly augmented

In the second step, for each of the predicted corners, we search for a viable detected corner. If a corner is found, then it is added to the list of checkerboard corners. Its xy-coordinates are the input of the GP and its uv-coordinates are the detected values (not the predicted ones). If no corner can be found within a reasonable distance, which is a fraction of the distance between the checkerboard corners, then this prediction is dropped. This fraction is a hyperparameter that can be tuned for the specific application. If the value is too large, this might result in more false positives being accepted or corners being attributed to the wrong position in the grid. Naturally, this is less of a problem when there are few or no false corner detections. If the value is too small, it might result in too few corners being added to the set of accepted corners. The latter manifests itself more when we are dealing with heavily warped images.

We repeat these steps, each time with a list of corners that is extended with the newly found ones. If no new corners can be found, then we expand the local xy-coordinates with two rows or columns instead of one. This allows us to bridge large gaps.

We keep iterating these steps until a maximum number of steps have been reached, no new points are within the image, or no more corners are without corresponding xy-coordinates (all corners are accounted for). Corners not close to a predicted corner are deemed false positives or outliers. These are falsely detected corners and are ignored on the checkerboard.

At the end of the algorithm, all detected corners are either part of the checkerboard or considered a false positive. A visualisation of a situation after a few iterations is given in Figure 3.



(**a**) (**b**)

Figure 3. An example of a situation given by the algorithm after four iterations. The red dots are the locations of detected checkerboard corners on a perfect checkerboard. The green dots are locations where we predict a value for the u and v coordinates of those corners, in (**a**,**b**), respectively. The values for the coordinates are given by the contour lines. Notice how these contour lines capture the curvature of the checkerboard in the real world. A flat checkerboard would result in straight lines for the contour lines. The deviation from this is the result of the bending of the paper checkerboard as seen in Figure 2.

2.3. Gaussian Process Refinement

After running the algorithm described in Section 2.2, we retrain the GPs one more time on all found corners. This allows us to exploit the GP in two ways.

First, we make predictions for the local xy-coordinates of corners that are without corresponding uv-coordinates. These are locations in the image where no corner is detected, even though the checkerboard tells us there should be one. This enables us to fill in occluded corners and even corners that are outside the borders of the image. An example can be seen in Figure 2, where a large flare hides some corners.

Second, we re-predict the uv-coordinates for all detected corners. In this work, we implemented a squared exponential kernel, which yields smooth functions that are infinitely many times differentiable [24]. This results in an extra refinement step, which removes noise from those predicted coordinates. This is due to the fact that the GP utilises the uv-coordinates of all corners to predict a single one and does not base its prediction only on the surrounding pixels.

3. Experimental Setup

3.1. Dataset Generation

In this work, two datasets are generated. The first is a synthetic dataset, where checkerboards are generated artificially and drawn on random background images. The background images are generated through DALL-E 2 and are made to resemble a typical laboratory environment. When creating the checkerboards on the background images, the ground truth points are known. When augmenting the images, these reference points are transformed in the same way, thus obtaining the actual ground truth for the checkerboard locations. The second dataset, referred to as the "real dataset", is generated using two types of cameras, including a thermal infrared (IR) camera (Xenics Ceres) and a snapshot multispectral imaging (MSI) camera (Photonfocus MV0-D2048x1088-C01-HS02-G2). The multispectral and thermal infrared cameras are specifically chosen because they pose a challenge to current checkerboard detection due to heavy blurring or limited contrast between checkers.

When using a synthetic dataset, the exact sub-pixel location of the corners can be easily determined with high accuracy. This level of precision is difficult to obtain when working with real datasets. Several different types of distortions are used to create real-

istic environmental conditions [26]. These techniques include Gaussian blur, shot noise, optical distortions—such as pincushion, moustache, and barrel distortion—perspective transformations, and rotational transformations, as is common in the literature [21]. For the Gaussian blur, multiplicative noise, and projective distortions, different levels of augmentations are used, from slightly distorted to heavily distorted. For each category and level of augmentation, 100 images are generated and evaluated.

3.2. Evaluation Methods

For the synthetic dataset, two metrics are used to evaluate the different methods: the amount of fully detected checkerboards and the pixel error of the corner locations. A fully detected checkerboard refers to cases where all the checkerboard corners are correctly detected by the algorithm. A checkerboard corner is considered to be correctly detected if its predicted location is within a Euclidean distance of two pixels from the true location. The pixel error is computed as the average Euclidean distance between the predicted and true locations of all the checkerboard corners in the image. The lower the value of this metric, the better the accuracy of the algorithm in predicting the location of the checkerboard corners. The subsequent processing of the image will yield better results.

4. Results

The results of both datasets, simulated and real, are discussed separately below. Four different methods were compared to each other: the method as proposed by Geiger with and without Gaussian enhancement (Geiger, Geiger + GP) and the industry standards OpenCV (OpenCV) and OpenCV Sector Based approach (OpenCV SB) [17]. The Geiger detector we used is a lightly modified version of the implementation in the libcbdetect library [27]. It should be noted that the simulated images are tailored for use with OpenCV: there are clear white and black edges, a thick white border around the checkerboard and the checkerboard are always completely in the image; not doing so would result in a lower detection rate.

4.1. Simulated Data Results

The results of the synthetic images, with varying degrees of blur, shot noise, and perspective transformations, are displayed in Figure 4a–c. Each of these figures is composed of two graphs. The left graph displays the number of fully detected checkerboards as a percentage of the total. The right graph shows the average error for each method. To obtain a fair comparison between the methods, the average of a single checkerboard is only taken into account when four or more methods find all the points of the respective checkerboard.

The Geiger detector, and by extension the proposed Geiger + GP method, fails to detect corners under a lower degree of blurring and shot noise compared to the OpenCV detectors, as can be seen in the left graphs in Figure 4a,b. The right graphs in Figure 4a,b show that the GP corner refinement is able to reduce the corner position error when blur and shot noise is introduced into the images. The Geiger method is able to detect more corners under higher degrees of perspective transformation compared to the OpenCV methods, and the GP enhancement is able to improve this further for higher degrees of perspective scaling. The OpenCV SB method performs notably worse for all degrees of perspective transformation. This is shown in the left graph of Figure 4c.

When different types of optical distortion are applied, all checkerboard detection methods are able to detect most if not all checkerboards, except for OpenCV SB, as shown in Table 1. The mean pixel error is similar for all methods but worst for the Geiger detector. The GP corner refinement is able to slightly improve this result, but not enough for it to be better than the OpenCV methods, as shown in Table 1.

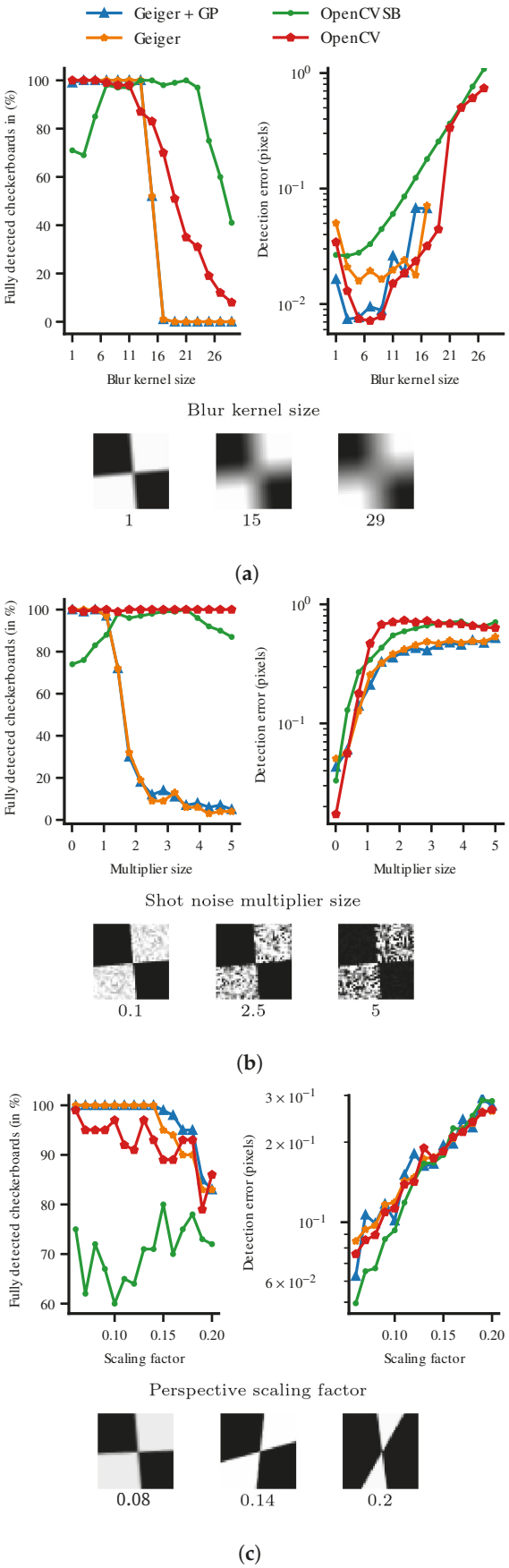


Figure 4. Results of testing different methods for their performance on (**a**) checkerboard blur, (**b**) shot noise, and (**c**) perspective distortion.

Table 1. Results for different types of distortion.

Fully Detected Checkerboards (%)				
Type of Distortion	**Geiger**	**Geiger + GP**	**OpenCV**	**OpenCV SB**
Barrel	100	100	100	65
Mustache	100	100	100	49
Pincushion	85	100	100	85
Average Corner Error (Pixels)				
Type of Distortion	**Geiger**	**Geiger + GP**	**OpenCV**	**OpenCV SB**
Barrel	0.078	0.069	0.051	0.032
Sunglints	0.085	0.066	0.063	0.048
Pincushion	0.179	0.178	0.170	0.140

When there are occlusions added to the images, we see that, using the GP enhancement step, it is still possible to obtain fully detected checkerboards. The average pixel error for the occluded pixels is still in the sub-pixel range, as can be seen in Table 2.

Table 2. Results for increasing amounts of occluded corners introduced into the dataset. Only the results for the Geiger + GP method are shown here. The OpenCV methods are not included as they are unable to cope with occlusions. The reported pixel error is only for specific occluded pixels.

Number of Missing Corners	Average Corner Error (Pixels)
1	0.0226
3	0.0589
5	0.1023

4.2. Real Data Results

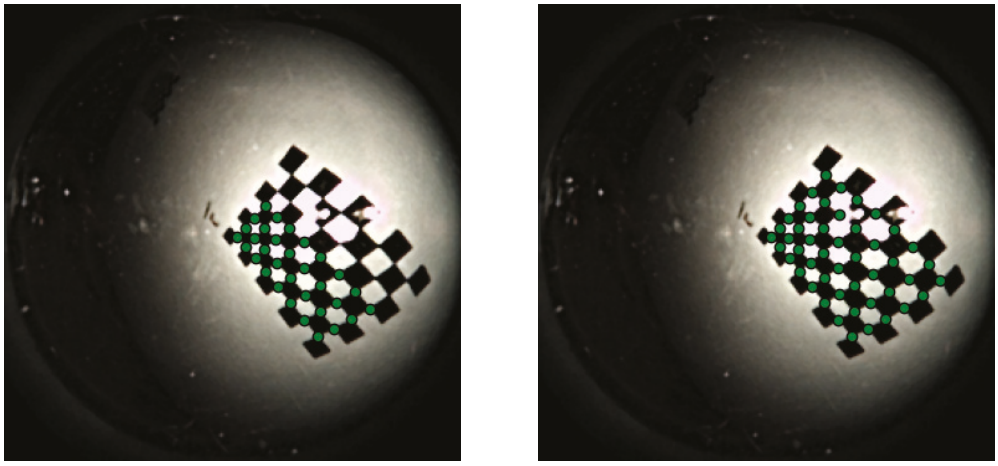
When reviewing the results in Table 3, it becomes evident that significant variations exist among the different datasets. However, the Geiger method with the proposed Gaussian process enhancement consistently demonstrates the best overall performance. The checkerboards in the IR dataset originally have white checkers and dark borders; in other words, the checkerboards in the images look like the inverse of regular checkerboard targets. Therefore, we performed two tests on the IR dataset—with the original images and with images whose grayscale was inverted so the checkerboards look like regular checkerboards. We noticed a substantial disparity between the IR and IR-inverted datasets, indicating that the performance of both OpenCV methods is adversely affected when the images are not preprocessed beforehand. Upon inverting the infrared images, we observed improvements across all methods, with the most significant enhancement occurring in the case of the OpenCV method, resulting in an eightfold increase in the amount of detected corners. Furthermore, a noteworthy observation can be made when analysing the results of the MSI on the large dataset. Surprisingly, the Geiger method outperforms the others, even though the simulated dataset results indicated that it struggled with increasing augmentations compared to the alternative methods. The application of Gaussian process enhancement further bolsters the detection of checkerboards, particularly in the case of large checkerboards (14×9). This substantial increase can be attributed to the higher chances of missing a corner in larger checkerboards. While a similar trend is observed with small checkerboards, the OpenCV method performs relatively better but still falls short of the performance achieved by the Geiger method with the proposed Gaussian process enhancement.

Table 3. Results for the different checkerboard detection methods for both the infrared (IR) images and the multispectral images (MSI). For the IR images, one set of results is without inverting the images and one with inverting the images. For the MSI images, a large and a small checkerboard are evaluated. The values in bold represent the highest scores.

	Detected Checkers (%)			
Method	**IR**	**IR Inverted**	**MSI Large CB**	**MSI Small CB**
OpenCV	12.65	**96.15**	14.23	58.4
OpenCV SB	6.44	46.15	24.21	25.6
Geiger	67.25	84.61	22.54	13.43
Geiger + GP	**95.77**	**96.15**	**77.64**	**88.8**

4.3. Use Case: Endoscopic Camera Calibration

As a final demonstration of our method, we describe the use case of working with images of calibration patterns taken to calibrate a Pillcam COLON2 double endoscope camera capsule. Several difficulties arise when calibrating this device. The resolution is low (320×320), the warping of the image is significant (172° field of view), and even the most minute artefacts in the glass casing are visible in the image. Detecting a checkerboard with the Geiger method fails, as not every corner is detected. This results in a significant portion of the corners not being used in the checkerboard. The current workaround is to add them manually, which is labour-intensive. Our method can infer the missing corners. We demonstrate this on images from the work of [28], taken from their accompanying online GitHub repository. An example can be seen in Figure 5. This image of a checkerboard is of very low quality, which resulted in several occlusions. The upper right part can no longer be fitted in a checkerboard by the Geiger method. Our Gaussian process method expands the initially-found checkerboard to include other corners, and predicts corner uv-coordinates for occluded corners as well. Moreover, the entire checkerboard is refined.



(**a**) (**b**)

Figure 5. An example of a low quality image of a checkerboard detected by (**a**) Geiger and (**b**) Geiger plus the Gaussian process enhancement.

5. Discussion

In this research, we are working with the mean of the posterior distribution of the Gaussian processes (the predicted uv-coordinates in an image for a given local xy-coordinate on a checkerboard); this tends to smooth out the results [24]. A squared exponential kernel is implemented, resulting in infinitely differentiable, and thus smooth, functions. The overall result is a checkerboard that has been smoothed out. This can be seen as removing noise or jitter on the found corners. We exploit this feature as an extra final step of corner refinement. However, caution is advised when working with heavily warped checkerboards. When

working with GPs, one could overly smooth out the corners in cases of heavily warped checkerboards or images from fisheye lenses. This could be addressed with more complex kernels or even deep Gaussian processes that will be investigated in future work.

Another way to look at this is through the insight that a GP will adjust the coordinates for a single corner based on the coordinates of all other corners. In a GP, all data points—in our case corners—are assumed to be jointly Gaussian [24]. This means they are not independent of each other—they co-vary. Other refinement methods are only based the values of neighbouring pixels, not the whole (possibly warped) grid. The justification for this is the fact that the underlying truth is a regularly spaced rectangular pattern.

The GP enhancement method can be used to interpolate and extrapolate point locations in between the corners and even outside the checkerboard, assuming the curvature pattern can be described by the points in the board. This allows us to unwarp and frontalise the image. We simply predict the corresponding pixel uv-coordinate for a dense grid of xy-coordinates. A demonstration can be seen in Figure 6.

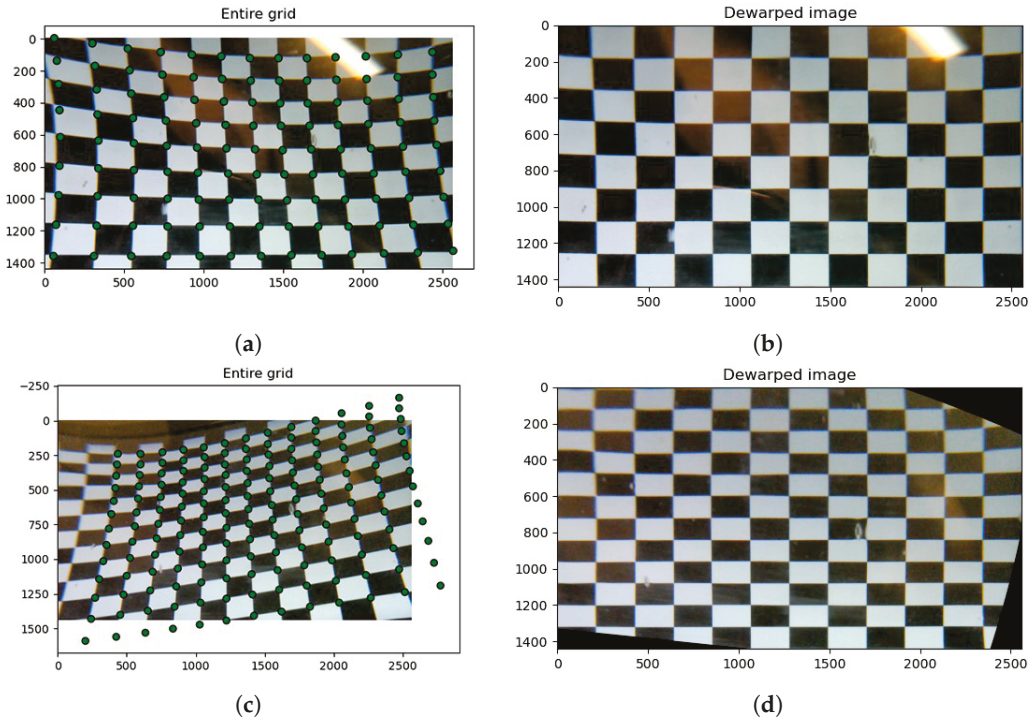


Figure 6. (**a**) A Gaussian process predicted the (green) corners of a checkerboard. Notice how the occlusions from the lighting are also filled in. (**b**) This model can be used to dewarp and frontalise an image. (**c**) A checkerboard with points beyond the image borders. (**d**) The dewarped and frontalised result, leaving pixels outside the original image black.

In future research, we will explore the possibility of working with fiducial markers [29] to assert the orientation of the board. As our current method is able to predict corners and, in fact, rows and columns that are not even on the image, the addition of fiducials becomes necessary to decide to expand the board in the image to the left, right, up, or downward directions.

Even though Gaussian processes can perform well in low data regimes [24], some numerical instabilities and oddities might occur. For instance, when working with a slightly warped 3×3 checkerboard, there is not much information upon which to build a solid statistical model. The danger is that, from a Bayesian point of view, the corners could be explained by considering their coordinates as pure noise [30]. This results in patterns for the predicted checkerboard corners that simply do not make sense. In this case, our software (version 1.2.2) allows for end-users to incorporate prior knowledge in the model by adjusting some of the options and hyperparameters in the software. In the code itself,

we provide a lot of comments to guide the end-user. We suggest two things to try: First, put a lower limit on the length scales. This will automatically force the model to consider configurations in which the points are on a smooth grid, which in fact they are. Second, consider an upper level on the noise. This will force the model to stay close to the values of the detected corners. All of this is included in comments in the code.

6. Conclusions

In this paper, we designed a new checkerboard detection pipeline, consisting of a checkerboard detector and a checkerboard enhancement with Gaussian processes. This research has shown that our Gaussian process enhancement is able to improve checkerboard detection results. By learning a mapping from local board coordinates to image pixel coordinates via a Gaussian process, we can fill in occluded corners, expand the board beyond the image borders, allocate detected corners that do not fit an initial grid, and remove noise on the detected corner's locations. We explained the role Gaussian processes play in enhancing the results. Lastly, we provide all our code as open source, in the form of a modular and easy-to-use Python checkerboard detection library called PyCBD.

The findings will be of interest to people working on camera calibration, camera localisation, SLAM, stereoscopic vision, depth sensing, and many others.

Author Contributions: Conceptualization, I.D.B.; software, I.D.B., M.H., S.S. and E.C.D.L.H.; validation, M.H. and J.G.; formal analysis, T.D.K.; investigation, I.D.B., M.H. and T.D.K.; resources, G.S., S.V. and R.P.; data curation, T.D.K.; writing—original draft preparation, I.D.B., M.H. and T.D.K.; writing—review and editing, S.S., S.V. and R.P.; visualization, J.G.; Project administration, M.H.; supervision, G.S., S.V. and R.P.; funding acquisition, G.S., S.V. and R.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by ETF Fund—PhairywinD project by Belgium SPF Economy and the University of Antwerp BOF [FFB200259, 42339, 39928].

Data Availability Statement: The code and datasets used in this article will be made available on a public GitHub repository (https://github.com/InViLabUAntwerp/PyCBD, accessed on 30 October 2023) and distributed through a PyPi package under the name PyCBD.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SLAM	Simultaneous Localisation and Mapping
CNN	Convolutional Neural Network
GP	Gaussian Process
MSI	Multispectral Imaging
IR	Infrared

References

1. Lochman, Y.; Liepieshov, K.; Chen, J.; Perdoch, M.; Zach, C.; Pritts, J. BabelCalib: A Universal Approach to Calibrating Central Cameras. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 15253–15262.
2. Chen, B.; Liu, Y.; Xiong, C. Automatic checkerboard detection for robust camera calibration. In Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, 5–9 July 2021; pp. 1–6.
3. Zhang, Y.; Zhao, X.; Qian, D. Learning-Based Distortion Correction and Feature Detection for High Precision and Robust Camera Calibration. *IEEE Robot. Autom. Lett.* **2022**, *7*, 10470–10477. [CrossRef]
4. Kang, J.; Yoon, H.; Lee, S.; Lee, S. Sparse Checkerboard Corner Detection from Global Perspective. In Proceedings of the 2021 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuala, Terengganu, 13–15 September 2021; pp. 12–17. [CrossRef]
5. Sels, S.; Ribbens, B.; Vanlanduit, S.; Penne, R. Camera Calibration Using Gray Code. *Sensors* **2019**, *19*, 246. [CrossRef]
6. Albarelli, A.; Rodolà, E.; Torsello, A. Robust Camera Calibration using Inaccurate Targets. In Proceedings of the British Machine Vision Conference, Aberystwyth, UK, 30 August–2 September 2010; BMVA Press: Durham, UK, 2010; pp. 16.1–16.10. . [CrossRef]

7. Sun, W.; Yang, X.; Xiao, S.; Hu, W. Robust Checkerboard Recognition for Efficient Nonplanar Geometry Registration in Projector-Camera Systems. In Proceedings of the 5th ACM/IEEE International Workshop on Projector Camera Systems, Marina del Rey, CA, USA, 10 August 2008; Association for Computing Machinery: New York, NY, USA, 2008; PROCAMS '08, pp. 1–7. [CrossRef]
8. Juarez-Salazar, R.; Diaz-Ramirez, V.H. Flexible camera-projector calibration using superposed color checkerboards. *Opt. Lasers Eng.* **2019**, *120*, 59–65. [CrossRef]
9. Ito, A.; Li, J.; Maeda, Y. SLAM-Integrated Kinematic Calibration Using Checkerboard Patterns. In Proceedings of the 2020 IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA, 12–15 January 2020; pp. 551–556.
10. Enebuse, I.; Foo, M.; Ibrahim, B.S.K.K.; Ahmed, H.; Supmak, F.; Eyobu, O.S. A Comparative Review of Hand-Eye Calibration Techniques for Vision Guided Robots. *IEEE Access* **2021**, *9*, 113143–113155. [CrossRef]
11. Gui, Y.; Wu, Y.; Wang, Y.; Yao, C. Visual Image Processing of Humanoid Go Game Robot Based on OPENCV. In Proceedings of the 2020 Chinese Control And Decision Conference (CCDC), Hefei, China, 22–24 August 2020; pp. 3713–3716. [CrossRef]
12. Vaníček, O.; Chaluš, M.; Liška, J. 3D Vision Based Calibration Approach for Robotic Laser Surfacing Applications. In Proceedings of the 2022 20th International Conference on Mechatronics-Mechatronika (ME), Pilsen, Czech Republic, 7–9 December 2022; pp. 1–6.
13. Geiger, A.; Moosmann, F.; Car, O.; Schuster, B. Automatic camera and range sensor calibration using a single shot. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, St. Paul, MN, USA, 14–18 May 2012; pp. 3936–3943. [CrossRef]
14. Kaehler, A. *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*; O'Reilly Media: Sebastopol, CA, USA, 2016.
15. Vezhnevets, V. OpenCV Calibration Object Detection. 2005. Available online: https://www.graphicon.ru/oldgr/en/research/calibration/opencv.html (accessed on 3 March 2023).
16. Rufli, M.; Scaramuzza, D.; Siegwart, R. Automatic detection of checkerboards on blurred and distorted images. In Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, Nice, France, 22–26 September 2008; pp. 3121–3126. [CrossRef]
17. Duda, A.; Frese, U. Accurate Detection and Localization of Checkerboard Corners for Calibration. In Proceedings of the British Machine Vision Conference, Newcastle upon Tyne, UK, 3–6 September 2018; pp. 1–10.
18. Fuersattel, P.; Dotenco, S.; Placht, S.; Balda, M.; Maier, A.; Riess, C. OCPAD—Occluded checkerboard pattern detector. In Proceedings of the 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, USA, 7–10 March 2016; pp. 1–9. [CrossRef]
19. Placht, S.; Fürsattel, P.; Mengue, E.A.; Hofmann, H.; Schaller, C.; Balda, M.; Angelopoulou, E. ROCHADE: Robust Checkerboard Advanced Detection for Camera Calibration. In Proceedings of the Computer Vision—ECCV 2014, Zurich, Switzerland, 6–12 September 2014; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Spring: Cham, Switzerland, 2014; pp. 766–779.
20. Scaramuzza, D.; Martinelli, A.; Siegwart, R. A Toolbox for Easily Calibrating Omnidirectional Cameras. In Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 9–13 October 2006; pp. 5695–5701. [CrossRef]
21. Chen, B.; Xiong, C.; Zhang, Q. CCDN: Checkerboard Corner Detection Network for Robust Camera Calibration. In Proceedings of the Intelligent Robotics and Applications, Newcastle, NSW, Australia, 9–11 August 2018; Proceedings, Part I; Chen, Z., Mendes, A., Yan, Y., Chen, S., Eds.; Spring: Cham, Switzerland, 2018; pp. 324–334.
22. Wang, G.; Zheng, H.; Zhang, X. A Robust Checkerboard Corner Detection Method for Camera Calibration Based on Improved YOLOX. *Front. Phys.* **2022**, *9*, 819019. [CrossRef]
23. Wu, H.; Wan, Y. A highly accurate and robust deep checkerboard corner detector. *Electron. Lett.* **2021**, *57*, 317–320. [CrossRef]
24. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; The MIT Press: Cambridge, MA, USA, 2006.
25. Duvenaud, D.K.; College, P. Automatic Model Construction with Gaussian Processes. Ph.D. Thesis, University of Cambridge, Cambridge, UK, 2014.
26. Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and Flexible Image Augmentations. *Information* **2020**, *11*, 125. [CrossRef]
27. ftdlyc. Libcbdetect: A Library for Chessboard Detection. Available online: https://github.com/ftdlyc/libcbdetect (accessed on 2 March 2023).
28. Ozyoruk, K.B.; Gokceler, G.I.; Bobrow, T.L.; Coskun, G.; Incetan, K.; Almalioglu, Y.; Mahmood, F.; Curto, E.; Perdigoto, L.; Oliveira, M.; et al. EndoSLAM dataset and an unsupervised monocular visual odometry and depth estimation approach for endoscopic videos. *Med. Image Anal.* **2021**, *71*, 102058. [CrossRef] [PubMed]
29. Kalaitzakis, M.; Cain, B.; Carroll, S.; Ambrosi, A.; Whitehead, C.; Vitzilaios, N. Fiducial markers for pose estimation: Overview, applications and experimental comparison of the artag, apriltag, aruco and stag markers. *J. Intell. Robot. Syst.* **2021**, *101*, 71. [CrossRef]
30. Lázaro-Gredilla, M. Bayesian warped Gaussian processes. In Proceedings of the 25th International Conference on Neural Information Processing Systems—Volume 1, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1619–1627.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Beyond Traditional Assessment: A Fuzzy Logic-Infused Hybrid Approach to Equitable Proficiency Evaluation via Online Practice Tests

Todorka Glushkova [1], Vanya Ivanova [2,*] and Boyan Zlatanov [3]

[1] Department of Computer Technology, Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 4000 Plovdiv, Bulgaria; glushkova@uni-plovdiv.bg
[2] Department of Computer Systems, Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 4000 Plovdiv, Bulgaria
[3] Department of Mathematical Analysis, Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 4000 Plovdiv, Bulgaria; bobbyz@uni-plovdiv.bg
* Correspondence: vantod@uni-plovdiv.bg; Tel.: +359-89-989-2044

Abstract: This article presents a hybrid approach to assessing students' foreign language proficiency in a cyber–physical educational environment. It focuses on the advantages of the integrated assessment of student knowledge by considering the impact of automatic assessment, learners' independent work, and their achievements to date. An assessment approach is described using the mathematical theory of fuzzy functions, which are employed to ensure the fair evaluation of students. The largest possible number of students whose reevaluation of test results will not affect the overall performance of the student group is automatically determined. The study also models the assessment process in the cyber–physical educational environment through the formal semantics of calculus of context-aware ambients (CCAs).

Keywords: cyber–physical educational space; testing and evaluation of students' knowledge; fuzzy functions; calculus of context-aware ambients (CCAs)

MSC: 94-10

1. Introduction

Student assessment is an important aspect of the educational process that aims to measure the knowledge, skills, and achievements of students in the course of their studies. The development of modern education, along with traditional learning, also requires the use of new assessment models, which must provide flexibility in terms of the ways and forms of education, personalization, and the possibility of taking into account the individual characteristics of each student. The problem is becoming more relevant also in terms of reducing the degree of formal education as compared to hybrid forms and the use of virtual educational platforms [1].

Cyber–physical educational environments provide integration of virtual and physical components in the educational process. This model of education is used in various contexts, including in educational institutions, vocational training, distance learning, and for nonformal, lifelong learning. Motivated by the advantages that these environments provide, teams from the Faculty of Mathematics and Informatics (FMI) of the University of Plovdiv are developing prototypes of cyber–physical educational platforms intended to train both university students and nonlearners [2]. Assessment is a key feature of any such platform, and the difficulties are related to the hybrid nature of learning—face-to-face, distance learning, self-study, project-based, etc.

This article presents a model for assessing students' knowledge in a cyber–physical educational environment by using the mathematical theory of fuzzy sets. Modeling of

processes and communication between intelligent components is realized through the formal semantics of the calculus of context-aware ambients (CCAs). To test the proposed model, data were used from the assessment of first-year students from FMI of the University of Plovdiv in their English language course.

An exceptional aspect of our work lies in the assessment approach utilizing fuzzy functions. Notably, our paper introduces a novel methodology by using a cyber–physical educational platform to automatically determine the largest feasible number of students whose work can undergo reevaluation through the application of fuzzy functions. This allows for the adjustment of individual grades—either upward or downward—without impacting the overall result of the entire student cohort. In other words, the average score of the entire group of students remains unaltered despite these individual adjustments. This innovative feature sets our work apart and significantly contributes to the existing body of knowledge in the field.

The structure of this article is as follows: In the second section, the motivation and research related to the topic are discussed. The third part considers the possibilities of applying the theory of fuzzy sets to evaluate students and the CCA modeling of the process of assessment, and the fourth part comments on the findings of the research. The conclusion summarizes the results and points out perspectives for future studies.

2. Motivation and Related Works

Based on the definition of the National Science Foundation (NSF) [3], cyber–physical systems (CPSs) provide integration between computational, network, and physical processes. Embedded digital components and networks monitor and control physical processes, supplying continuous feedback and control. CPSs integrate the dynamics of physical processes with those of software and networks, providing opportunities for modeling, design, and analysis techniques. These systems must ensure the dynamic interaction with objects from the physical and virtual worlds, which requires the use of autonomous intelligent components. Thus, cyber–physical systems move into cyber–physical spaces, in which users are placed at the center of social interactions. In this sense, the cyber–physical–social space (CPSS) is a fusion of the physical space, cyberspace, and the social space. The evolution from the CPS to the CPSS [4] is a long process that involves solving challenges and problems of a different nature. These spaces have the potential to be adapted in all spheres of the modern world—in a "smart city", agriculture, animal husbandry, transport, medicine, tourism, and, of course, education.

The creation of educational cyber–physical platforms is a challenge but also a necessity [5]. In these spaces, continuous blended (face-to-face and virtual) learning is possible, tailored to the personal characteristics of each student. An important component of these environments is the development of an appropriate test platform that provides opportunities for assessment and self-assessment of learners [6].

The Virtual–Physical Space (ViPS) is being developed by a team of the Distributed e-Learning Center (DeLC) laboratory at the University of Plovdiv as a reference IoT-based architecture that can be adapted to different CPSS applications in different application areas. In the field of higher education, the Virtual Education Space (VES) platform is being developed, which builds on the DeLC educational platform. DeLC has been actively used in university education since 2004 and provides e-learning services and SCORM-based learning content [7]. Some of the strengths of the testing platform in DeLC are that it is user-friendly and employs metadata, which allows tests to be compiled based on different criteria, enables the use of photos and other visual components, provides an option to preview the test before being administered to students, and offers the option to set a time limit and different validity periods for each test.

An alternative platform used for e-tests at the Faculty of Mathematics and Informatics is the Distributed Platform for e-Learning (DisPeL), whose key elements are administration of the learning process and adaptability of the learning content [8]. DisPeL enhances the educational process by providing the following electronic services:

- Automation of the administrative process;
- Maintaining an adaptive learning process;
- Online review of student progress and assessment;
- Supporting conventional testing and evaluation.

Due to its advantages and characteristics, the DisPeL test system has been successfully implemented in several universities in Bulgaria. By choosing the appropriate platform, the cyber–physical space allows for the automatic processing of the learners' scores, and through fuzzy sets, corrections can be made, leading to a fairer assessment. We have further illustrated that there is a limit on the number of students being reevaluated, which ensures that after their grades are corrected, the mean and the variance of the group does not change statistically significantly. If this set of revised scores is increased, not only does a different distribution result, but the corrected scores may not correspond to the actual knowledge of the learners.

3. Materials and Methods

The study was carried out in the context of study at FMI of the University of Plovdiv. All students study English as a foreign language for 160 or 200 academic hours in the first academic year depending on their major. Learners have seminars in English once a week, and they are taught in groups of about 20 learners of a similar level of knowledge and skills in accordance with the Common European Framework of Reference for Languages (CEFR) [9]. Since grading is done through continuous assessment, besides the placement test, students at FMI sit for a midterm and a final test. Their final grades in English are formed at the end of the course on a six-point scale (2 is the lowest grade and corresponds to fail, and 6 is the highest grade and corresponds to excellent).

3.1. Application of the Fuzzy Set Theory in Student Assessment

The standard evaluation of online practice tests at FMI of the University of Plovdiv is conducted by using the following grading system:

- An excellent 6 grade is awarded for test scores from 87% to 100%;
- A very good 5 grade from 75% to 86%;
- A good 4 grade from 62% to 74%;
- A satisfactory 3 grade from 50% to 61%;
- A poor 2 grade for 49% and below.

However, the fairness of marginal, "borderline" scores that determine whether a learner is assigned the higher or the lower grade can be viewed as questionable. It can be considered unjust for a learner to pass a test with 50 points out of 100, and for another one to fail the same exam with only a single point less than him/her. In an attempt to make students' boundary grades more equitable, we have employed a fuzzy-set technique to change the grades of the borderline cases in two examination tests. Similar efforts have been described in [10,11], where fuzzy logic and fuzzy functions were applied to estimate learners' tests with a view to allocate them a more objective grade.

The first exam was considered to be final test taken by 78 students at the end of the language course. It consisted of 60 closed questions and one open, with a maximum score of 80 points, 20 of which were awarded for the open question. When test questions measure similar capability or expertise, they yield a large inner consistency reliability. If a test comprises various types of questions evaluating different kinds of capabilities and knowledge, Cronbach's coefficient tends to be smaller as a consequence of the dissimilarity of the questions in the context of layout and content. For this reason, initially, we needed to decide what weight to assign the open question to guarantee the reliability of the test; an erroneous choice can often prove discriminatory. The alpha coefficient of Cronbach can effortlessly be estimated by means of the following formula:

$$\alpha = \frac{k}{k-1}\left(1 - \frac{\textit{Sum of Item Variances}}{\textit{Test Variances}}\right) \quad (1)$$

The evaluated test item is $k = 61$ with sum of variances 50.91, and the test variance is 179.45. Inserting these numbers in (1), we acquire $\alpha = 0.72$, which means that the reliability is simply acceptable. The largest variance is present in the open question; consequently, we search for $\gamma > 0$ to multiply the scores of the open-ended question for α to become larger. If we choose $\gamma = 0.2$, we obtain the sum of the variances 12.92, and the test variance is 114.99. Implementing the new numbers in (1), we obtain $\alpha = 0.902$; therefore, the reliability is excellent.

Thus, the maximum score that a learner can receive is 64 (60 for the closed questions + 20×0.2 for the open one); i.e., the maximum points for the closed questions are 60, and the maximum score for the open one is 4. With the standard grading of online practice tests at FMI, the results should be interpreted as follows: an excellent 6 grade is from 56 to 64 points, a very good 5 grade is from 48 to 55, a good 4 grade is from 40 to 47, a satisfactory 3 grade is from 32 to 39, and a poor 2 grade is 31 points and below.

The second test is a midterm test, taken approximately in the middle of the language course. It was administered to 36 students altogether. The test consists of 70 closed questions, with 1 point awarded for a correct answer, and 3 open questions, with a maximum number of 6 for each. Thus, the overall maximum test points are 88, and the highest possible score for the open items is 18. The Cronbach's alpha was 0.82, meaning that the reliability of the test is good. There were no great differences in the variances of the different types of questions, and thus searching for a scaling coefficient to increase Cronbach's alpha coefficient, as was done for the first test, is not required. At first glance, the results of the second test appear to be worse when compared with the first one because not a single test-taker obtained the maximum number of points. Solely to simplify the calculations and the notations and in order to use one and the same function in the fuzzy sets technique, we scaled the results of the second test. Thus, we scaled the points to ensure that the maximum score obtained by a student in the second test would represent the maximum points of the test; i.e. the highest number of points received by the students was 80; therefore, we scaled the results of all students with the factor $64/80$. Since the scores from the open question(s) are used for fuzzifying some of the test results, we scaled the 18 points from the results from the open questions in the second test by a factor of $20/18$ in order to be able to use the same functions in the calculations.

Fuzzy logic, fuzzy sets, and fuzzy functions have been widely used since first being introduced by Zadeh [11]. We would like to emphasize only several works that have a connection with our investigation in e-learning and e-testing: [12–15].

A classical technique for reevaluations of test results with fuzzy sets is considering some borderline grades that need to be reassessed [12,13]. However, we present a different approach, in which we search for a maximum number of possible borderline grades to be fuzzified without changing the statistical distribution of the overall grade. We considered five functions that stand for the fuzzy membership ones to the sets of marks. We would like to mention that in very recent years there has been a large increase in the usage of fuzzy logic in the evaluation of students' performance [16–26].

Let $f : \mathbb{R}^3 \to \mathbb{R}$ and $g : \mathbb{R}^3 \to \mathbb{R}$ be two functions. We specify five functions, which denote the bell-like functions of fuzzy membership to the sets of marks, as follows:

$$\mu_{Poor} = \begin{cases} 1 & x < x_1 \\ f(x_1, x_2, x) & x_1 \le x < x_2 \\ 0 & x_2 \le x \le x_9 \end{cases} \quad \mu_{Satis.} = \begin{cases} 0 & x_0 \le x < x_1 \\ g(x_1, x_2, x) & x_1 \le x < x_2 \\ 1 & x_2 \le x < x_3 \\ f(x_3, x_4, x) & x_3 \le x < x_4 \\ 0 & x_4 \le x \le x_9 \end{cases}$$

$$\mu_{Good} = \begin{cases} 0 & x_0 \le x < x_3 \\ g(x_3, x_4, x) & x_3 \le x < x_4 \\ 1 & x_4 \le x < x_5 \\ f(x_5, x_6, x) & x_5 \le x < x_6 \\ 0 & x_6 \le x \le x_9 \end{cases} \quad \mu_{VeryGood} = \begin{cases} 0 & x_0 \le x < x_5 \\ g(x_5, x_6, x) & x_5 \le x < x_6 \\ 1 & x_6 \le x < x_7 \\ f(x_7, x_8, x) & x_7 \le x < x_8 \\ 0 & x_8 \le x \le x_9 \end{cases}$$

$$\mu_{Excellent} = \begin{cases} 0 & x_0 \leq x < x_7 \\ g(x_7, x_8, x) & x_7 \leq x < x_8 \\ 0 & x_8 \leq x \leq x_9 \end{cases}$$

We can consider the following functions:

$$f(a,b,x) = \frac{\cos\left(\frac{a\pi}{a-b} - \frac{\pi x}{a-b}\right)}{2} + \frac{1}{2}, \;\; g(a,b,x) = \frac{\cos\left(\frac{(2a-b)\pi}{a-b} - \frac{\pi x}{a-b}\right)}{2} + \frac{1}{2}$$

from which we obtain a bell–shaped fuzzy function μ_{Poor}, $\mu_{Satis.}$, μ_{Good}, $\mu_{VeryGood}$, $\mu_{Excellent}$, denoted by μ_P, μ_S, μ_G, μ_V, μ_E, respectively.

3.2. The Test Construction

The test consists of several sections, each one aimed to check specific knowledge.

Criterion I (reproduction of information) is the lowest level in the cognitive domain; hence, multiple choice questions (MCQ) are employed most frequently to determine whether students remember correctly the form of certain expressions. The following is an example:

He kept explaining his point of view until he was ..... in the face but the inspectors were not impressed

(a) *red*
(b) *blue*
(c) *black*
(d) *pink.*

The test questions related to criterion II usually incorporate multiple choice, true/false, or closed questions to match words or expressions with their definitions, synonyms, and antonyms. The following is an example:

Choose the word which is a SYNONYM (a word with a similar meaning) to the capitalized word in the sentence:

The company was FOUNDED by two partners.

(a) *set up*
(b) *discovered*
(c) *reformed*
(d) *destroyed*

The test questions relating to criterion III (detection of errors in various contexts) are usually multiple choice ones to choose the part of the sentence which contains a spelling or grammar mistake and true or false to determine whether the sentences are free of lexical or grammatical errors or not. The following is an example:

Choose the part of the sentence which contains a spelling or grammar mistake.
The money doesn't smell.

(a) *the*
(b) *money*
(c) *doesn't*
(d) *smell*

Criterion IV (analysis of the lexical and grammatical items of a sentence) most often comprises the MCQ to select a sentence which most accurately explains the meaning of another one, choosing the correct grammatical form of a verb, or short-answer test items to write the most appropriate word(s)/ expression(s) or a grammatical form in a sentence. For instance,

Select the sentence which most accurately explains the meaning of the given one.
He was considered a nut and was ridiculed for standing out.

(a) *He was believed to be crazy so everyone mocked him because he wasn't sitting down like the others.*

(b) *The others made fun of him because he liked nuts more than anything else.*
(c) *He wasn't like the others so people thought he was out of his mind and praised him.*
(d) *People thought he was a lunatic and laughed at him because he was different.*

Finally, the test questions with reference to criterion V (text creation) usually instruct students to compose a text with a limited number of words to demonstrate specific knowledge or skills such as writing an email or a review of a product that they have bought, to explain and illustrate the meaning and use of idioms, etc. An example would be the following task:

WRITING: Your friend Kate had asked you to look after her cat while she was abroad but the cat disappeared. Write a short email (50–70 words) to her in which you:

- Include a greeting;
- Apologize and say that you lost her cat;
- Explain how exactly it happened;
- Say what you have done about it.
- Close your email.

3.3. Illustration of the Fuzzy Logic Usage in Recalculating Students' Marks

We illustrate the functions defined above in the case when $x_0 = 0$, $x_1 = 29.5$, $x_2 = 34.5$, $x_3 = 37.5$, $x_4 = 42.5$, $x_5 = 45.5$, $x_6 = 50.5$, $x_7 = 53.5$, $x_8 = 58.5$, $x_9 = 64$ (Figure 1).

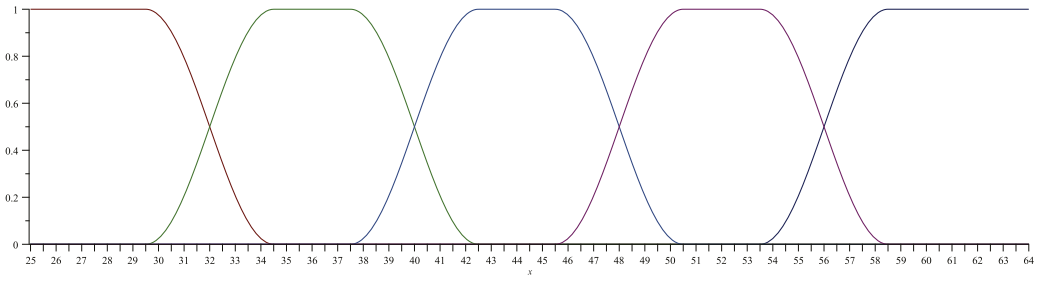


Figure 1. Plots of μ_{Poor}, $\mu_{Satis.}$, μ_{Good}, μ_{VGood} and $\mu_{Excellent}$.

Therefore, a student with 41 overall points belongs to the set of satisfactory grades with a degree 0.21; to the class of good grades with a degree 0.79; and to the other sets of poor, very good, or excellent with a degree of 0. A student with an overall score of 44 points belongs to the set of good grades with a degree 1 and to the other sets of poor, satisfactory, very good, or excellent with a degree 0.

In case the points a learner has received in a test do not belong definitively to a given set, we need to choose a different criterion, which depends on the learner's result, in order to decide which grade to assign him/her, and that will be the learner's result on the open items. In addition, we once again divide the learners' marks into 5 groups: poor (from y_0 to y_1 points), satisfactory (from y_2 to y_3), good (from y_4 to y_5), very good (from y_6 to y_7), and excellent (from y_8 to y_9), and we also define their membership functions μ, which we denote by ν_P, ν_S, ν_G, ν_V, and ν_E

We illustrate the defined above functions in the case $y_0 = 0$, $y_1 = 6.9$, $y_2 = 9.1$, $y_3 = 10.4$, $y_4 = 12.6$, $y_5 = 13.4$, $y_6 = 15.6$, $y_7 = 16.4$, $y_8 = 18.6$, $y_9 = 20$ (Figure 2).

The norms for executing set procedures of union (OR), intersection (AND), and complement (NOT) that concern us the most are given below.

For onion, we look at the degree of membership for each set and pick the lower one of the two as follows: $\mu_{A\cup B} = \min(\mu_A, \mu_B)$ (Figure 3).

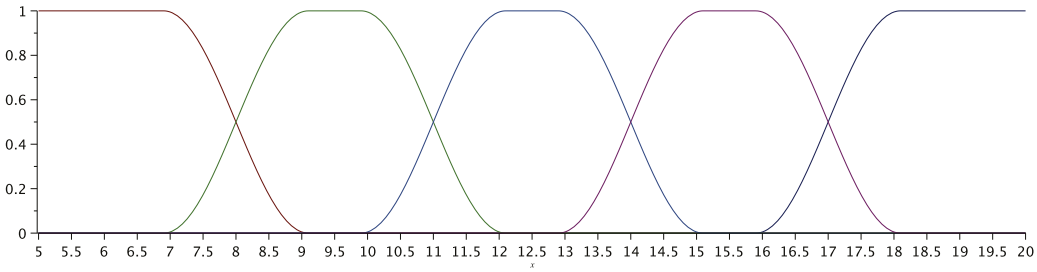


Figure 2. Plots of ν_{Poor}, $\nu_{Satis.}$, ν_{Good}, ν_{VGood} and $\nu_{Excellent}$.

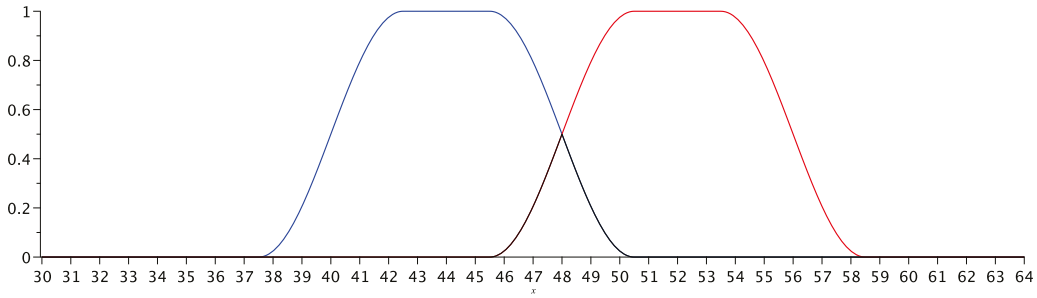


Figure 3. Plots of μ_A, μ_B and $\mu_{A\cap B} = \min(\mu_A, \mu_B)$.

For intersection, we inspect the degree of membership for each set and choose the larger of the two as follows: $\mu_{A\cap B} = \max(\mu_A, \mu_B)$ (Figure 4).

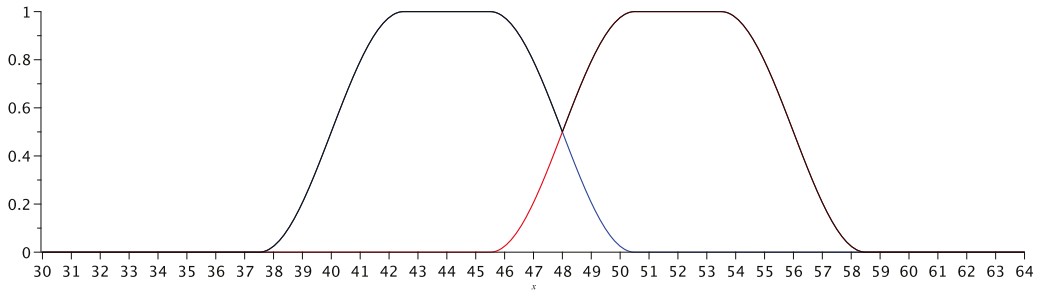


Figure 4. Plots of intersection $\mu_{A\cup B} = \max(\mu_A, \mu_B)$.

The fuzzy associative matrix (Table 1) provides an appropriate manner to immediately integrate the input relations in order to get the fuzzified output results [12] or [10]. The input values for the scores of the open-ended items are at the upper section of the matrix, and the input values for the total results of the test are down left section in the matrix. We have used the conventional Bulgarian grading scale.

Table 1. Fuzzy associative matrix.

	ν_{Poor}	$\nu_{Satisf.}$	ν_{Good}	$\nu_{VeryGood}$	$\nu_{Excellent}$
μ_{Poor}	2	2	3	3	4
$\mu_{Satisf.}$	2	3	3	4	4
μ_{Good}	2	3	4	5	5
$\mu_{VeryGood}$	3	4	5	5	6
$\mu_{Excellent}$	3	4	5	6	6

Let us review a learner with a total result of 49 points and a mark on the open-ended item of 19 points. He or she belongs to the set of very good marks with a degree of $\mu_{VeryGood}(49) = 0.79$ and to the set of good grades with a degree of $\mu_{Excellent}(49) = 0.21$. Normally, he/she will be assessed as very good (5). Nevertheless, the crossing of the

two marks—the total points together with the points of the open question—denotes the following: according to the matrix in Table 1, he or she belongs to the set $\mu_{VeryGood} \cap \nu_{Excellent}$ with 0.79 degree, to the set $\mu_{Good} \cap \nu_{Excellent}$ with 0.21 degree, to the set $\mu_{VeryGood} \cap \nu_{VeryGood}$ with 0.005 degree, and to the set $\mu_{Good} \cap \nu_{VeryGood}$ with 0.005 degree. Consequently, we can assign him or her excellent (6).

In accordance with [12] and [10], we have to recalculate the mark for each learner, whose test score does not belong definitely to a given set. For this purpose, we can consider the table, in which the function F returns the minimums of μ and ν. The highest membership grade that is obtained from the table stands for the corrected mark from the matrix (Table 2).

Table 2. The different combinations of the minimums of μ and ν, computed by means of F.

$F(\mu_P(p),\nu_P(q))$	$F(\mu_P(p),\nu_S(q))$	$F(\mu_P(p),\nu_G(q))$	$F(\mu_P(p),\nu_{VG}(q))$	$F(\mu_P(p),\nu_E(q))$
$F(\mu_S(p),\nu_P(q))$	$F(\mu_S(p),\nu_S(q))$	$F(\mu_S(p),\nu_G(q))$	$F(\mu_S(p),\nu_{VG}(q))$	$F(\mu_S(p),\nu_E(q))$
$F(\mu_G(p),\nu_P(q))$	$F(\mu_G(p),\nu_S(q))$	$F(\mu_G(p),\nu_G(q))$	$F(\mu_G(p),\nu_{VG}(q))$	$F(\mu_G(p),\nu_E(q))$
$F(\mu_V(p),\nu_P(q))$	$F(\mu_V(p),\nu_S(q))$	$F(\mu_V(p),\nu_G(q))$	$F(\mu_V(p),\nu_{VG}(q))$	$F(\mu_V(p),\nu_E(q))$
$F(\mu_E(p),\nu_P(q))$	$F(\mu_E(p),\nu_S(q))$	$F(\mu_E(p),\nu_G(q))$	$F(\mu_E(p),\nu_{VG}(q))$	$F(\mu_E(p),\nu_E(q))$

Now, by way of illustration, let a student have a total test score of 57 points and open question result of 18 points ($p = 57$ and $q = 18$). As shown in Table 3, after the fuzzification, the student will be marked with excellent (6), which coincides with the traditional evaluation.

Table 3. The fuzzified learner's mark when $p = 57$, $q = 18$.

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0.005	0.21
0	0	0	0.005	0.79

If we analyze another learner that has received 42 points (which corresponds to good (4) in the traditional scoring system) and 17 points for the open-ended item, it is seen from Table 4 that after the correction, this particular learner should get a higher mark, namely very good (5).

Table 4. The fuzzified learner's mark when $p = 42$, $q = 17$.

0	0	0	0	0
0	0	0	0.02	0.02
0	0	0	0.5	0.49
0	0	0	0	0
0	0	0	0	0

To recalculate the test results, we only need to input the test scores, and MapleSoft 2016.0 automatically chooses the scores to be fuzzified and calculates the fuzzified grades.

We define two parameters $\varepsilon > 0$ and $\delta > 0$ and we specify the following: $x_0 = 0$, $x_1 = 32 - \varepsilon$, $x_2 = 32 + \varepsilon$, $x_3 = 40 - \varepsilon$, $x_4 = 40 + \varepsilon$, $x_5 = 48 - \varepsilon$, $x_6 = 48 + \varepsilon$, $x_7 = 56 - \varepsilon$, $x_8 = 56 + \varepsilon$, $x_9 = 64$, $y_0 = 0$, $y_1 = 8 - \delta$, $y_2 = 8 + \delta$, $y_3 = 11 - \delta$, $y_4 = 11 + \delta$, $y_5 = 14 - \delta$, $y_6 = 14 + \delta$, $y_7 = 17 - \delta$, $y_8 = 17 + \delta$, $y_9 = 20$.

As a result, Maple indicates that when $\varepsilon = 2.6$ and $\delta = 1.5$, the two distributions do not differ statistically, and $\varepsilon + \delta = 4.1$ is the largest possible sum. In this case, we have changed 46 marks. When we fuzzify the grades with $\varepsilon = 2.7$ and $\delta = 1.5$, the two distributions differ statistically. We acquire 56 marks to be changed as follows: by increasing the set of fuzzified marks, we only add new students, whose marks will be reevaluated. That is why

we have listed in Table 5 the 56 fuzzified marks (the information below is organized in this order: [classical test grade, classical open question grade], student's number in the list, test score, open question score, fuzzified grade, classical grade). We have used a bold font in Table 5 for the 10 new students that were added by increasing $\varepsilon = 2.6$ to $\varepsilon = 2.7$.

Table 5. Fuzzified students' marks.

[4, 6], 1, 42.6, 18, 5, 4	[6, 6], 2, 58.0, 20, 6, 6	**[4, 6], 3, 42.80, 19, 5, 4**
[4, 6], 4, 42.80, 19, 5, 4	[5, 6], 6, 50.6, 18, 6, 5	[5, 6], 8, 54.0, 20, 6, 5
[5, 5], 10, 49.4, 17, 5, 5	[2, 5], 11, 30.8, 14, 3, 2	**[2, 5], 12, 29.20, 16, 3, 2**
[5, 5], 13, 50.80, 14, 5, 5	**[2, 5], 14, 29.20, 16, 3, 2**	[5, 6], 15, 48.6, 18, 6, 5
[4, 2], 17, 40.6, 8, 2, 4	[4, 6], 18, 46.0, 20, 5, 4	**[5, 5], 19, 53.20, 16, 5, 5**
[6, 6], 20, 58.80, 19, 6, 6	[4, 5], 21, 46.2, 16, 4, 4	[5, 5], 24, 47.8, 14, 5, 4
[4, 6], 26, 40.6, 18, 5, 4	[3, 6], 27, 33.6, 18, 4, 3	[4, 2], 28, 41.0, 0, 2, 4
[4, 5], 29, 40.8, 14, 4, 4	[4, 2], 30, 40.4, 7, 2, 4	**[3, 3], 31, 34.80, 9, 3, 3**
[5, 2], 33, 46.6, 8, 3, 4	[5, 4], 34, 50.6, 13, 5, 5	[4, 6], 35, 42.0, 20, 5, 4
[5, 5], 39, 50.2, 16, 5, 5	[4, 6], 41, 39.6, 18, 5, 3	[2, 4], 42, 30.6, 13, 3, 2
[3, 6], 43, 32.8, 19, 4, 3	[5, 6], 46, 47.8, 19, 6, 4	[5, 2], 47, 49.6, 8, 3, 5
[3, 6], 49, 34.0, 20, 4, 3	[4, 6], 50, 40.8, 19, 5, 4	[3, 5], 51, 32.2, 16, 4, 3
[5, 5], 53, 47.2, 16, 5, 4	[2, 5], 54, 31.0, 15, 3, 2	[3, 6], 56, 39.0, 20, 4, 3
[5, 6], 57, 54.80, 19, 6, 5	[5, 6], 59, 55.0, 20, 6, 5	[6, 6], 60, 58.0, 20, 6, 6
[5, 6], 62, 50.80, 19, 6, 5	[3, 6], 63, 38.8, 19, 4, 3	[5, 2], 65, 49.0, 0, 3, 5
[4, 5], 67, 40.2, 16, 4, 4	[3, 3], 68, 34.0, 10, 3, 3	**[4, 5], 69, 42.80, 14, 4, 4**
[5, 5], 70, 48.8, 14, 5, 5	[5, 6], 71, 47.6, 18, 6, 4	[4, 6], 72, 41.6, 18, 5, 4
[2, 2], 73, 29.4, 7, 2, 2	[6, 6], 75, 57.4, 17, 6, 6	[4, 5], 76, 40.2, 16, 4, 4
[5, 5], 77, 48.2, 16, 5, 5	[5, 2], 78, 47.2, 6, 3, 4	

At any stage of the calculations, Maple tests the standard t-test with paired samples. In the case of $\varepsilon = 2.6$ and $\delta = 1.5$, we can understand that we should accept the hypothesis that the distributions have equal means and in the case $\varepsilon = 2.7$, $\delta = 1.5$, Maple indicates that we should reject the hypothesis that the two distributions have equal means.

We will analyze the fuzzified grades in the Discussion below in order to justify that we have obtained fairer marks in the first case and less equitable marks in the second one.

By comparing the results of second exam in terms of overall points (78, 76, 80, 63, 67, 61, 68, 75, 13, 72, 74, 76, 72, 79, 70, 58, 57, 66, 66, 59, 74, 67, 36, 35, 75, 38, 36, 52, 47, 55, 25, 47, 73, 35, 31, 70) and points on the open questions (15, 14, 15, 7, 6, 8, 9, 13, 0, 13, 13, 14, 11, 15, 15, 3, 9, 11, 8, 9, 13, 12, 0, 3, 14, 0, 0, 6, 7, 8, 0, 3, 13, 0, 0, 7) to those first exam in terms of overall points (57, 74, 58, 58, 22, 65, 19, 70, 21, 63, 42, 42, 62, 42, 63, 23, 47, 62, 66, 74, 59, 56, 69, 59, 67, 55, 48, 41, 52, 46, 42, 37, 53, 61, 58, 30, 35, 69, 63, 66, 54, 41, 48, 47, 50, 63, 56, 41, 50, 56, 45, 36, 60, 43, 69, 55, 70, 77, 71, 74, 24, 66, 54, 43, 49, 59, 53, 42, 54, 60, 62, 56, 35, 59, 71, 53, 61, 52) and points on the open questions (18, 20, 19, 19, 7, 18, 0, 20, 0, 17, 14, 16, 14, 16, 18, 10, 8, 20, 16, 19, 16, 16, 20, 14, 20, 18, 18, 0, 14, 7, 9, 0, 8, 13, 20, 9, 0, 12, 16, 18, 18, 13, 19, 15, 17, 19, 8, 15, 20, 19, 16, 0, 16, 15, 20, 20, 19, 20, 20, 20, 10, 19, 19, 0, 0, 0, 16, 10, 14, 14, 18, 18, 7, 19, 17, 16, 16, 6), we can see that the results from the second test f0r the open questions are much lower than those from the first.

When we fuzzify the grades with $\varepsilon = 2.4$ and $\delta = 1.2$, the two distributions differ statistically. We obtain 18 grades to be modified, without changing the distributions of the overall marks before and after the fuzzification.

3.4. CCA Modeling of the Assessment Process in a Cyber–Physical Educational Environment

Ambient-oriented modeling (AOM) is a type of computational process, in the context of which interactions between objects from the physical and the virtual worlds play a major role. The calculus of context-aware ambient (CCA) formalism models the system's ability to respond to changes in the surrounding space [27]. A CCA is an identity that is used to describe an object or a component—a process, device, location, etc. Each ambient has a name and boundaries and can contain other ambients within itself and can be included in another ambient. There are three possible relationships between any two ambients—

parent, child, and relative. Each ambient can communicate with the ambients around it, and ambients can exchange messages with each other. The process of exchanging messages is done using the handshaking process. In the notation, "::" is a symbol for relative ambients; "↑" and "↓" are parent and child symbols, respectively; "<>" means sending; and "()" means receiving a message. An ambient can be mobile; i.e., it can move within its surroundings. With $CCAs$, there are two movement options, in and out, which allow ambients to move from one location to another. In $CCAs$, four syntactic categories can be distinguished:

- Processes P;
- Capabilities M;
- Locations α;
- Context expressions k.

As we have already pointed out, the concept of ambients is an abstraction of the limited space where some computation is performed. Ambients are mobile and can build ambient hierarchies. Through these hierarchies, any entity in a cyber–physical system can be modeled regardless of its nature (physical, logical, mobile, or static) or the environment (or context) of that entity. In addition, an ambient contains a process representing its capabilities; i.e., the actions that this ambient is allowed to perform, as well as mobility capabilities, contextual capabilities, and communication capabilities.

Due to its dynamic and hybrid nature, the process of assessing student knowledge in the context described in the previous section can be modeled using the mathematical notation of $CCAs$. The cyber–physical educational environment is, by its nature, a multiagent system that implements processes and services through interaction between various intelligent agents. Each component of the environment is served by one or more specialist assistants, and users are represented in the platform by their personal assistants. Each such intelligent environment component can be represented by a separate CCA. Let us consider the following ambients:

- PA_T—a personal assistant to the teacher;
- PA_Si—a personal assistant of the i-th student;
- SA_TS—a specialist assistant serving the test system in the education space;
- SA_DM—a specialist assistant providing services related to the use of data from the data module;
- SA_SB—a specialist assistant supporting interaction with the student books component;
- AA—an analytical assistant that provides services related to information analysis by using the described fuzzy set approach.

We model the processes of these ambients according to the hybrid approach described above.

The instructor, through their personal assistant, sends a message to the assistant of the test system requesting to open the test for all students. After a student completes the test, their score is recorded in the data module, and the teacher receives information about it. The instructor's personal assistant communicates with the AA ambient with a request to analyze the results of that student according to the considered approach and as a consequence, receives a proposal for an assessment, which he/she sends to the student's virtual student book. The process of this ambient is represented by (2).

$$P_{PA_T} \equiv \left(\begin{array}{c} SA_TS ::< \text{Open the test} > .0| \\ SA_DM :: (\text{Student}_i \text{ completed the test}).0| \\ AA ::< \text{Analyze the results of student}_i > . \\ AA :: (\text{Post–analysis evaluation proposal}). \\ SA_SB ::< \text{Record the grade of student}_i > .0) \end{array} \right) \quad (2)$$

After receiving a request to open the test from the teacher's personal assistant PA_T, the SA_TS ambient sends information to the students' personal assistants. This communication with the i-th student is modeled in (3).

$$P_{SA_TS} \equiv \begin{pmatrix} PA_T :: (\text{Open the test}). \\ SA_Si ::< \text{Test is open, you can start} > .0 \end{pmatrix} \tag{3}$$

As soon as the student finishes working on the test, his/her personal assistant sends a message to the specialist assistant of the data module SA_DM with a request to record the results obtained. The ambient process is represented by (4).

$$P_{PA_Si} \equiv \begin{pmatrix} SA_TS :: (\text{Test is open, you can start}) \\ SA_DM ::< \text{The test is complete, save the result} > .0 \end{pmatrix} \tag{4}$$

The specialist assistant of the data module SA_DM records the results of the students and sends information to the teacher. When it receives a request from the AA ambient, it selects the requested data and sends it for analysis. The process of this ambient is represented in (5).

$$P_{SA_DM} \equiv \begin{pmatrix} PA_Si :: (\text{The test is complete, save the result}) \\ PA_T ::< \text{Student}_i \text{ completed the test} > .0| \\ AA :: (\text{Need data for analysis}). \\ AA ::< \text{Set of data} > .0 \end{pmatrix} \tag{5}$$

The AA ambient analyzes the results of the conducted test after a request from the teacher's personal assistant. To access a particular set of data, it sends a request to the SA_DM ambient. The process is presented in (6).

$$P_{AA} \equiv \begin{pmatrix} PA_T :: (\text{Analyze the results of student}_i).0| \\ SA_DM ::< \text{Need data for analysis} > .0| \\ SA_DM :: (\text{Set of data}). \\ PA_T ::< \text{Postanalysis evaluation proposal} > .0 \end{pmatrix} \tag{6}$$

The closing stage of the implementation of the process is the recording of the final assessment of the students in the administrative system of the virtual student book (SA_SB).

The *ccaPL* programming language is a computer-readable version of the CCA syntax. The interpreter of this language enables testing and verification of the modeled scenario (Figure 5).

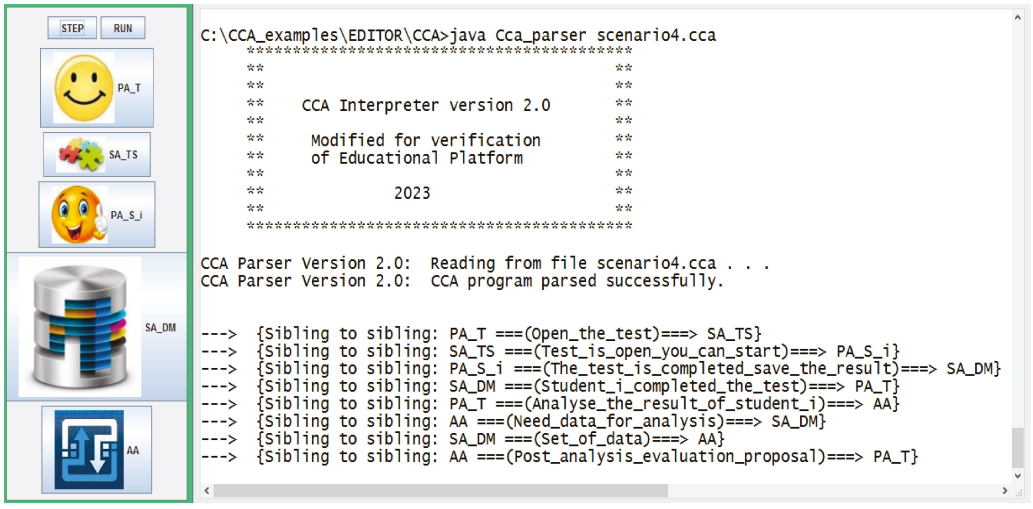


Figure 5. Testing and verification using the *ccaPL* interpreter and animator.

4. Results and Discussion

We analyze the fuzzified marks in this section in order to illustrate that we have obtained fairer marks regardless of the fact that the two distributions differ statistically.

From practical experience, we can verify the use of practice tests can be an effective tool for both student evaluation and learning. By combining tests with open and closed questions and using the fuzzy set technique to correct the results, we achieve more accurate assessment, which helps, to a certain extent, avoid the possibility of randomly selected answers, which is a well-known risk when administering tests. To justify the ethics and credibility of changing borderline grades, we explain why we consider it fair to apply changes to the grades when the two distributions after the fuzzification do not differ statistically and vice versa—that it would not be equitable to make alterations when the two distributions differ statistically.

Let us take as an example the grades of of student number 35 from the list of test takers [4, 6] including 35, 42.0, 20, 5, and 4. Initially, he had obtained the grade good (4) with 42 out of 64 points. Inspecting his test, it becomes obvious that he has some knowledge gaps related to grammar usage; for example, his use of the present perfect tense is rather inconsistent because in some cases he uses it correctly but in others he confuses it with past simple tense. However, he has grasped the use of the present simple and present continuous tenses and has completed all the tasks involving these two tenses appropriately. On the other hand, the student has received the maximum of 20 points for the open question—his answer is coherent, logical, and clearly structured, and his thesis is supported by a relevant example. Moreover, from our observations as teachers of this particular student, we can confirm that he is diligent and hard-working and that he puts a good deal of effort in his work. Therefore, we are convinced that it would be unfair to assess his test with the lower grade—good (4)— but he well deserves the grade of very good 5, obtained after the process of fuzzification when the two distributions do not differ statistically.

Next, we consider the instance when delta is larger, in which case new students are added, whose grades will be automatically revised. As the hypothesis test shows, a great number of the distributions of assessments are statistically different from the classical evaluation, and we prove that the changes in the assessments that occur in this case can be assumed as unfair.

Now, let us regard another example of a student whose grade would be unjust to alter (when the two distributions differ statistically). Let us review the results, for instance, of student number 6 from the list of takes: [5, 6] including 6, 50.6, 18, 6, and 5. His original grade, based on the standard grading system, is very good 5. Although he has obtained a considerably high score on the open question, the mistakes made on the MCQ are quite sporadic; for instance, in certain cases, he has applied a grammar rule correctly, while in others he has not, or the student he has selected the appropriate preposition from a list of options in a closed question and then has used it incorrectly in his written text, which suggests a random choice of answers in the test. Consequently, we believe that it would be improper to correct his grade to the maximum possible score.

On the other hand, ambient-oriented CCA modeling makes it possible to describe, in a unified way, different objects from the physical and the virtual worlds. The development of cyber–physical educational platforms is a labor-intensive, long, and expensive process, which is why the preliminary modeling of the main processes and services in the space is of particular importance. In the context of the need to assess students' knowledge in the cyber–physical educational space, CCA modeling provides abundant opportunities for preliminary verification, testing, and process analysis.

5. Conclusions

The article discusses a hybrid approach to assessing students' foreign language knowledge in a cyber–physical educational environment. The presented assessment approach, through the use of the mathematical theory of fuzzy functions, ensures a fair assessment of

students, which motivates them to take tests conscientiously in order to get the most out of them.

Ambient-oriented CCA modeling provides ample opportunities for preliminary testing, verification, and analysis of the base scenarios related to both student assessment and provision of the entire learning process in the cyber–physical educational space.

Author Contributions: Formal analysis: T.G., V.I. and B.Z.; Methodology: T.G., V.I. and B.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This study was financed by the European Union-NextGenerationEU through the National Recovery and Resilience Plan of the Republic of Bulgaria under project no. DUECOS BG-RRP-2.004-0001-C01.

Data Availability Statement: The datasets used and/or analyzed during the current study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Morales-Salas, R.E.; Infante-Moro, J.C.; Gallardo-Pérez, J. Evaluation of virtual learning environments. A management to improve. *Int. J. Educ. Res. Innov.* **2020**, *2020*, 126–142. [CrossRef]
2. Todorov, J.; Krasteva, I.; Ivanova, V.; Doychev, E. BLISS-A CPSS-like Application for Lifelong Learning. In Proceedings of the IEEE International Symposium on Inovations in Intelligent SysTems and Applications, INISTA 2019—Proceedings, Sofia, Bulgaria, 3–5 July 2019. [CrossRef]
3. National Science Foundation (US). Cyber-Physical Systems (CPS). 2008. Available online: https://www.nsf.gov/pubs/2008/nsf08611/nsf08611.htm (accessed on 20 December 2023).
4. Wang, X.; Yang, J.; Han, J.; Wang, W.; Wang, F.Y. Metaverses and DeMetaverses: From digital twins in CPS to parallel intelligence in CPSS. *IEEE Intell. Syst.* **2022**, *37*, 97–102. [CrossRef]
5. Gürdür Broo, D.; Boman, U.; Törngren, M. Cyber-physical systems research and education in 2030: Scenarios and strategies. *J. Ind. Inf. Integr.* **2021**, *21*, 100192. [CrossRef]
6. National Academies of Sciences, Engineering, and Medicine. A 21st Century Cyber-Physical Systems Education. 2017. Available online: https://nap.nationalacademies.org/catalog/23686/a-21st-century-cyber-physical-systems-education (accessed on 20 December 2023).
7. Stoyanov, S.; Glushkova, T.; Stoyanova-Doycheva, A.; Todorov, J.; Toskova, A. A Generic Architecture for Cyber-Physical-Social Space Applications. *Intell. Syst. Theory Res. Innov. Appl. Stud. Comput. Intell.* **2020**, *864*, 319–343. [CrossRef]
8. Rahnev, A.; Pavlov, N.; Golev, A.; Stieger, M.; Gardjeva, T. New electronic education services using the Distributed E–Learning Platform (DisPeL). *Int. Electron. J. Pure Appl. Math. (IEJPAM)* **2014**, *7*, 63–72. [CrossRef]
9. Council of Europe. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment*; Press Syndicate of the University of Cambridge: Cambridge, UK, 2011. Available online: https://rm.coe.int/CoERMPublicCommonSearchServices/DisplayDCTMContent?documentId=0900001680459f97 (accessed on 20 December 2023).
10. Ivanova, V.; Zlatanov, B. Implementation of fuzzy functions aimed at fairer grading of students' tests. *Educ. Sci.* **2019**, *8*, 214. [CrossRef]
11. Zadeh, L. Fuzzy Sets. *Inf. Control* **1965**, *8*, 338–353. [CrossRef]
12. Fahad, S.; Shah, A. Intelligent testing using fuzzy logic: Applying fuzzy logic to examination of students. *Innov. Learn. Instr. Technol. Assess. Eng. Educ.* **2007**, 95–98. [CrossRef]
13. Gokmena, G.; Akinci, T.; Tektau, M.; Onat, N.; Kocyigit, G.; Tektau, N. Evaluation of student performance in laboratory applications using fuzzy logic. *Procedia Soc. Behav. Sci.* **2010**, *2*, 902–909. [CrossRef]
14. Dias, S.; Diniz, J. *FuzzyQoI* model: A fuzzy logic-based modelling of users' quality of interaction with a learning management system under blended learning. *Comput. Educ.* **2013**, *69*, 38–59. [CrossRef]
15. Troussas, C.; Krouska, A.; Sgouropoulou, C. Collaboration and fuzzy–modeled personalization for mobile game–based learning in higher education. *Comput. Educ.* **2020**, *144*, 103698. [CrossRef]
16. Aldana-Burgos, L.; Gaona-García, P.; Montenegro-Marín, C. A Fuzzy Logic Implementation to Support Second Language Learning Through 3D Immersive Scenarios. *Smart Innov. Syst. Technol.* **2023**, *320*, 501–511. [CrossRef]
17. Nandwalkar, B.; Pardeshi, S.; Shahade, M.; Awate, A. Descriptive Handwritten Paper Grading System using NLP and Fuzzy Logic. *Int. J. Perform. Eng.* **2023**, *19*, 273–282. [CrossRef]
18. Brimzhanova, S.; Atanov, S.; Moldamurat, K.; Baymuhambetova, B.; Brimzhanova, K.; Seitmetova, A. An intelligent testing system development based on the shingle algorithm for assessing humanities students' academic achievements. *Educ. Inf. Technol.* **2022**, *27*, 10785–10807. [CrossRef] [PubMed]
19. Doz, D.; Cotič, M.; Felda, D. Random Forest Regression in Predicting Students' Achievements and Fuzzy Grades. *Mathematics* **2023**, *11*, 4129. [CrossRef]

20. Doz, D.; Felda, D.; Cotič, M. Demographic Factors Affecting Fuzzy Grading: A Hierarchical Linear Regression Analysis. *Mathematics* **2023**, *11*, 1488. [CrossRef]
21. Doz, D.; Felda, D.; Cotič, M. Assessing Students' Mathematical Knowledge with Fuzzy Logic. *Educ. Sci.* **2022**, *12*, 266. [CrossRef]
22. Doz, D.; Felda, D.; Cotič, M. Combining Students' grades and Achievements on the National assessment of Knowledge: A fuzzy Logic Approach. *Axioms* **2022**, *11*, 359. [CrossRef]
23. Doz, D.; Felda, D.; Cotič, M. Using Fuzzy Logic to Assess Students' Mathematical Knowledge. In Proceedings of the Nauka i Obrazovanje—Izazovi i Perspektive, Užice, Serbia, 21 October 2022; pp. 263–278. [CrossRef]
24. Özseven, B.E.; Çağman, N. A Novel Student Performance Evaluation Model Based on Fuzzy Logic for Distance Learning. *Int. J. Multidiscip. Stud. Innov. Technol.* **2022**, *6*, 29–37. [CrossRef]
25. Özseven, B.E.; Çağman, N. A novel evaluation model based on fuzzy logic for distance learning. *Soft Comput.* 2022, *preprint*. [CrossRef]
26. Sobrino, A. Fuzzy Logic and Education: Teaching the Basics of Fuzzy Logic through an Example (by Way of Cycling). *Educ. Sci.* **2013**, *3*, 75–97. [CrossRef]
27. Siewe, F.; Zedan, H.; Cau, A. The calculus of context-aware ambients. *J. Comput. Syst. Sci.* **2011**, *77*, 597–620. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Analysis of a Predictive Mathematical Model of Weather Changes Based on Neural Networks

Boris V. Malozyomov [1], Nikita V. Martyushev [2,*], Svetlana N. Sorokova [3], Egor A. Efremenkov [3], Denis V. Valuev [4] and Mengxu Qi [2]

[1] Department of Electrotechnical Complexes, Novosibirsk State Technical University, 630073 Novosibirsk, Russia; borisnovel@mail.ru
[2] Department of Materials Science, Tomsk Polytechnic University, 634050 Tomsk, Russia; mensyuy1@tpu.ru
[3] Department of Mechanical Engineering, Tomsk Polytechnic University, 634050 Tomsk, Russia; s_sorokova@tpu.ru (S.N.S.); egorefr@tpu.ru (E.A.E.)
[4] Yurga Technological Institute (Branch), Tomsk Polytechnic University, 652055 Yurga, Russia; valuev@tpu.ru
* Correspondence: martjushev@tpu.ru

Abstract: In this paper, we investigate mathematical models of meteorological forecasting based on the work of neural networks, which allow us to calculate presumptive meteorological parameters of the desired location on the basis of previous meteorological data. A new method of grouping neural networks to obtain a more accurate output result is proposed. An algorithm is presented, based on which the most accurate meteorological forecast was obtained based on the results of the study. This algorithm can be used in a wide range of situations, such as obtaining data for the operation of equipment in a given location and studying meteorological parameters of the location. To build this model, we used data obtained from personal weather stations of the Weather Underground company and the US National Digital Forecast Database (NDFD). Also, a Google remote learning machine was used to compare the results with existing products on the market. The algorithm for building the forecast model covered several locations across the US in order to compare its performance in different weather zones. Different methods of training the machine to produce the most effective weather forecast result were also considered.

Keywords: weather mathematical model; forecast; neural network; algorithm for building weather forecasts

MSC: 62M45

1. Introduction

One of the most widespread problems with providing scientific, industrial, and other human activities has been and remains the problem of obtaining information about the environment at the required time [1]. This problem becomes especially important for the support of technical systems (TSs) dependent on meteorological conditions and operating in an autonomous mode. Their parameters cannot be calibrated depending on the meteorological conditions at a certain moment, which makes it necessary to solve the problem of predicting meteorological conditions over a long period of time [2–4]. While predicting values for single-parameter technical systems (SPTs) does not pose special problems, multi-parameter technical systems (MTSs) require a thoughtful approach to creating an algorithm for building a predictive model, which makes this problem very relevant.

The problem of obtaining information about the environment and weather forecasting is of great importance for various spheres of human activity. Scientific research, industrial operations, agriculture, urban planning, tourism, and many other sectors require accurate and up-to-date information on weather conditions, climatic changes, and environmental conditions [5–8].

In the past, environmental information was based mainly on observations and data that were obtained from radio meteorological stations, satellites, sensors, and other devices. Nevertheless, this approach has its limitations due to the large amount of data, the complexity of data processing, and limitations in space and time [9,10].

Due to the development of neural networks and machine learning, new approaches to analysing and predicting weather and environmental conditions have become possible. Neural networks can process large amounts of data and detect complex patterns in weather events, the climate, and other aspects of the environment. They are able to use these patterns to create predictive models that can predict weather and other factors with high accuracy. The application of neural networks in the field of weather and hydrometeorology can improve the quality of forecasts and provide more accurate and reliable data about weather conditions. Neural networks are able to take into account complex interactions between different factors, such as temperature, atmospheric pressure, humidity, and wind, which allows us to obtain more accurate weather forecasts for short and long periods of time [11–13].

In addition to weather forecasting, neural networks can be used to analyse and predict other aspects of the environment, such as air pollution, water quality, vegetation, and ecosystem health. This enables informed decision-making in environmental protection, resource management, and other fields of endeavour where environmental information is an important factor. Determining and predicting weather using neural networks require a large amount of data for training and tuning models. Therefore, it is important to create high-quality datasets that include information on past weather events as well as data collected in real time. In addition, machine learning algorithms and neural networks need to be further developed and improved to achieve more accurate and reliable results. Therefore, the application of neural networks in analysing and forecasting weather and the environment makes it possible to solve complex problems related to the acquisition and processing of information in various spheres of human activity. This increases not only the efficiency of scientific and production processes, but also contributes to environmental conservation and informed decision-making in the field of sustainable development [14–17].

At the moment, various methods and means of obtaining and predicting meteorological data at the desired location of their parameters are known [18–21], including the following:

1. The study of weather phenomena at the current location using physical laws and the Weather Research & Forecasting Model available at https://www.mmm.ucar.edu/models/wrf (accessed on 1 December 2023).
2. Investigation of weather conditions by means of mathematical transformations of data received from the probes.
3. The use of radar and satellite observations to obtain meteorological information. Radars and satellites can provide data on wind speed, temperature, atmospheric pressure, humidity, and other parameters using a variety of sensors and instruments.
4. Application of meteorological balloons and aerological probes that are released into the atmosphere and equipped with meteorological instruments to measure vertical profiles of temperature, humidity, pressure, and wind speed. The obtained data help in weather forecasting and analysing weather conditions.
5. The use of a network of meteorological stations located at various locations. These stations have meteorological instruments that continuously monitor and record data on weather conditions, such as temperature, humidity, pressure, and precipitation. The information from these stations helps in forecasting and analysing the weather at the location.
6. Utilization of computer weather prediction models that are used to analyse and forecast meteorological data. These models take into account the physical laws of the atmosphere, data from meteorological observations, and other parameters to create forecasts of future weather conditions.

7. Usage of remote sensing, such as LIDAR (laser sensing), and infrared sensing to measure atmospheric parameters and obtain weather data. These technologies are based on the analysis of reflected or emitted radiation and can measure temperature, humidity, cloud cover, and other weather characteristics at the desired location.

All these methods and tools are used together to acquire and forecast meteorological data at a given location, which is essential for weather forecasting and the planning of agricultural and urban activities, including safety measures and protection against weather disasters.

The current number of solutions to the problem of building meteorological forecast models and analysing them is staggering. One of the most successful, but at the same time most costly, solutions is the solution provided by Weather Research & Forecasting, which is reduced to the construction of a mathematical model of physical phenomena and, based on the current data, the application of a certain location. However, this approach rarely gives accurate results because it requires a huge number of resources for the solution and constant calculations taking into account the changing parameters of the system. Such a solution also demands the same construction of models for neighbouring regions, which increases the cost of accurate forecasting exponentially, and also does not take into account the essential factors.

Neural networks use one of the most promising methods of MTS forecasting, which is mathematical extrapolation, as this method is based only on the statistical analysis of data [22–24]. The drawback of this approach is the impossibility of extrapolating the further development of the process for a long time interval. For this purpose, patterns between input and output values are built, relying on training data. In the case of meteorological forecasting, such data are previous observations collected by the personal weather stations (PWSs).

This paper uses a learning paradigm in which a finite number of neural networks are trained based on the same data to obtain the desired result. This paradigm is called grouping neural networks [25–28].

The purpose of this study is to describe, analyse, and compare an algorithm for building a predictive model for an MTS for a particular location using the example of data obtained from personal weather stations for certain regions using self-learning machines. The structure of the paper is as follows. The introduction presents the relevance and scientific novelty of the analysis of the predictive mathematical model of weather changes based on neural networks. The second section analyses the structure and application of artificial neural networks. In the third section, the mathematical models of the used neural networks are given. The fourth section describes the mathematical model of the grouping of neural networks proposed for weather forecasts. The fifth section gives an overview of the performance results of the considered mathematical models. The sixth section presents the main conclusions of this work.

2. Analysing the Application of Artificial Neural Networks

2.1. Stages of Neural Network Construction

When building a neural network, several stages can be distinguished, which can be described in general terms as follows:

- Defining goals and objectives;
- Data collection;
- Data preprocessing;
- Selection of the neural network architecture;
- Determination of hyperparameters;
- Training the network;
- Model evaluation and tuning;
- Model development and optimisation.

Each of these stages has its own peculiarities and may require additional actions and experiments to achieve the desired result. To summarise, the main stages of a neural network's construction are shown in Figure 1.

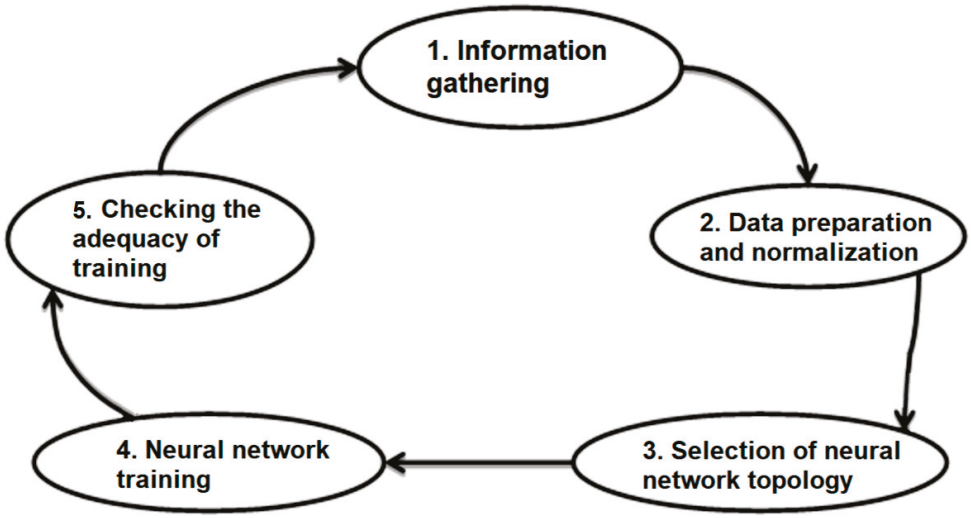


Figure 1. Main stages of a neural network's construction.

The first step is to collect weather data. The second step is to normalise and discard corrupted data obtained from Weather Underground and the US National Digital Forecast Database. The normalisation process depends entirely on the choice of topology in the previous step; for example, for RBFN, it is necessary to generate the data in such a way that the training data contain the correct answer.

At the third stage, a mathematical model is selected to build functional dependencies between input and output data to obtain the required result. In this paper, three topologies are considered (D-PNN, RBFN, and MLPN). A new one is also proposed (the grouping method of neural networks).

At the fourth stage, the neural network is trained by feeding pronormalised data to the inputs of the mathematical model. The neural network builds functional dependencies between the data. Finally, the output is a function of the dependence of input and output values in a certain training dataset.

At the fifth stage, the obtained model is checked in terms of accuracy and adequacy. In forecasting, it is necessary to accurately establish the concept of the necessary accuracy of the result. In this paper, the model is considered adequate when its forecast accuracy is 10%, and the accuracy score is calculated as the percentage ratio between the data obtained by the model and the real data.

2.2. Improving the Prediction Accuracy of Neural Networks

Increasing the prediction accuracy of neural networks is an important goal in the field of machine learning, and various approaches and techniques have been used to achieve this goal. Furthermore, ways to improve the prediction accuracy of neural networks have been formulated and proposed, including the following:

- Increasing the training sample size: A large amount of diverse data can help a neural network learn a wider range of patterns and improve generalisability. This may involve collecting additional data, augmenting existing data, or using generative modelling techniques to create new samples [29–31].
- Improving data quality: Data quality is also important. Data preprocessing should be done to eliminate noise, outliers, missing values, and other problems. In addition, methods can be applied to eliminate class imbalance or sampling imbalance problems.
- Fitting the network architecture: Choosing the right neural network architecture plays an important role in achieving high prediction accuracy. By using techniques such

as hyperparameter search or optimisation by automatic machine learning, different network configurations can be explored, including the number of layers, the number of neurons in each layer, and activation functions.

- Model regularising: Regularisation techniques can help to control overfitting and generalise the model better to new data. This includes methods such as L1 or L2 regularisation, neuron or layer dropout, and data dimensionality reduction techniques, such as principal component analysis (PCA) and stochastic neighbour embedding with t-distribution (t-SNE).
- Using pretraining: Pretraining can be useful, especially when there are not enough labelled data available. One can use pretrained models using large unlabelled datasets (e.g., autoencoders or generative models) and then pretrain them based on smaller labelled datasets for a specific task.
- Combining models: Sometimes, combining multiple models or using ensembles of models can significantly improve the forecast accuracy. This can be done by combining forecasts from several models or by using stacking or bagging techniques.
- Hyperparameter tuning: The hyperparameters of a model also affect its prediction accuracy. Optimisation techniques, such as grid search or random search, can be used to find optimal values for the hyperparameters.
- Improving the learning process: Applying various optimisation techniques, such as stochastic gradient descent with momentum (SGD with momentum), Adam's method, and early stopping, can help to train the neural network more efficiently and stably.

Figure 2 shows the increase in neural network prediction accuracy using generalised data from the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) project [32].

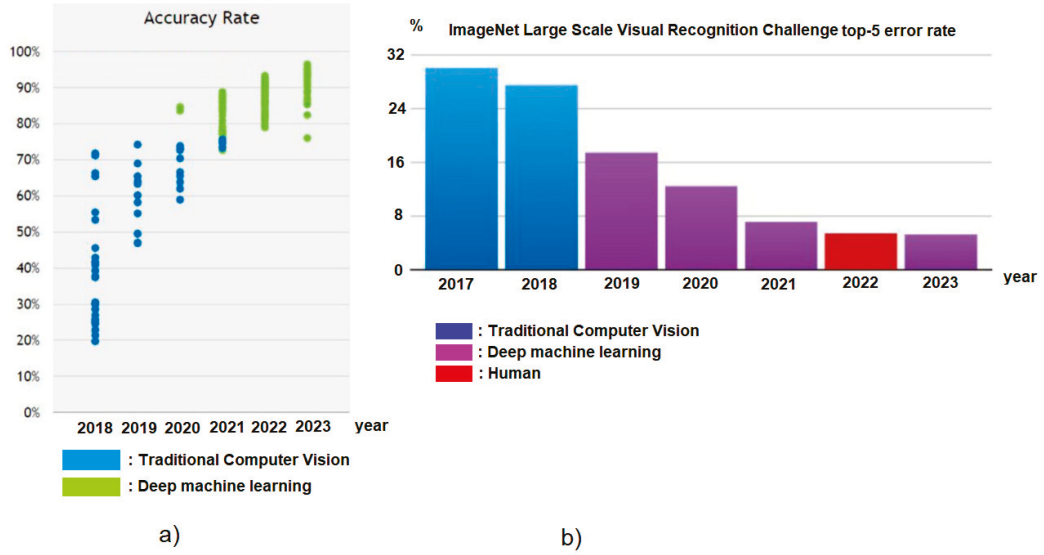


Figure 2. Increase in neural network prediction accuracy: (**a**) accuracy rate and (**b**) top-5 error rate in the ILSVRC.

The top-five error rate indicates the percentage of test cases (Figure 2b) where the correct class was not among the top five predicted classes. For instance, if a test image features a Persian cat and the top five predicted classes, ranked from highest to lowest probability, are (Pomeranian (0.4), Mongoose (0.25), Dingo (0.15), Persian cat (0.1), and Tabby cat (0.02)), it is still considered accurate because the real class is within the top five predictions. With ImageNet's 1000 classes, achieving a low error rate in the top five predictions is challenging [33].

Modern artificial neural networks can be divided into simple and complex (perceptrons) depending on the number of neuron layers (one or more). They are also divided into direct and recurrent networks [34–36].

Direct neural networks are used to send the signal from the input to the output in a straight line, like a train on a track. Recurrent ones allow for the possibility that the resulting intermediate value can be sent back to the input again and go through the neural network from the beginning. The biological brain is a recurrent network, which is the reason why it is almost impossible to understand how it works.

Each neural network is a data analysis system that is very powerful and accurate but requires special tuning (training). Moreover, taking into account the self-learning capacity of a neural network, its tuning is not reduced to setting specific parameters according to which it will work. The principle of training is quite different:

- A certain value is given to the input.
- It is known in advance what value should be at the output.
- In case the output value is different from the desired value, the network is adjusted until the difference is minimised [37].

The second variant of neural network training is the so-called method of error back-propagation, in which the value obtained at the output, if it differs from the initial one, is transmitted back through the same neurons through which it came to the output. In the process of transmission, the weights of these neurons are increased or decreased. Then, a new attempt follows, and so on until the result becomes optimal. Figure 3 shows the general scheme of the neural network training process proposed for predicting weather changes.

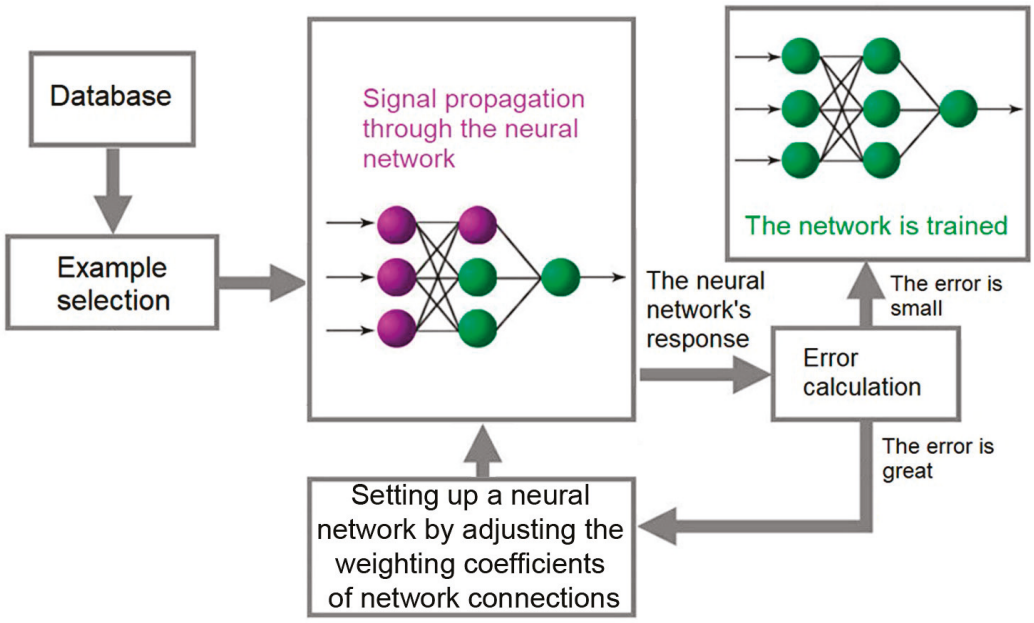


Figure 3. Schematic diagram of the training process of the weather neural network.

3. Mathematical Models of the Used Neural Networks

The first neural network is based on the solution of the differential polynomial (the differential polynomial neural network (D-PNN). The D-PNN describes functional dependencies of input parameters and their properties, which are often used for forecasting, analysing time series, and revealing hidden relationships in data [38–40]. This is achieved by templating the dependencies between the data. The main idea of the D-PNN is the approximation of functions described by differential equations that describe the relationships between the input parameters of the system. The relationships between the data are described similarly to the Kolmogorov–Gabor polynomial presented in 1971 [2]:

$$y = a_0 + \sum_{i=0}^{m}\sum_{j=0}^{m} a_{ij}x_i x_j + \sum_{i=0}^{m}\sum_{j=0}^{m}\sum_{k=0}^{m} a_{ijk}x_i x_j x_k + \ldots, \quad (1)$$

where m is the number of variables, $x(x_1, x_2, x_3, \ldots)$ are vectors of input variables, and $a(a_1, a_2, a_3, \ldots)$ are vectors of parameters.

The second neural network is a neural network that uses radial basis functions as activation functions. This network is called the Radial Basis Function Network (RBFN) [41–43]. The RFBN is very popular for function approximation, belt series prediction, and classification. In such networks, it is very important to determine the number of neurons in the hidden layer, as it strongly affects the complexity of the network and its generalisation capabilities. In the hidden layer, each neuron has an activating function. The Gaussian function, which has a parameter controlling the behaviour of the function, is the most preferable activation function [34].

$$f(x) = ae^{-\frac{(x-b)^2}{2c^2}}, \tag{2}$$

where a, b, and c are real numbers.

The third neural network is based on the multilayer perceptron network (MLPN) [35–37].

This type of network is known for requiring the presence of the desired result in the training dataset. In our case, it is necessary to supply an additional value that will be the correct result for the input data. In other words, this model relates the input values to their result using historical weather data to predict the weather in the future, which is very important in normalising the data [38].

MLPNs have shown the ability to find approximate solutions for extremely complex problems. In particular, they are a universal function approximator, so they can be successfully used in the construction of regression models. Since classification can be considered as a special case of regression when the output variable is categorical, MLPs can be used to build classifiers [39,40].

Table 1 summarises the comparative performance of the neural networks proposed for weather forecasting.

Table 1. Comparative characteristics of neural networks.

Principle of Model Construction	Mathematical Description	Input Information	Output Information
D-PNN			
Approximation of functions described by differential equations that describe relationships between input parameters of the system	$y = a_0 + \sum_{i=0}^{m}\sum_{j=0}^{m} a_{ij}x_i x_j + \sum_{i=0}^{m}\sum_{j=0}^{m}\sum_{k=0}^{m} a_{ijk}x_i x_j x_k + \ldots$	$x(1), x(2), x(3), x(4) \ldots x(n)$	$x(n+1)$
RBFN			
Approximation of the unknown solution by means of functions of a special kind, whose arguments are distant	$y_i(t) = w_{i0} + \sum_{j=1}^{n_k} \lambda_{ij} vl(t) + \sum_{j=1}^{n_k} \varphi(\left\|v(t) - c_j(t)\right\|)$	$x(1), x(2), x(3), x(4) \ldots x(n)$	$x(n+1)$
MLPN			
Approximation of the unknown solution using nonlinear functions	$y_i(t) = \phi(\sum_{j=1}^{n} w_{ij}x_j^k + b_i)$	$x(1), x(2), x(3), x(4) \ldots x(n)$	$x(n+1)$

4. Description of the Mathematical Model of Neural Network Grouping

Neural network grouping, also known as a neural network ensemble or a neural network committee, is a technique in which multiple independent neural networks are trained on the same task in order to improve the prediction quality or system reliability [41–43].

The application of neural network clustering has several advantages. First, different neural networks can capture different aspects of the data or extract different characteristics. By combining the predictions of these neural networks, the diverse aspects of the data

can be taken into account and the generalisability of the model can be improved. Second, grouping neural networks can help with robustness in terms of noise and outliers. If one neural network makes an error using certain examples, other neural networks can compensate for this error and improve the overall prediction accuracy [44–47].

The grouping of neural networks can be implemented in various ways:

- Bagging: This method involves training a set of independent neural networks on subsets of training data obtained by selecting bootstrap examples. The predictions from each neural network are then combined, for example, by majority voting or averaging.
- Boosting: Unlike bagging, boosting builds a sequence of neural networks, each of which learns to correct the errors of previous networks. At each iteration, the weights of the samples are adjusted to focus on the examples where the previous networks made a mistake. Hence, subsequent neural networks focus on complex and poorly classified examples [48].
- Stacking: This method is used when the predictions of several neural networks become inputs to another model (a meta-model) that produces the final prediction. In this way, the meta-model is trained to combine and utilise the predictions of different neural networks to produce the best overall result.

Each of these approaches has its own characteristics and is suitable for different scenarios and tasks. The choice of a particular neural network clustering method depends on data characteristics, the required accuracy, computational resources, and other factors. Neural network clustering is a powerful tool used to improve the prediction accuracy and increase the confidence in decision-making. This method is actively used in various fields including computer vision, natural language processing, and speech recognition [49].

Grouping neural networks can be used to predict weather parameters, such as temperature, humidity, and wind speed. This is an important task that helps with planning and decision-making in various fields including agriculture, energy, urban planning, and tourism. One way of applying neural network grouping for weather forecasting is by using the bagging method. In this case, different neural networks are trained on different subsets of the original data with different characteristics (e.g., different time intervals and geographical areas). The forecasts from each neural network are then combined to produce the final weather forecast. This approach can help to account for different features and nuances of weather conditions in different areas [50–52].

When applying neural network boosting, a sequence of neural networks learns to predict weather parameters by correcting the errors of previous neural networks. At each iteration, the sample weights or errors of the previous networks are used to emphasise areas where the previous neural networks made an error. In this way, more complex and difficult-to-predict situations can be handled by subsequent neural networks. When neural network stacking is applied to weather forecasting, forecasts from multiple neural networks become inputs to a meta-model that produces the final weather forecast. The meta-model is trained to combine and utilise the forecasts from different neural networks to produce the best overall result [53,54].

An important aspect in weather forecasting is the use of various input features, such as data from weather stations, satellite observations, geographical data, and historical data. Grouping neural networks allows us to combine information from these different sources, which can improve the quality and accuracy of forecasts [55].

All these methods of grouping neural networks for weather forecasting require large amounts of data and computational resources. In addition, it is important to carefully select the structure and parameters of neural networks and train them on a sufficient amount of diverse data to achieve good results [56,57].

The development of more powerful computing resources and the advancement of deep learning have made grouping neural networks increasingly popular and successful in the field of weather forecasting. However, it is always necessary to consider the nature of the complexity of weather processes and the limitations of modelling. Weather forecasts by

their nature remain tentative and there remains the possibility that an error will appear in the result, especially in the case of long-term forecasts.

In view of this, neural network grouping is defined by an approach to building a self-learning machine in which a finite number of neural networks is trained to solve the same task. This approach originates from a paper by Hansen and Salamon [58], which shows that the neural network system can be significantly improved by the grouping approach, which means that the predictions produced by such a machine are much more accurate. In general, the grouping of neural networks can be divided into two steps: training several neural networks and then combining and processing the predictions of each. The result of such a system is the averaged value of the outputs of each neural network separately, combined with a function describing the comparative deviation of values obtained at the training stage relative to each other. The results of such systems significantly improve the accuracy of predictions [59,60]. In this paper, a new approach for training these systems will be considered. The weighting coefficients are proportional to the corresponding output values. The essence of the approach is to determine which neural network produces more accurate forecasts. Let us consider an example. One may suppose that there are two neural networks that have to perform a simple classification task. If the input is 1, then the output is 1; if the input is 0, then the output is 0. Let the neural networks' output be 0.6 and 0.9, respectively, at a certain step. In this case, the second machine receives much more reliable data because 0.9 is closer to 1.

Backpropagation networks set the initial weighting factors randomly to reduce the standard deviation [61,62]. The difference in initial weighting coefficients gives different results. Therefore, grouping neural networks integrates these independent networks to improve the generalisation ability. This method also guarantees an increase in accuracy in terms of a standard deviation [63–66].

In this paper, we propose a grouping of nonlinear leading networks created using a constructive algorithm. In constructive algorithms, the number of neurons in hidden layers is initially small and then gradually increases. Hence, in constructive algorithms, the skills acquired by the network before increasing the number of neurons are preserved.

Constructive algorithms differ in their rules for setting parameter values in the new neurons added to the network:

- Parameter values are random numbers from a given range;
- Values of synaptic weights of the new neuron are determined by splitting one of the old neurons.

The first rule does not require significant computation, but its use leads to an increase in the value of the error function after each addition of a new neuron. As a result of the random assignment of parameter values of new neurons, a redundancy in the number of hidden layer neurons may appear. Neuron splitting is devoid of these two disadvantages. The essence of the splitting algorithm is illustrated in Figure 4.

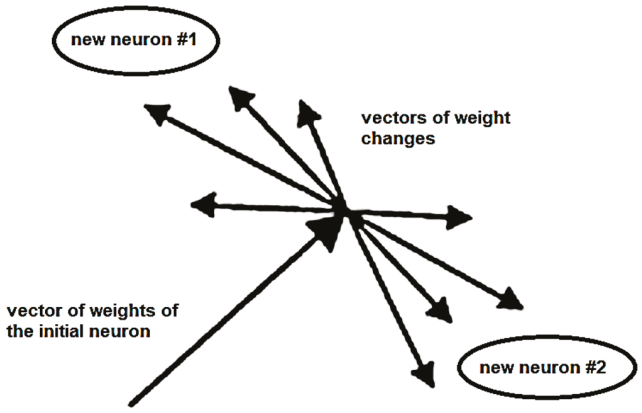


Figure 4. Vector of hidden layer neuron weights and changes corresponding to individual training examples.

Figure 4 shows the vector of weights of the hidden layer neuron at some training step and the vectors of weight changes corresponding to individual training examples. The change vectors have two preferential directions and form a region in space that is significantly different from the spherical one. The essence of the algorithm is to identify and split such neurons. As a result of splitting, there are two neurons instead of the one initial neuron in the network. The first of these neurons has a vector of weights, which is the sum of the vector of weights of the original neuron, and vectors of changes in the weights of one of the preferential directions. The summation of the vectors of weight changes in the other preferential direction and the vector of weights of the original neuron results in the synaptic weights of the second new neuron [64].

It is necessary to split neurons whose change vectors have two preferential directions because the presence of such neurons leads to oscillations during training by backpropagation. When training the method with an integral error function, the presence of such neurons leads to a hit in the local minimum with a large error value.

The splitting algorithm includes the construction of a covariance matrix of vectors of changes in synaptic weights and the calculation of eigenvectors and eigenvalues of the obtained matrix using the iterative Oja algorithm, according to which stochastic gradient lifting and Gramm–Schmidt orthogonalisation are performed [65].

It is necessary to consider a single neural network that is trained on some dataset. Let x be the input vector that appears for the first time in this network and d be the desired outcome. Values x and d represent the realisation of a random vector X and a random variable D, respectively. Let $F(x)$ be the input–output function realised using the network. Then,

$$E_D\left[(F(x) - E[D \mid X = x])D^2\right] = B_D\left(F(x)\right) + V_D\left(F(x)\right), \tag{3}$$

where $E[D \mid X = x]$ is the mathematical expectation, $B_D(F(x))$ is the square of the bias

$$B_D\left(F(x)\right) = \left(E_D\left[F(x)\right] - E[D \mid X = x]\right)^2, \tag{4}$$

and $V_D\left(F(x)\right)$ is the difference:

$$V_D\left(F(x)\right) = E_D\left[(F(x) - E_D\left[F(x)\right])^2\right]. \tag{5}$$

The expectation E_D on the set D is called the set covering the distribution of all training data, such as input and output values, and the distribution of all initial conditions. There are several ways to individually train a neural network and several ways to group the output data. This paper assumes that the networks have the same configurations, but their training starts from different initial conditions. An average simple grouping is used to combine the outputs of a group of neural networks. Let ψ be the set of all initial conditions and $F_I\left(x\right)$ be the average input–output value of the network functions. Then, by analogy with Equation (3), we obtain

$$E_\psi\left[\left(F_I\left(x\right) - E[D \mid X = x]\right)^2\right] = B_\psi\left(F(x)\right) + V_\psi\left(F(x)\right), \tag{6}$$

where $B_\psi\left(F(x)\right)$ is the square of the deviation defined by the set ψ:

$$B_\psi\left(F(x)\right) = \left(E_\psi\left[F(x)\right] - E[D \mid X = x]\right)^2 \tag{7}$$

and $V_\psi\left(F(x)\right)$ is the corresponding difference:

$$V_\psi\left(F(x)\right) = E_\psi\left[\left(E_I\left(x\right)E_\psi\left[F(x)\right]\right)^2\right]. \tag{8}$$

The mathematical expectation E_ψ is taken for the set ψ.

From the definition of set D, we can think of it as the set of initial conditions ψ and the remaining set is denoted by D'. By analogy with Equation (3), we obtain:

$$E_{D'}\left[(F_I(x) - E[D \mid X = x])^2\right] = B_{D'}(E_I(x)) + V_{D'}(F(x)), \tag{9}$$

where $B_{D'}(F_I(x))$ is the square of the deviation defined by the set D':

$$B_{D'}(F_I(x) = (E_{D'}[E_I(x)] - E[D \mid X = x])^2 \tag{10}$$

and $V_{D'}(F_I(x))$ is the corresponding difference:

$$V_{D'} = E_{D'}\left[(E_I(x) - E_{D'}[E_I(x)])^2\right]. \tag{11}$$

From the definition of sets D, ψ, and D', it is obvious that:

$$E_{D'}[F_I(x)] = E_D[F(x)]. \tag{12}$$

It follows that Equation (10) can be rewritten in an equivalent form:

$$V_{D'}(F_I(x)) = (E_D[F(x)] - E[D \mid X = x])^2 = B_D(F(x)). \tag{13}$$

Having the difference $V_{D'}(F_I(x))$ from Equation (11), since the difference in a random variable is equivalent to the RMS value of a random variable, let us subtract its square of deviation:

$$V_{D'}(F_I(x)) = E_{D'}\left[(F(x))^2\right] - (E_{D'}[F(x)])^2 = E_{D'}[(F(x))] \tag{14}$$

or

$$V_D(F_I(x)) = E_D\left[(F(x))^2\right] - (E_D[F(x)])^2. \tag{15}$$

It is worth considering that the RMS value of function $F(x)$ on set D must be greater than or equal to the RMS value of function $F_I(x)$ on set D'.

$$E_D\left[(F(x))^2\right] \geq E_{D'}[(F_I(x))]. \tag{16}$$

Provided that there are Equations (14) and (15), we have:

$$V_{D'}(F_I(x)) \leq V_D(F(x)). \tag{17}$$

Hence, based on Equations (13) and (17), two conclusions can be drawn:

1. The bias of function $F_I(x)$ referring to multiple classification systems is exactly the same as the bias of function $F(x)$ referring to a single neural network.
2. The difference in function $F_I(x)$ is smaller than that in function $F(x)$.

5. Review of the Results of the Considered Mathematical Models

The grouping of linear neural networks has shown improved performance with respect to individual networks. This paper presents the grouping of nonlinear anticipatory networks generated by a constructive algorithm. A similar approach has been applied to the previously discussed D-PNN, RBGN, and MLP.

Statistically, data from Weather Underground personal weather stations and the US National Digital Forecast Database (NDFD) were used to build this model. In addition, Google's remote learning machine was used to compare the results with existing products on the market.

As a result, the input parameters were historical data obtained from the PWS, and the result of training is a function of the dependence of these parameters on each other. Then, when entering sufficient data in the request for a forecast into the future, we will obtain the data we need. In order to build a forecast based on the above-mentioned mathematical

models, we need to arrange the obtained data in the form of an array of enumerated parameters (Table 2).

Table 2. Example of input data for training.

Temp	City	Daytime	Day_of_Year	Year	Humidity%	Wind Speed kph	Pressure in mBar	Weather Conditions
13.2	1	3	1	2022	18	6	1300	3
14.6	1	4	1	2022	19	7	1150	4
...	...	...	...	...	...	...	...	...

In this case, since the model operates using numerical data, cities and weather descriptions were assigned specific indices, e.g., 1 in the city column means Los Angeles. In this study, several training methods were applied by manipulating the input values. Table 3 shows the average deviations of the predicted temperature values.

Table 3. Average deviations of predicted temperature values.

Time of Year	Assemble	D-PNN	RBFN	MLPN
Winter	12%	18%	15%	16%
Spring	10%	16%	14%	13%
Summer	8%	14%	16%	11%
Autumn	9%	13%	16%	12%

Let us determine how the models work for different seasons. For this purpose, we enter data for 5 years and compare the obtained data with real values for different periods (0:00–23:00 on 1 January 2022, 0:00–23:00 on 1 April 2022, 0:00–23:00 on 1 July 2022, and 0:00–23:00 on 1 October 2022). Figure 5 shows the predicted air temperature values obtained by each method and the grouping method for 1 January 2022.

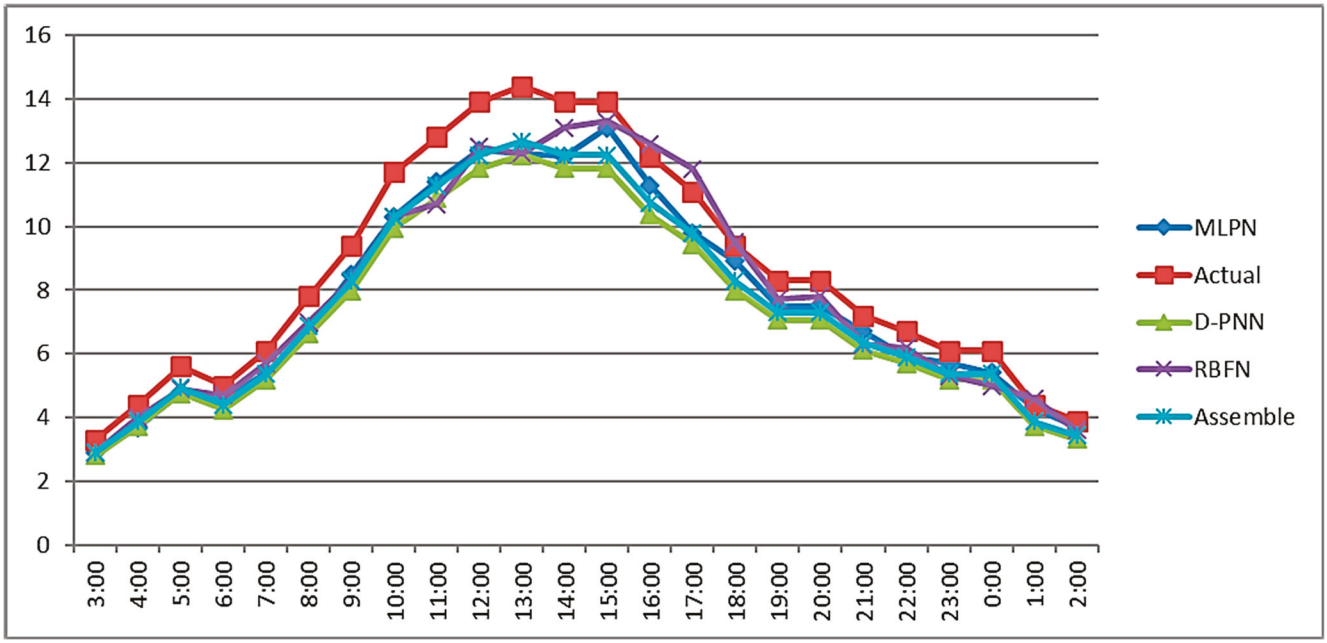


Figure 5. Graph of forecast values obtained by each method as of 1 January 2022.

When building the forecasting system, its main parameters were determined (forecast horizon, forecast period, and forecast interval). The forecast period is the basic unit of time for which the forecast is formed. The forecast horizon is the number of time periods in the future that the forecast covers. For example, there is a forecast for 10 weeks into the future

with data for each week. In this case, the period is one week and the horizon is ten weeks. The forecast interval is the frequency with which a new forecast is generated. Usually, the forecast interval coincides with the forecast period. In this case, the forecast is revised each period using the last period's claim and other current information as a baseline for the revised forecast. In our case, the forecast theoretical horizon is 1 year.

This work presents a grouping of nonlinear leading networks created using a constructive algorithm. A similar approach was applied to the previously considered D-PNN, RBGN, and MLP. Therefore, the created method of neural network clustering has higher accuracy in all seasons and has an acceptable accuracy of 91%. The pro-forecast accuracy of the other linear neural networks showed the following results of maximum–maximum error: for MLPN it is 13%, for D-PNN it is 15%, and for RBFN it is 11%. The computation time required to obtain a monthly forecast is 4 min. So, predicting the meteorological forecast of temperature changes for a year would take 48 min. Hence, it is possible to realise a forecasting model in a real-time system.

The forecast quality of the weather neural network model has increased by about 2 percent in the last two years (2021–2022). Table 4 shows the rankings of temperature prognostic models in 2021–2022 for different countries, with a forecast accuracy of 180 h. To date, the accuracy of most existing predictive weather models is limited to 120 h, while we managed to increase this forecast period to 180 h.

Table 4. Ranking of temperature predictive models in 2021–2022. Forecast accuracy at 180 h.

№	Country	Accuracy Years 2021/2022, %			
		Assemble	**D-PNN**	**RBFN**	**MLPN**
1	USA	89.1/91.4	75.65/77.3	79.21/80.99	77.4/79.2
2	Canada	87.3/90.1	73.9/76.5	77.43/80.1	75.6/78.3
3	Europe	88.7/91.6	74.8/77.35	78.32/80.99	76.5/79.2
4	UK	89.4/90.0	75.65/76.5	79.21/80.1	77.4/78.3
5	Australia	87.2/90.3	73.95/76.5	77.43/80.1	75.6/78.3
6	Germany	88.8/90.7	74.8/76.5	78.32/80.1	76.5/78.3
7	Japan	86.2/88.0	73.1/74.8	76.54/78.4	74.8/76.6
8	China	87.9/88.2	73.95/74.8	77.43/78.3	75.6/76.6
9	Korea	88.3/90.8	74.8/76.5	78.32/80.1	76.5/78.3
10	Russia	86.5/90.2	73.1/76.5	76.54/80.1	74.8/78.3

The neural network grouping method is an approach that combines the forecasts of multiple neural networks to produce a more accurate and stable weather forecast. This method uses an ensemble of neural networks that are trained based on historical weather data and other relevant parameters. One of the key aspects of this method is the use of different training algorithms for each neural network in the ensemble. This can include different neural network architectures, various hyperparameters, and various optimisation algorithms. This approach allows for a variety of models that can better capture the characteristics and complexities of different weather seasons.

To achieve a higher forecast accuracy in all weather seasons, the neural network clustering method can apply the averaging or weighting of the contributions of each neural network. This allows us to smooth out possible errors in individual models and take into account different forecast scenarios for different weather seasons. One of the advantages of the neural network clustering method is its ability to adapt to changes in weather conditions. The ability to train and combine models based on current and updated data allows the method to respond quickly to weather changes and provide high forecast accuracy even in new seasons.

Of course, achieving such forecast accuracies requires the preparation of high-quality data and the careful tuning of parameters. An optimal choice of models, the optimisation of hyperparameters, and the use of a large amount of data allow us to achieve the best results. It should be noted that the neural network clustering method may have the potential

for application in other areas requiring accurate forecasting, such as financial markets, transportation systems, and energy. Its ability to combine forecasts from multiple models may be useful for improving the forecast accuracy in other areas as well. In this work, we determined that the neural network clustering method is a powerful tool for weather forecasting that provides higher forecast accuracy in all weather seasons and is able to adapt to changes in weather conditions. The application of this method can lead to more reliable and accurate weather forecasts in different areas and has significant practical application.

Therefore, models built using the proposed neural network grouping method can account for a larger number of dependent variables than that of individual neural models, which ultimately improves their accuracy. The grouping of neural networks is an effective tool for working with satellite data in the form of large amounts of weather data, which was shown by the example of the classification of forecast values of objects (Figure 5). Currently, we are developing clustered weather models for such phenomena as snow, cloudiness, ice, and short-term precipitation forecasting based on the applied method of neural network clustering using satellite data and numerical predictive models. All approaches described in this paper were tested using specially generated datasets by experienced decoding specialists. The current algorithms accepted in real operational practice were also compared. The successful results from testing the models presented in this paper allowed us to implement them in the operations of the Novosibirsk weather centre of SIC "Climat". There were almost no limitations in applying our model in different locations and weather zones in terms of data, as the proposed model involves the processing of large datasets.

6. Conclusions

In this paper, an algorithm for the construction of meteorological forecast models using a grouping of neural networks was developed and investigated. The algorithm for building a mathematical model for predicting future states of meteorological system parameters based on differential polynomials, radial basis functions, multilayer perceptrons, and groupings of neural networks was considered.

The model built using the proposed method of neural network clustering allowed us to consider a larger number of dependent variables than that of individual neural models, which ultimately improved their accuracy. Clustering neural networks is an effective and promising method for processing meteorological information in the form of large datasets on various weather factors, including snow, cloudiness, ice, and short-term precipitation, based on the applied method of neural network clustering and the creation of numerical prognostic models.

Based on the numerical experiment, it can be concluded that the combination of mathematical modelling and "correct" input data related to weather phenomena can make the meteorological forecast model more accurate. In the future, we will continue to study meteorological forecast models in order to not only improve the order and accuracy of input data but also change the mathematical basis for building the model itself.

Author Contributions: Conceptualization, B.V.M. and N.V.M.; methodology, S.N.S. and E.A.E.; software, M.Q.; validation, S.N.S. and E.A.E.; formal analysis, S.N.S. and E.A.E.; investigation, D.V.V.; resources, D.V.V.; data curation, D.V.V.; writing—original draft preparation, B.V.M. and N.V.M.; writing—review and editing, B.V.M. and N.V.M.; visualization, M.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available from the corresponding authors upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Baldauf, M.; Seifert, A.; Forstner, J.; Majewski, D.; Raschendorfer, M.; Reinhardt, T. Operational convective-scale numerical weather prediction with the COSMO model: Description and sensitivities. *Mon. Wea. Rev.* **2011**, *139*, 3887–3905. [CrossRef]
2. Brisson, E.; Blahak, U.; Lucas-Picher, P.; Purr, C.; Ahrens, B. Contrasting lightning projection using the lightning potential index adapted in a convection-permitting regional climate model. *Clim. Dyn.* **2021**, *57*, 2037–2051. [CrossRef]
3. Voitovich, E.V.; Kononenko, R.V.; Konyukhov, V.Y.; Tynchenko, V.; Kukartsev, V.A.; Tynchenko, Y.A. Designing the Optimal Configuration of a Small Power System for Autonomous Power Supply of Weather Station Equipment. *Energies* **2023**, *16*, 5046. [CrossRef]
4. Kleyko, D.; Rosato, A.; Frady, E.P.; Panella, M.; Sommer, F.T. Perceptron Theory Can Predict the Accuracy of Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, 1–15. [CrossRef] [PubMed]
5. Stephan, K.; Schraff, C. Improvements of the operational latent heat nudging scheme used in COSMO-DE at DWD. *COSMO Newsl.* **2008**, *9*, 7–11.
6. Steppeler, J.; Doms, G.; Schaettler, U.; Bitzer, H.-W.; Gassmann, A.; Damrath, U.; Gregoric, G. Meso-gamma scale forecasts using the nonhydrostatic model LM. *Meteorol. Atmos. Phys.* **2003**, *82*, 75–96. [CrossRef]
7. Armenta, M.; Jodoin, P.-M. The Representation Theory of Neural Networks. *Mathematics* **2021**, *9*, 3216. [CrossRef]
8. Bengio, Y.; Goodfellow, I.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2015.
9. Meng, L.; Zhang, J. IsoNN: Isomorphic Neural Network for Graph Representation Learning and Classification. *arXiv* **2019**, arXiv:1907.09495. [CrossRef]
10. Sorokova, S.N.; Efremenkov, E.A.; Qi, M. Mathematical Modeling the Performance of an Electric Vehicle Considering Various Driving Cycles. *Mathematics* **2023**, *11*, 2586. [CrossRef]
11. Dozat, T. Incorporating Nesterov momentum into Adam. In Proceedings of the ICLR Workshop, San Juan, Puerto Rico, 2–4 May 2016; Volume 1, pp. 2013–2016.
12. Xie, S.; Kirillov, A.; Girshick, R.; He, K. Exploring Randomly Wired Neural Networks for Image Recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving Deep into Rectifiers: Surpassing Human-level Performance on ImageNet Classification. *arXiv* **2015**, arXiv:1502.01852. Available online: https://arxiv.org/pdf/1502.01852.pdf (accessed on 1 December 2023).
14. Kukartsev, V.V.; Martyushev, N.V.; Kondratiev, V.V.; Klyuev, R.V.; Karlina, A.I. Improvement of Hybrid Electrode Material Synthesis for Energy Accumulators Based on Carbon Nanotubes and Porous Structures. *Micromachines* **2023**, *14*, 1288. [CrossRef]
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
16. Kalnay, E. *Atmospheric Modelling, Data Assimilation and Predictability*; Cambridge University Press: Cambridge, UK, 2003.
17. Kazakova, E.; Rozinkina, I.; Chumakov, M. Verification of results of the working technology SNOWE for snow water equivalent and snow density fields determination as initial data for COSMO model. *COSMO Newsl.* **2016**, *16*, 25–36.
18. Chernykh, N.; Mikhalev, A.; Dmitriev, V.; Tynchenko, V.; Shutkina, E. Comparative Analysis of Existing Measures to Reduce Road Accidents in Western Europe. In Proceedings of the 2023 22nd International Symposium INFOTEH-JAHORINA, INFOTEH, Sarajevo, Bosnia and Herzegovina, 15–17 March 2023. [CrossRef]
19. Krasnopolsky, V.M.; Lin, Y. A neural network nonlinear multimodel ensemble to improve precipitation forecasts over continental US. *Adv. Meteorol.* **2012**, *2012*, 649450. [CrossRef]
20. Marzban, C. A neural network for post-processing model output: ARPS. *Mon. Wea. Rev.* **2003**, *131*, 1103–1111. [CrossRef]
21. Warner, T.T. *Numerical Weather and Climate Prediction*; Cambridge University Press: Cambridge, UK, 2010.
22. Ye, C.; Zhao, C.; Yang, Y.; Fermuller, C.; Aloimonos, Y. LightNet: A Versatile, Standalone Matlab-based Environment for Deep Learning. *arXiv* **2016**. Available online: https://arxiv.org/pdf/1605.02766.pdf (accessed on 1 December 2023).
23. Zurada, J.M. *Introduction to Artificial Neural Systems*; PWS: New York, NY, USA, 1992.
24. Goyal, M.; Goyal, R.; Lall, B. Learning Activation Functions: A New Paradigm of Understanding Neural Networks. *arXiv* **2019**, arXiv:1906.09529.
25. Burgers, G.; Van Leeuwen, P.J.; Evensen, G. Analysis scheme in the ensemble Kalman filter. *Mon. Weather. Rev.* **1998**, *126*, 1719–1724. [CrossRef]
26. Dey, R.; Chakraborty, S. Convex-hull & DBSCAN clustering to predict future weather. In Proceedings of the 2015 International Conference and Workshop on Computing and Communication (IEMCON), Vancouver, BC, Canada, 15–17 October 2015; pp. 1–8. [CrossRef]
27. Saima, H.; Jaafar, J.; Belhaouari, S.; Jillani, T.A. Intelligent methods for weather forecasting: A review. In Proceedings of the 2011 National Postgraduate Conference, Perak, Malaysia, 19–20 September 2011; pp. 1–6. [CrossRef]
28. Salman, A.G.; Kanigoro, B.; Heryadi, Y. Weather forecasting using deep learning techniques. In Proceedings of the International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 10–11 October 2015; pp. 281–285. [CrossRef]
29. Singh, N.; Chaturvedi, S.; Akhter, S. Weather forecasting using machine learning algorithm. In Proceedings of the 2019 International Conference on Signal Processing and Communication (ICSC), Noida, India, 7–9 March 2019; pp. 171–174. [CrossRef]

30. Sela, J.G. The Implementation of the Sigma Pressure Hybrid Coordinate into the GFS. Office Note (National Centers for Environmental Prediction (U.S.)). 2009; p. 461. Available online: https://repository.library.noaa.gov/view/noaa/11401 (accessed on 1 December 2023).
31. Weather Research & Forecasting Model (WRF) Mesoscale & Microscale Meteorology Laboratory. NCAR. Available online: https://www.mmm.ucar.edu/models/wrf (accessed on 1 December 2023).
32. Yonekura, K.; Hattori, H.; Suzuki, T. Short-term local weather forecast using dense weather station by deep neural network. In Proceedings of the IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 1683–1690. [CrossRef]
33. Bauer, P.; Thorpe, A.; Brunet, G. The quiet revolution of numerical weather prediction. *Nature* **2015**, *525*, 47–55. [CrossRef]
34. Buschow, S.; Friederichs, P. Local dimension and recurrent circulation patterns in long-term climate simulations. *arXiv* **2018**, *1803*, 11255. [CrossRef]
35. C3S: ERA5: Fifth Generation of ECMWF Atmospheric Reanalyses of the Global Climate, Copernicus Climate Change Service Climate Data Store (CDS). 2017. Available online: https://cds.climate.copernicus.eu/cdsapp#!/home (accessed on 7 June 2019).
36. Compo, G.P.; Whitaker, J.S.; Sardeshmukh, P.D.; Matsui, N.; Allan, R.J.; Yin, X.; Gleason, B.E.; Vose, R.S.; Rutledge, G.; Bessemoulin, P.; et al. The twentieth century reanalysis project. *Q. J. R. Meteor. Soc.* **2011**, *137*, 1–28. [CrossRef]
37. Coors, B.; Paul Condurache, A.; Geiger, A. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 518–533.
38. Dee, D.P.; Uppala, S.M.; Simmons, A.; Berrisford, P.; Poli, P.; Kobayashi, S.; Andrae, U.; Balmaseda, M.; Balsamo, G.; Bauer, P.; et al. The ERA-Interim reanalysis: Configuration and performance of the data assimilation system. *Q. J. R. J. R. Meteor. Soc.* **2011**, *137*, 553–597. [CrossRef]
39. Dueben, P.D.; Bauer, P. Challenges and design choices for global weather and climate models based on machine learning. *Geosci. Model Dev.* **2018**, *11*, 3999–4009. [CrossRef]
40. Faranda, D.; Messori, G.; Yiou, P. Dynamical proxies of North Atlantic predictability and extremes. *Sci. Rep.* **2017**, *7*, 41278. [CrossRef] [PubMed]
41. Faranda, D.; Messori, G.; Vannitsem, S. Attractor dimension of time-averaged climate observables: Insights from a low-order ocean-atmosphere model. *Tellus A* **2019**, *71*, 1–11. [CrossRef]
42. Fraedrich, K.; Jansen, H.; Kirk, E.; Luksch, U.; Lunkeit, F. The Planet Simulator: Towards a user-friendly model. *Meteorol. Z.* **2005**, *14*, 299–304. [CrossRef]
43. Freitas, A.C.M.; Freitas, J.M.; Todd, M. Hitting time statistics and extreme value theory. *Probab. Theory Rel.* **2010**, *147*, 675–710. [CrossRef]
44. Tynchenko, V.; Kurashkin, S.; Murygin, A.; Bocharov, A.; Seregin, Y. Software for optimization of beam output during electron beam welding of thin-walled structures. *Procedia Comput. Sci.* **2022**, *200*, 843–851. [CrossRef]
45. Kukartsev, V.V.; Tynchenko, V.S.; Bukhtoyarov, V.V.; Wu, X.; Tyncheko, Y.A.; Kukartsev, V.A. Overview of Methods for Enhanced Oil Recovery from Conventional and Unconventional Reservoirs. *Energies* **2023**, *16*, 4907. [CrossRef]
46. Krasnopolsky, V.M.; Fox-Rabinovitz, M.S. Complex hybrid models combining deterministic and machine learning components for numerical climate modelling and weather prediction. *Neural Netw.* **2006**, *19*, 122–134. [CrossRef]
47. Filina, O.A.; Sorokova, S.N.; Efremenkov, E.A.; Valuev, D.V.; Qi, M. Stochastic Models and Processing Probabilistic Data for Solving the Problem of Improving the Electric Freight Transport Reliability. *Mathematics* **2023**, *11*, 4836. [CrossRef]
48. Krasnopolsky, V.M.; Fox-Rabinovitz, M.S.; Belochitski, A.A. Using ensemble of neural networks to learn stochastic convection parameterisations for climate and numerical weather prediction models from data simulated by a cloud resolving model. *Adv. Artif. Neural Syst.* **2013**, *2013*, 485913. [CrossRef]
49. Lorenz, E.N. Deterministic nonperiodic flow. *J. Atmo. Sci.* **1963**, *20*, 130–141. [CrossRef]
50. Das, P.; Manikandan, S.; Raghavendra, N. Holomorphic aspects of moduli of representations of quivers. *Indian J. Pure Appl. Math.* **2019**, *50*, 549–595. [CrossRef]
51. Konyukhov, V.Y.; Oparina, T.A.; Zagorodnii, N.A.; Efremenkov, E.A.; Qi, M. Mathematical Analysis of the Reliability of Modern Trolleybuses and Electric Buses. *Mathematics* **2023**, *11*, 3260. [CrossRef]
52. Filina, O.A.; Tynchenko, V.S.; Kukartsev, V.A.; Bashmur, K.A.; Pavlov, P.P.; Panfilova, T.A. Increasing the Efficiency of Diagnostics in the Brush-Commutator Assembly of a Direct Current Electric Motor. *Energies* **2024**, *17*, 17. [CrossRef]
53. Frankle, J.; Carbin, M. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
54. Nooteboom, P.D.; Feng, Q.Y.; López, C.; Hernández-García, E.; Dijkstra, H.A. Using network theory and machine learning to predict El Nino. *Earth Syst. Dynam.* **2018**, *9*, 969–983. [CrossRef]
55. O'Gorman, P.A.; Dwyer, J.G. Using Machine Learning to Parameterize Moist Convection: Potential for Modeling of Climate, Climate Change, and Extreme Events. *J. Adv. Model. Earth Syst.* **2018**, *10*, 2548–2563. [CrossRef]
56. Golik, V.I.; Brigida, V.; Kukartsev, V.V.; Tynchenko, Y.A.; Boyko, A.A.; Tynchenko, S.V. Substantiation of Drilling Parameters for Undermined Drainage Boreholes for Increasing Methane Production from Unconventional Coal-Gas Collectors. *Energies* **2023**, *16*, 4276. [CrossRef]

57. Volneikina, E.; Kukartseva, O.; Menshenin, A.; Tynchenko, V.; Degtyareva, K. Simulation-Dynamic Modeling of Supply Chains Based on Big Data. In Proceedings of the 2023 22nd International Symposium INFOTEH-JAHORINA, INFOTEH 2023, Sarajevo, Bosnia and Herzegovina, 15–17 March 2023. [CrossRef]
58. Poli, P.; Hersbach, H.; Dee, D.P.; Berrisford, P.; Simmons, A.J.; Vitart, F.; Laloyaux, P.; Tan, D.G.H.; Peubey, C.; Thépaut, J.-N.; et al. ERA-20C: An Atmospheric Reanalysis of the Twentieth Century. *J. Clim.* **2016**, *29*, 4083–4097. [CrossRef]
59. Semenova, E.; Tynchenko, V.; Chashchina, S.; Suetin, V.; Stashkevich, A. Using UML to Describe the Development of Software Products Using an Object Approach. In Proceedings of the 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 1–4 June 2022. [CrossRef]
60. Scher, S. Toward Data-Driven Weather and Climate Forecasting: Approximating a Simple General Circulation Model with Deep Learning. *Geophys. Res. Lett.* **2018**, *45*, 12616–12622. [CrossRef]
61. Martyushev, N.V.; Malozyomov, B.V.; Sorokova, S.N.; Efremenkov, E.A.; Valuev, D.V.; Qi, M. Review Models and Methods for Determining and Predicting the Reliability of Technical Systems and Transport. *Mathematics* **2023**, *11*, 3317. [CrossRef]
62. Scher, S. Videos for Weather and climate forecasting with neural networks: Using GCMs with different complexity as studyground. *Zenodo* **2019**. [CrossRef]
63. Scher, S. Code and data for Weather and climate forecasting with neural networks: Using GCMs with different complexity as study-ground. *Zenodo* **2019**. [CrossRef]
64. Martyushev, N.V.; Malozyomov, B.V.; Kukartsev, V.V.; Gozbenko, V.E.; Konyukhov, V.Y.; Mikhalev, A.S.; Kukartsev, V.A.; Tynchenko, Y.A. Determination of the Reliability of Urban Electric Transport Running Autonomously through Diagnostic Parameters. *World Electr. Veh. J.* **2023**, *14*, 334. [CrossRef]
65. Scher, S.; Messori, G. Predicting weather forecast uncertainty with machine learning. *Q. J. R. Meteor. Soc.* **2018**, *144*, 2830–2841. [CrossRef]
66. Schneider, T.; Lan, S.; Stuart, A.; Teixeira, J. Earth System Modeling 2.0: A Blueprint for Models That Learn from Observations and Targeted High-Resolution Simulations. *Geophys. Res. Lett.* **2017**, *44*, 12396–12417. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Hybrid Model of Natural Time Series with Neural Network Component and Adaptive Nonlinear Scheme: Application for Anomaly Detection

Oksana Mandrikova * and Bogdana Mandrikova

Institute of Cosmophysical Research and Radio Wave Propagation, Far Eastern Branch of the Russian Academy of Sciences, Mirnaya St, 7, Paratunka, 684034 Kamchatskiy Kray, Russia; 555bs5@mail.ru
* Correspondence: oksanam1@mail.ru; Tel.: +7-919-436-1208

Abstract: It is often difficult to describe natural time series due to implicit dependences and correlated noise. During anomalous natural processes, anomalous features appear in data. They have a nonstationary structure and do not allow us to apply traditional methods for time series modeling. In order to solve these problems, new models, adequately describing natural data, are required. A new hybrid model of a time series (HMTS) with a nonstationary structure is proposed in this paper. The HMTS has regular and anomalous components. The HMTS regular component is determined on the basis of an autoencoder neural network. To describe the HMTS anomalous component, an adaptive nonlinear approximating scheme (ANAS) is used on a wavelet basis. HMTS is considered in this investigation for the problem of neutron monitor data modeling and anomaly detection. Anomalies in neutron monitor data indicate negative factors in space weather. The timely detection of these factors is critically important. This investigation showed that the developed HMTS adequately describes neutron monitor data and has satisfactory results from the point of view of numeric performance. The MSE model values are close to 0 and errors are white Gaussian noise. In order to optimize the estimate of the HMTS anomalous component, the likelihood ratio test was applied. Moreover, the wavelet basis, giving the least losses during ANAS construction, was determined. Statistical modeling results showed that HMTS provides a high accuracy of anomaly detection. When the signal/noise ratio is 1.3 and anomaly durations are more than 60 counts, the probability of their detection is close to 90%. This is a high rate in the problem domain under consideration and provides solution reliability of the problem of anomaly detection in neutron monitor data. Moreover, the processing of data from several neutron monitor stations showed the high sensitivity of the HMTS. This shows the possibility to minimize the number of engaged stations, maintaining anomaly detection accuracy compared to the global survey method widely used in this field. This result is important as the continuous operation of neutron monitor stations is not always provided. Thus, the results show that the developed HMTS has the potential to address the problem of anomaly detection in neutron monitor data even when the number of operating stations is small. The proposed HMTS can help us to decrease the risks of the negative impact of space weather anomalies on human health and modern infrastructure.

Keywords: nonstationary time series; neural networks; wavelet transform; correlated noise; anomaly detection; space weather

MSC: 62C12; 62L20; 68T05

1. Introduction

Information retrieval from natural data includes stages for the construction of models, methods and algorithms of analysis. The known problems in this field are nonstationary data and incomplete a priori knowledge on the information component and noise. This significantly complicates the process of model construction and methods for natural data analysis. In some critical related fields (physics and technique, biology, medicine, etc.),

such a problem results in the insufficient efficiency of existing methods. This is of high significance regarding the problems of natural anomaly detection (earthquakes, space weather anomalies, tsunamis, etc.) [1].

The object of research in this paper is the data from global network neutron monitors recording intensity variations (particles per a minute) in cosmic rays (CRs). Neutron monitor data time series contain regular cyclic components (22-year, 11-year, 27-day and solar-diurnal variations) and anomalous features of nonstationary structures having the form of bursts of different amplitudes and durations, series of spikes, etc. [2]. Generation factors of such anomalies are coronal mass ejections (CMEs), high-speed solar wind streams from coronal holes and anomalous processes in the near-Earth space. Neutron monitor data anomalies indicate negative factors of space weather such as occurrences of Forbush effects (sudden decreases and/or increases in cosmic ray intensity) or GLE events (ground-level enhancement events—strong anomalous increases in cosmic ray intensity on the Earth's surface) [3]. During such anomalies, radiation hazards for astronauts, airplane crews and passengers on polar routes increase. Space systems also undergo negative impacts due to technique losses. Thus, the near-real-time detection of anomalies in neutron monitor data is an important applied aspect of research [1].

The investigation of cosmic ray (CR) variations and the development of methods for their analysis began in the first half of the 20th century and has continued through today. One of the most successful and known methods for the investigation of CR parameters is the global survey method [4]. This method includes the method of CR variation communication functions, the method of particle trajectory calculations and spherical analysis for the detection of particle significant spherical harmonics [4]. The global survey method makes it possible to estimate CR flux characteristics with satisfactory accuracy in the case of an uninterrupted operation of a certain number of recording stations. However, due to calculation complexity, the method cannot be realized in automatic mode and it is not effective for the detection of initial proton increases.

Threshold algorithms, used by the Australian Space Weather Service [5] and GLE Alert system [6] to analyze CR variations, make short-term forecasts of radiation hazards according to the data of neutron monitors in an on-line regime. However, the efficiency of these algorithms is low due to their insensitivity to low-amplitude anomalies. The investigations in [7] showed that the GLE Alert algorithm can give unreliable results. Application of this algorithm for 3 years did not allow identification of more than 50% of solar proton events.

Thus, the applied methods for neutron monitor data analysis do not satisfy modern requirements and new approaches are needed in this field [8].

A high proportion of uncertainty in knowledge, the significant nonstationarity of neutron monitor data and noise correlation make it difficult to apply classical methods for time series analysis (ARIMA models, methods for time series decomposition, etc.) and result in low efficiency [9]. At the present time, methods of artificial intelligence and machine learning are actively applied in different applied fields to solve these problems. For example, the authors of [10,11] suggested using neural networks to forecast the state of oil pipelines. The approach in [10], based on the application of neural networks of direct propagation, made it possible to forecast the conditions of oil pipeline exploitation and to classify metal loss defects. In the research in [12], the authors proposed using neural networks with Bayesian regularization in order to solve this problem. A new approach was suggested in [12] to forecast the operation capability of a dry gas transmission system and to classify whether the amount of metal loss was effective even in the case of the absence of a priori data. For the problem of the diagnostics of rotating machine faults, a group of researchers [13] developed a model of a neural network with deep momentum transfer (SNN). The investigations in [13] showed that the developed method allows one to carry out contactless processing of visual data event fluxes, distinguish features and make diagnostics of machine faults.

In the field of this investigation, the author of [14] studied the possibility of application of graph neural networks to investigate the CR data energy spectrum. The neural network approach in [14] made it possible to decrease the time and computational efforts and showed more accurate results compared to the application of the likelihood function. However, the method results in [14] significantly depend on CR primary energy and on the configuration features of an applied detector.

The authors of [15] suggested a method for analysis of the arrival directions of super-high energy cosmic ray fluxes using a deep convolutional neural network. The results in [15] showed that the suggested method is more effective and more sensitive than the approach based on the angle power spectrum. The advantages of the method are the low generalization and efficiency decreases of the neural network when test models deviate from those used for training.

At the present time, hybrid models are actively developing to solve the problems indicated above. They combine classical statistical methods for data analysis and new developments in the fields of artificial intellect, machine learning and signal digital processing. The synthesis of mathematical techniques with adaptive modern tools allows one to extend the capabilities of applied constructs and to improve their efficiency [8,16–22]. For example, the paper [17] suggests a hybrid model based on the combination of the method of empirical mode decomposition (EMD) and the deep LSTM neural network. The approach proposed in [17] makes it possible to forecast the time series of climatic indexes and solar spots with long-term periodic behavior. The forecast changeability (or uncertainty) is suggested to be detected on the basis of the combination of the EMD method and the K-nearest neighbor. The paper [21] proposes an approach to the forecast of total electron content in the ionosphere (ionospheric parameter). This approach includes a combination of methods for ensemble empirical mode decomposition, K-averages and the self-service LSTM neural network. According to the estimates, the model suggested in [21] has a higher performance compared to some typical forecast methods applied to this problem. A hybrid model of a time series is suggested in [22]. It includes ARIMA components and multiple-scale wavelet analysis (MSA) components. The combination of the time series classical models with the MSA allowed one to obtain the adaptive model of an ionospheric parameters time series.

In this investigation, we propose a new hybrid model of a time series (HMTS) with nonstationary structure. It includes a neural network component and a nonlinear adaptive approximating scheme. The HMTS neural network component describes the data's regular time variation and its parameters are estimated on the basis of an autoencoder network. The autoencoder is a nonlinear method of main components. It allows us to approximate dependences of a priori unknown structure and suppress noise. Today the autoencoder is successfully applied to solve different applied tasks. For example, in the paper [23] the authors suggested a federated semi-supervised method for the diagnostics of data transfer errors called targeted transfer learning through distribution barycenter medium (TTL-DBM). The application of the autoencoder network in [23] made it possible to aggregate key data distribution parameters and generate a distribution barycenter in an intermediate link for federated adaptation.

We applied the autoencoder network for the first time to approximate the regular time variation of neutron monitor data in the investigation in [24]. In the same paper [24] the autoencoder efficiency was shown for anomaly detection based on the search for the change points in a system. For the first time, we considered application of nonlinear approximating schemes for detection of anomalies in CR variations in [25]. The results in [25] showed the prospects of this approach for detection of multi-scale anomalies. Moreover, the possibility of application of both the autoencoder and the nonlinear approximating schemes was considered in the papers [24,25]. The investigations in [24,25] showed that the autoencoder is more effective for the detection of narrow-band anomalies, and the nonlinear approximating scheme is more effective for the detection of short-period different-scale anomalies. In the following research [26], we compared the method of singular spectral analysis (SSA) with the autoencoder and adaptive threshold filtering (AADA algorithm). The study [26]

showed that the SSA can be applied to detect CR variation components during the process of dynamics analysis. However, the autoencoder and adaptive threshold wavelet filtering are more effective for anomaly detection. This paper continues these investigations. The combination of the autoencoder with the nonlinear approximating scheme in the form of the hybrid model of a time series with nonstationary structure develops that theory. This paper presents a formal form of the HMTS, and the methods to estimate its parameters have been developed. In order to optimize the calculation of the HMTS anomalous component, the likelihood ratio test was applied. The analytical expressions of threshold function parameters were obtained. They minimize data approximation errors. Moreover, the wavelet basis selection criterion, when constructing a nonlinear approximating scheme, was proposed. This allowed us to improve the HMTS efficiency for anomaly detection. Based on statistical modeling, the HMTS's high efficiency was proved for the problem of anomaly detection in neutron monitor data.

2. Materials and Methods

2.1. Hybrid Model of a Natural Time Series of a Nonstationary Structure

A natural time series of a nonstationary structure can be represented in the following form:

$$f(t) = R(t) + A(t) + e(t) = \sum_r S_r(t, \theta_r) + \sum_m a[m]\langle f, g_m\rangle g_m + e(t), \quad (1)$$

where $R(t) = \sum_r S_r(t, \theta_r)$ is the regular component, which has a parametric form and is a linear combination of the components $S_r(t, \theta_r)$, r is the component number, $\theta = \{\theta_r\}$ is the parameter set, $A(t) = \sum_m a[m]\langle f, g_m\rangle g_m$ is the nonstationary component, which includes local features of different forms and durations and is represented in the form of a nonlinear approximating scheme [27], in which $a[m]$ depends on $\langle f, g_m\rangle$, g_m is the orthonormal basis, $\langle\cdot\rangle$ is the scalar product, t is the time.

2.2. Determination of the Regular Component $R(t)$ of Model (1)

The regular component $R(t)$ of model (1) includes implicit complex regularities. Thus, we represent it by means of the autoencoder neural network. The autoencoder is a nonlinear method of main components, and it makes it possible to determine the form of hidden regularities and estimate their parameters [28]. In terms of the autoencoder, we obtain the following estimate for the regular component $R(t)$ of model (1):

$$\hat{R}(t) = \sum_r S_r(t, \theta_r) = Z\left(\theta_r{}^{(2)}\left(H\left(\theta_r{}^{(1)} f(t)\right)\right)\right), \quad (2)$$

where Z, H are the activation functions of the neural network for output and inner layers, respectively, $\theta = \left\{\theta_r{}^{(i)}\right\}_{i=1,2}$ are the parameters, $\theta_r{}^{(i)}$ are the weight matrix of the i-th layer.

From ratio (2) we estimate the parameters $\hat{\theta} = \left\{\hat{\theta}_r^{(i)}\right\}_{i=1,2}$ of model (1) based on minimizing the cost function [28].

$$C_{NET} = \sum_{k=1}^{l} \left\|\hat{R}(t) - f^{(k)}(t)\right\|^2; \hat{\theta} = argmin_{\theta\in\Theta} \sum_{k=1}^{l} \left\|\hat{R}(t) - f^{(k)}(t)\right\|^2, \quad (3)$$

where $f^{(k)}(t)$ is the k-th example (data sampling), Θ is the parameter set.

To verify the adequacy of the estimated regular component $\hat{R}(t)$ (ratio (2)), for example, Ljung–Box Q-statistics can be applied [29]. The Q-statistics tests the hypothesis on the absence of model residuals' autocorrelation. In order to test the normality of the model residuals' distribution, the Jarque–Bera test can be used [29].

2.3. Determination of the Anomalous Component $A(t)$ of Model (1)

The anomalous component $A(t)$ of model (1) has a nonstationary structure and is represented in the form of the nonlinear approximating scheme (1).

$$A(t) = \sum_m a[m]\langle f, g_m\rangle g_m, \tag{4}$$

where it is assumed that $f \in L^2(R)$, $L^2(R)$ is the Lebesgue space.

For nonparametric representation (4), the problem of estimation of the component $A(t)$ parameters is reduced to the filtering problem, which consists of the construction of the operator D_A minimizing the risk:

$$\mathcal{R}(D_A f, A) = \inf_{D_A \in O} E\left\{\|D_A f - A\|^2\right\}, \tag{5}$$

where E is the mathematical expectation, $\|\cdot\|$ is the norm in $L^2(R)$, O is the operator set.

In ratio (4), the nonlinear diagonal operator is considered as the operator D_A [30], thus, risk (5) is determined by the following value [27]:

$$\mathcal{R}(D_A f, A) = E\left\{\left\|\hat{A} - A\right\|^2\right\} = \sum_{m=1}^{N} E\left\{\left|\langle A, g_m\rangle - \langle f, g_m\rangle a[m]\right|^2\right\}, \tag{6}$$

N is the series length.

Then, we assume that

$$f(t) = A(t) + e_A(t), \tag{7}$$

where the noise $e_A(t)$ is correlated and the noise coefficient dispersion $\sigma_m^2 = E\left\{\left|\langle e_A, g_m\rangle\right|^2\right\}$ depending on g_m, the mathematical expectation of noise $E\{e_A(t)\} = 0$. Then, from ratio (7), the risk $\mathcal{R}(D_A f, A)$ (6) is minimum when [27]

$$a[m] = \frac{\left|\langle A, g_m\rangle\right|^2}{\left|\langle A, g_m\rangle\right|^2 + \sigma_m^2}. \tag{8}$$

If we assume that $a[m] \in \{0, 1\}$, the operator D_A, minimizing the risk in (8), can be determined as [30]

$$a[m] = \begin{cases} 1, if\left|\langle A, g_m\rangle\right| \geq h \times \sigma_m \\ 0, if\left|\langle A, g_m\rangle\right| < h \times \sigma_m \end{cases}, h = \sqrt{2lnN}. \tag{9}$$

Taking into account (9), the risk [30]

$$\mathcal{R}(D_A f, A) = \mathcal{R}_{\sigma_m}(A) = E\left\{\|D_A f - A\|^2\right\} = \sum_{m=0}^{N} min\left(\left|\langle A, g_m\rangle\right|^2, \sigma_m^2\right). \tag{10}$$

Then, applying the threshold estimate in the orthonormal wavelet basis $G_\Psi = \left\{\Psi_{k,n}\right\}_{(k,n)\in Z^2}$, $\Psi_{k,n}(t) = 2^{\frac{k}{2}}\Psi(2^k t - n)$, the operator D_{A_Ψ}, minimizing the risk, is [31]

$$\hat{A}(t) = D_{A_\Psi} f(t) = \sum_k \sum_{n=1}^{N} P_{\sigma_{k,n}}\left(\left\langle f, \Psi_{k,n}\right\rangle\right)\Psi_{k,n}(t), \tag{11}$$

where $P_{\sigma_{k,n}}\left(\left\langle f, \Psi_{k,n}\right\rangle\right) = \begin{cases} \left\langle f, \Psi_{k,n}\right\rangle, if\left|\left\langle f, \Psi_{k,n}\right\rangle\right| \geq h' * \sigma_{k,n} \\ 0, if\left|\left\langle f, \Psi_{k,n}\right\rangle\right| < h' * \sigma_{k,n} \end{cases}$, $\sigma_{k,n}^2 = E\left\{\left|\left\langle e_{A_\Psi}, \Psi_{k,n}\right\rangle\right|^2\right\}$.

The operator D_{A_Ψ} in ratio (11) determines the nonlinear approximating scheme in the wavelet basis.

Note. In ratio (11) the threshold dependence $T_{k,n} = h' * \sigma_{k,n}$ is considered both on the space variable n (time) and on the scale k. This makes it possible to optimize the threshold

both to time and scale changes of data. Moreover, as is shown below, in this case, thresholds close to optimal ones can be found based on the ratio criterion [32].

When joining the expression in (2) and (11), we obtain the hybrid model of a nonstationary time series (HMTS):

$$\begin{array}{l} f(t) = R(t) + A(t) + e(t) = \sum\limits_{r} S_r(t,\theta_r) + \sum\limits_{m} a[m]\langle f, g_m\rangle g_m + e(t) = \\ = Z\left(\theta_r{}^{(2)}\left(H\left(\theta_r{}^{(1)} f(t)\right)\right)\right) + \sum_k \sum_{n=1}^{N} P_{\sigma_{k,n}}\left(\langle f, \Psi_{k,n}\rangle\right)\Psi_{k,n}(t) + e(t), \end{array} \quad (12)$$

where Z is the linear function of the network output layer activation, H is the signoidal function of the network inner layer activation, $\theta = \left\{\theta_r{}^{(i)}\right\}_{i=1,2}$ are the parameters of the neural network determined in ratio (3), $P_{\sigma_{k,n}}(\cdot)$ is the threshold function with the thresholds determined in ratio (11).

2.4. Optimization of the HMTS. Determination of the Wavelet Basis Giving the Least Errors

From ratio (10), the risks of the estimates $\hat{A}(t)$ on the wavelet basis can be limited by the value

$$\mathcal{R}_\Psi\left(D_{A_\Psi} f, A\right) = E\left\{\left\|D_{A_\Psi} f - A\right\|^2\right\} = \sum_{k,n} min\left(\left|\langle A, \Psi_{k,n}\rangle\right|^2, \sigma_{k,n}^2\right) \le \sum_{k,n} \sigma_{k,n}^2. \quad (13)$$

Ratio (13) makes it possible to control the risk of the estimate $\hat{A}(t)$ based on noise dispersion $\sigma_{k,n}^2$. Moreover, having estimated the risk for different bases from the dictionary Ω and using (13), we can determine the wavelet basis giving the least losses.

$$\Psi^{op}\colon\ \mathcal{R}_{\Psi^{op}}\left(D_{A_\Psi} f, A\right) = \min_{\Psi\in\Omega} \mathcal{R}_\Psi\left(D_{A_\Psi} f, A\right). \quad (14)$$

2.5. Optimization of the Estimate of the Model Anomalous Component $\hat{A}(t)$ by the Ratio Criterion

Derivation of an optimal estimate (10) is based on the assumption that $E\{e_A(t)\} = 0$, which for the correlated noise σ_m is not always realized. However, the function mapping into wavelet space gives an important advantage, which follows from the Jaffar theorem [27]: an increase in the amplitude of wavelet coefficients $\left|\langle f, \Psi_{k,n}\rangle\right|$ indicates the appearance of a local feature of the function $f(t)$ in the vicinity of the point $t = n$; out of the neighborhood, containing local features, wavelet coefficient amplitudes satisfy the condition

$$\left|\langle f, \Psi_{k,n}\rangle\right| \le Qk^{q+0.5} = \Pi_k, \quad (15)$$

$Q = const > 0$, q is the Lipschitz index.

Thus, it follows from (15) that the values $\left|\langle f, \Psi_{k,n}\rangle\right| = \left|\langle e_{A_\Psi}, \Psi_{k,n}\rangle\right|$ with respect to the argument n are close to zero. Thus, the following condition is provided in the wavelet space: $E\left\{e_{A_\Psi}(t)\right\} = 0$. The $\left|\langle f, \Psi_{k,n}\rangle\right| = \left|c_{k,n}\right|$ value's deviation from zero is the result of noise effect in this case (Figure 1).

Then, using the ratio criterion [32], we assume that there is a local feature in the vicinity of the point $t = n$ if

$$\frac{W\left(\left|c_{k,n}\right|\Gamma_1\right)}{W\left(\left|c_{k,n}\right|\Gamma_0\right)} \ge T_{k,n}, \quad (16)$$

where $W\left(\left|c_{k,n}\right|\Gamma_0\right)$ and $W\left(\left|c_{k,n}\right|\Gamma_1\right)$ are the absolute value probability densities for the coefficients $\left|c_{k,n}\right|$ when a local feature is absent or is present, respectively. In its turn, following the Neyman–Pearson criterion [32], the threshold

$$T_{k,n} = T_{\alpha,k,n}\ (\alpha \text{ is the probability of } I\text{-kind error}) \quad (17)$$

is determined from the condition $\int_\Phi^0 W\left(\left|c_{k,n}\right|\Gamma_0\right) d\left|c_{k,n}\right| = \alpha$, where Φ is the critical region (Figure 1).

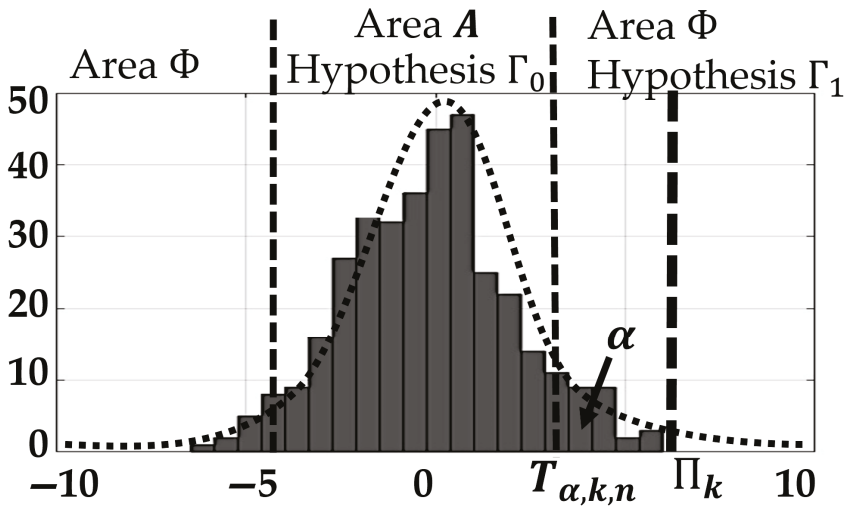


Figure 1. Histogram of wavelet coefficients $c_{k,n}$ for $k = 4$, constructed on the basis of neutron monitor data, which do not contain local features.

Thus, from (11), (16), (17) we obtain the optimized estimate of the model anomalous component in the form of the nonlinear adaptive approximating scheme (NAAS):

$$\hat{A}(t) = D_{A_\Psi} f(t) = \sum_k \sum_{n=1}^{N} P_{T_{\alpha,k,n}}(c_{k,n}) \Psi_{k,n}^{op}(t), \tag{18}$$

where $P_{T_{\alpha,k,n}}(c_{k,n}) = \begin{cases} c_{k,n}, if \left|c_{k,n}\right| \geq T_{\alpha,k,n} \\ 0, if \left|c_{k,n}\right| < T_{\alpha,k,n} \end{cases}$, $T_{\alpha,k,n} = t_{\frac{1-\alpha}{2},\, M} \hat{\sigma}_{k,n}$, $t_{\frac{1-\alpha}{2},\, M}$ is the quantile of the level $\frac{1-\alpha}{2}$.

As the result, from (12), (18) we have the optimized hybrid model on a nonstationary time series (HMTS):

$$\begin{aligned} f(t) = R(t) + A(t) + e(t) = \sum_r S_r(t, \theta_r) + \sum_m a[m] \langle f, g_m \rangle g_m + e(t) = \\ = Z\left(\theta_r^{(2)}\left(H\left(\theta_r^{(1)} f(t)\right)\right)\right) + \sum_k \sum_{n=1}^{N} P_{T_{\alpha,k,n}}(c_{k,n}) \Psi_{k,n}^{op}(t) + e(t), \end{aligned} \tag{19}$$

Z, H are the activation functions, $\theta = \left\{\theta_r^{(i)}\right\}_{i=1,2}$ are the model regular component parameters, $P_{T_{\alpha,k,n}}(c_{k,n})$ are the threshold functions with the thresholds $T_{\alpha,k,n} = t_{\frac{1-\alpha}{2},\, M} \hat{\sigma}_{k,n}$, $\Psi_{k,n}^{op}(t) = 2^{\frac{k}{2}} \Psi^{\text{оп}}(2^k t - n)$ is the wavelet.

3. Construction of the HMTS for Neutron Monitor Data

In the investigation, data from Nain, Inuvik, Oulu, Tule and South Pole stations were used [33]. The choice of the stations was determined by the experience of the experts from the applied subject field [4]. NM data record cosmic ray intensities (particles per minute). In the HMTS terms (representation (19)), NM data time variation can be represented as

$$f(t) = R(t) + A(t) + e(t) = Z\left(\theta_r^{(2)}\left(H\left(\theta_r^{(1)} f(t)\right)\right)\right) + \sum_k \sum_{n=1}^{N} P_{T_{\alpha,k,n}}(c_{k,n}) \Psi_{k,n}(t) + e(t), \tag{20}$$

where the regular cyclic component $R(t) = Z\left(\theta_r^{(2)}\left(H\left(\theta_r^{(1)} f(t)\right)\right)\right)$ includes periodicities of different amplitude and duration, shifts, spikes etc. (27-day and solar-diurnal, seasonal variations, etc.).

The nonstationary (anomalous) component $A(t) = \sum_k \sum_{n=1}^{N} P_{T_{\alpha,k,n}}(c_{k,n}) \Psi_{k,n}(t)$ contains different-scale anomalous features in the form of bursts, spikes of different amplitude

and duration, etc., which characterize the occurrences of Forbush effects and GLE events and are determined by anomalous processes on the Sun and in the near-Earth space; $e(t)$ is the noise.

The regular component of the NM data in the terms of the typical autoencoder neural network has the following form (see Formula (20)).

$$\hat{R}(t) = Z\left(\theta_r{}^{(2)}\left(H\left(\theta_r{}^{(1)}f(t) + b^{(1)}\right)\right) + b^{(2)}\right), \quad (21)$$

where the upper index (i) is the layer number, H is the nonlinear activation function of the encoder (sigmoid), $\theta_r{}^{(1)}$ is the weight matrix of the network's first layer, Z is the linear activation function of the decoder (inner layer), $\theta_r{}^{(2)}$ is the weight matrix of the second layer, $b^{(i)}$ are the shift vectors.

Taking into account the local natural factor effect, neural networks for each station were trained separately. The neural networks were trained using the data from high-latitudinal and polar ground stations of neutron monitors. The network architecture for each station was typical [28] and included two layers. Taking into account data on diurnal variation, the network output vector length was 1440 counts (minute sampling). The network hidden layer length was determined empirically and was equal to 720 counts. The neural networks were trained on the basis of backpropagation, taking into account the application of a sparsity regularizer [28]. The training samples were formed from the neutron monitor data for 2013–2015 during the periods of absence of anomalous processes in the near-Earth space (calm periods). The calm periods were determined by the space weather data [34,35]. Solar and diurnal cycles were also taken into account during the selection.

Table 1 presents the results of the estimates of the model's regular component $\hat{R}(t)$ adequacy (Formula (21)). Data for the calm periods were used, and the mean squared error (MSE) was estimated. The results in Table 1 show that the MSE values are close to zero. We should note that the MSE during high solar activity (SA) significantly exceeded that during low solar activity. The highest MSE was obtained from the South Pole station data that is likely to be associated with its geographical location (the South Pole station is polar and the rest of the stations are high-latitudinal). The results from Table 1 confirm the adequacy of the estimated regular component $\hat{R}(t)$ of the model.

Table 1. Estimate of the HMTS regular component $\hat{R}(t)$ adequacy.

$\frac{1}{N}\sum_{t=1,N}\left(f(t)-\hat{R}(t)\right)^2$	**Mean Squared Error**	
	High SA (2013–2015)	**Low SA (2018–2020)**
Inuvik st.	0.20418	0.17003
Nain st.	0.17231	0.15162
Tule st.	0.20212	0.16181
Oulu st.	0.19323	0.13104
South Pole st.	0.24245	0.19006

The model's $\hat{R}(t)$ adequacy was verified using the Ljung–Box Q-statistics and the Jarque–Bera test [29]. The Ljung–Box Q-statistic (Q_{stat}) verifies the hypothesis on the joint equality of all the autocorrelations of the model residuals series up to the order m inclusive. The Jarque–Bera fit test (JB) was applied to verify the normality of the model residuals' distribution. The test results are illustrated in Table 2. The results in Table 1 indicate that the estimated model $\hat{R}(t)$ adequately describes the NM data's regular time variation, and the errors are the white Gaussian noise (column 1, Table 2). We should note that during the anomalies, the network errors significantly increase (column 2, Table 2). During the research, we also tried to train the network on the data containing anomalies (column 4, Table 2). However, due to the great diversity of anomaly forms, it is impossible to obtain an

adequate model and to estimate its parameters. This fact confirms the necessity of taking into account the HMTS nonstationary (anomalous) component.

Table 2. Results of the neural network performance.

	Results of a Network Trained on Data without Anomalies	**Results of a Network Trained on Data without Anomalies**	**Results of a Network Trained on Data with Anomalies**
NN input	Data without anomalies $JB = 4.44$; $Q_{stat} = 54.75$;	Data with anomalies $JB = 104.61$; $Q_{stat} = 3.85 \times 10^4$	Data with anomalies $JB = 104.61$; $Q_{stat} = 3.85 \times 10^4$
NN output	$JB = 2.77$; $Q_{stat} = 3.2 \times 10^3$;	$JB = 209.95$; $Q_{stat} = 1.428 \times 10^4$	$JB = 225.86$; $Q_{stat} = 2.11 \times 10^4$
NN errors	$JB = 2.33$; $Q_{stat} = 30.23$	$JB = 232.45$; $Q_{stat} = 8.836 \times 10^3$	$JB = 68.64$; $Q_{stat} = 9.25 \times 10^4$

The anomalous component $A(t)$ parameters of the model were estimated using the NM data during the periods containing anomalous events in cosmic rays (Forbush effects) and observed during the disturbances in the near-Earth space (magnetic storms). The data were transformed by the discrete wavelet transform and threshold functions (see Formula (20)):

$$f(t) = \sum_{k=0}^{K} \sum_{n=1}^{N} P_{T_{\alpha,k,n}}(c_{k,n}) \Psi_{k,n}(t), \tag{22}$$

where $\Psi_{k,n}(t) = 2^{\frac{k}{2}} \Psi\left(2^k t - n\right)$ are basic wavelets, $n,\ k \in N, c_{k,n} = \left\langle f(t), \Psi_{k,n} \right\rangle$ are the coefficients of the $f(t)$ function's decomposition into a series, $P_{T_{\alpha,k,n}}(c_{k,n}) = \begin{cases} c_{k,n}, if \left|c_{k,n}\right| \geq T_{\alpha,k,n} \\ 0, if \left|c_{k,n}\right| < T_{\alpha,k,n} \end{cases}$ is the threshold function, N is the signal length, K is the largest scale.

The thresholds $T_{\alpha,k,n} = t_{\frac{1-\alpha}{2},\ M} \hat{\sigma}_{k,n}$ (Formula (22)) where determined with the significance level $\alpha = 0.05$ (Neyman–Pearson criterion). The standard deviation $\hat{\sigma}_{k,n} = \sqrt{\frac{1}{M} \sum_{n=1}^{M} (c_{k,n} - \overline{c_{k,n}})^2}$ was estimated in the time window of the length $M = 1440$ (determined by the solar-diurnal cycle). Orthonormal wavelets of Coiflets and Daubechies families were used as the basic wavelets. Table 3 shows the results of the estimates of the NM data approximation errors using different wavelets (Formula (13)). The estimates show that the least error was obtained when using the Coiflet 2 and Daubechies 2 wavelets. When selecting the wavelet basis, the carrier size and the wavelet smoothness order were taken into consideration besides the errors. It is known [27] that the carrier size determines the vicinity dimensions containing the boundary effect. The smoothness order characterizes the capability of a wavelet to detect high-order features. Taking into account the wavelet properties indicated above, the Coiflet 2 wavelet was determined as the best one.

Table 3. Estimates of the data approximation error using different wavelets.

$\sum_{k,n} \sigma_{k,n}^2 / \|f\|$		**Wavelets**				
		coif_1	**coif_2**	**coif_3**	**db_1**	**db_2**
st. Inuvik	High SA	0.0175	0.0170	0.0176	0.0178	0.0174
	Low SA	0.0177	0.0175	0.0174	0.0178	0.0177
st. Oulu	High SA	0.0189	0.0187	0.0188	0.0190	0.0189
	Low SA	0.0202	0.0202	0.0201	0.0203	0.0200
Carrier size		6	12	18	2	4
Smoothness order		2	4	6	1	2

4. Application of the HMTS to Detect Anomalies

As illustrated above (column 2, Table 2), data time variation is disturbed during anomaly occurrences, and, as a consequence, modeling errors increase. Thus, detection of anomalies in NM data can be based on data modeling using the autoencoder (Formula (13)) and on the determination of the periods of neural network error increases. Then, to estimate anomaly parameters, the nonlinear approximating scheme on the wavelet basis was constructed (Formula (14)). The anomaly intensity was estimated from wavelet coefficient absolute values as

$$E_n = \sum_{k=0}^{K} P_{T_{\alpha,k,n}}(c_{k,n}), \tag{23}$$

where $K = 250$ is the largest scale analyzed.

The control-flow chart realizing the proposed method is illustrated in Figure 2.

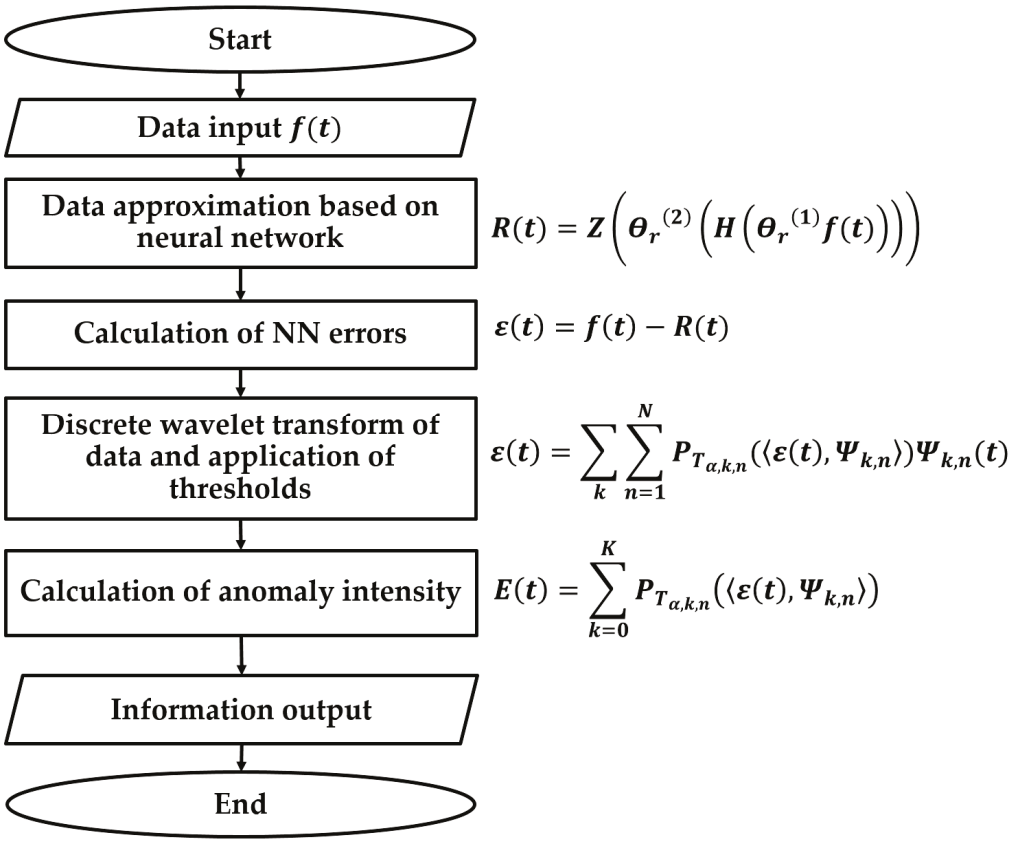


Figure 2. The control-flow chart for anomaly detection.

As an example, Figure 3 shows the results of the NM data modeling at Oulu station during an anomaly occurrence. Analysis of the results in Figure 3 shows the neural network error increase during the anomalous period that confirms the method's efficiency.

The method's efficiency was estimated using the neutron monitor data for 2013–2023. The data samples were formed for the periods containing the anomalies in cosmic ray variations (anomalous periods). The statistical modeling was performed. Both initial natural data and model data, formed on the basis of the natural data, were used in the estimates. The model data were formed using the wavelet packet operations [27] as follows:

(1) the data time series with the length of 1440 counts (diurnal variations) were formed, each count was equal to the corresponding median value of the neutron monitor's initial data during calm periods;

(2) the obtained time series were decomposed into wavelet packets up to the 7th scale level (using Coiflet 1 wavelet; the 7th scale level was determined taking into account the series length);

(3) wavelet reconstruction of the time series' smoothed components was carried out not taking into account the details (X^{trend});

(4) local features of triangle-pulse form and the Gaussian-modeled pulse, which had different amplitude and duration (X^{anom}), were added to the obtained time series;

(5) additive correlated noise (X^{noise}) was added to the formed time series.

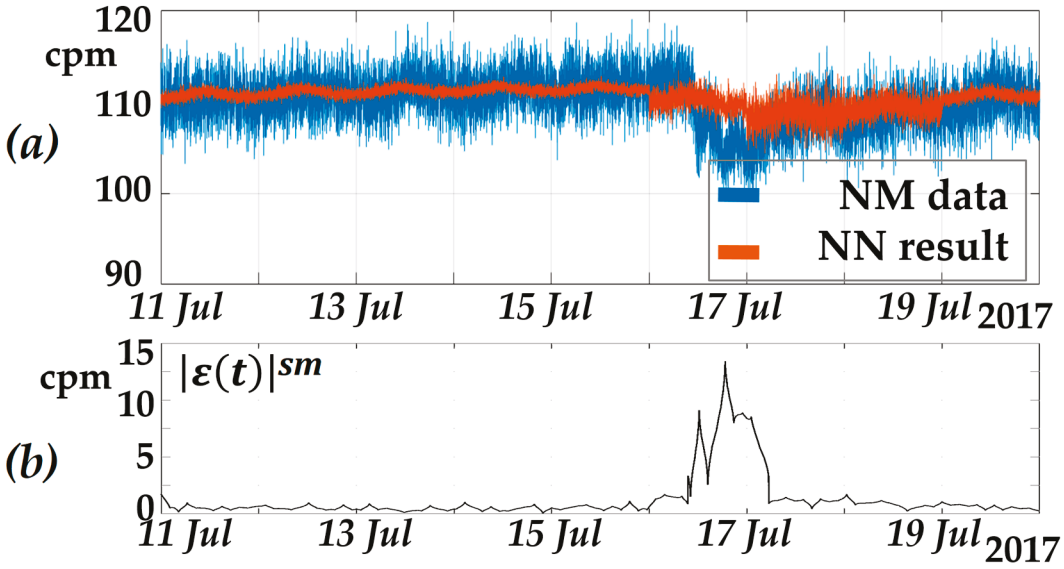


Figure 3. Processing results: (**a**) NM data are in blue, NN error is in orange; (**b**) smoothed $\left|\varepsilon(t)\right|^{sm}$.

The graphs of anomaly detection probabilities, depending on anomaly amplitude (signal–noise ratio) and on anomaly duration, are illustrated in Figure 4. The analysis of the results shows that the detection probability for the anomaly, having the duration of 20 counts, is more than 80% for the signal–noise ratio of 1.5 (at the false alarm rate $\alpha = 0.05$). When the anomaly is more than 60 counts, the detection probability is close to 90% for the signal–noise ratio of 1.3. The results show the high accuracy of the method.

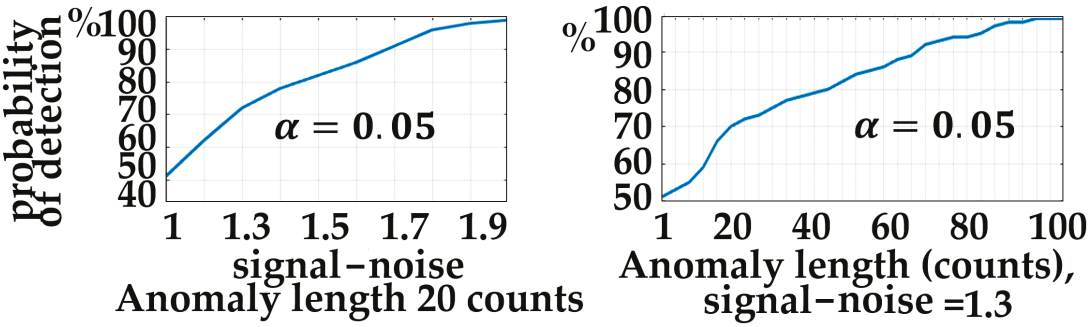


Figure 4. Graphs of the anomaly detection probabilities.

Figure 5 shows the results of the detection of a low-amplitude anomaly (signal–noise ratio is 1.2) in the model signal against the background of correlated noise (pink noise was added). We should note that the anomaly in the noised model signal (Figure 5b) is not observed visually. The obtained result (Figure 5d) (see Formula (23)) confirms the high accuracy of the method for detection of anomalies including low-amplitude anomalies.

Figure 6 shows the results of the method (Figure 6d,e) on the model data with correlated (pink) noise (Figure 6b). In order to make the comparison, the results of application of

a continuous wavelet transform (CWT) are illustrated (Figure 6c,d). The CWT results agree well with the method results. However, due to the correlated noise effect (including the solar-diurnal cycle), it is impossible to detect an anomaly in the data based on the CWT.

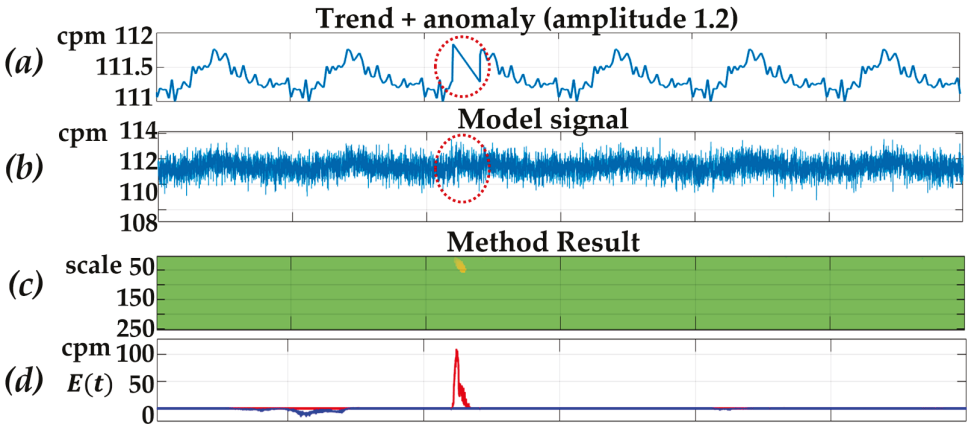


Figure 5. Processing results: (**a**) trend of the model signal with anomaly; (**b**) model signal; (**c**,**d**) results of the method with model data.

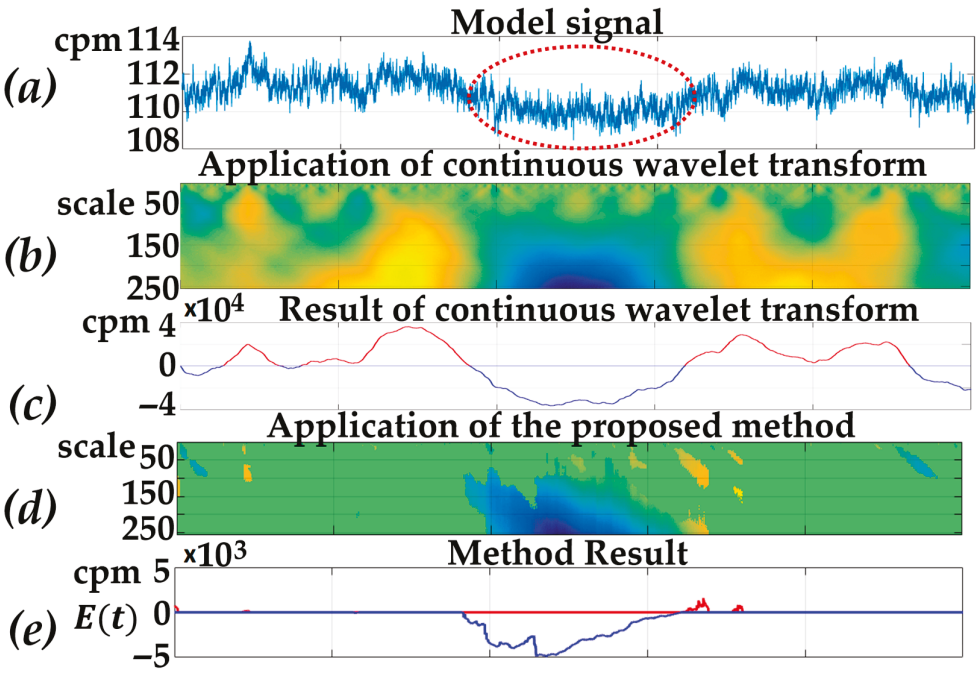


Figure 6. Processing results: (**a**) model signal; (**b**,**c**) results of the CWT application; (**d**,**e**) results of the suggested method.

Figure 7 illustrates the results of the suggested method using the NM data from Oulu station (Figure 6a) and using the model data (Figure 7e). Based on the data [34], the Forbush effect was recorded at 05:59 UT on 16 July 2017. The magnetic storm [35] was recorded at Novosibirsk station at 06:00 UT on 17 July 2017. The period, containing anomalous changes, is marked by a red dashed-line oval. The results show that application of the suggested method makes it possible to detect anomalies both in natural (Figure 7c,d) and in model data (Figure 7g,h).

Figure 8 illustrates the results of the suggested method with the NM data from different stations. An example, containing the period of Forbush decrease in cosmic ray variations (Figure 8 on the left) and the period containing an anomalous decrease in cosmic ray intensity (Figure 8 on the right), is shown. The times of the magnetic storm commencements are indicated in Figure 8 by yellow lines. The time of the Forbush decrease according to the

data [34] (global survey method was applied) is indicated in Figure 8 by a red line. The results in Figure 8 illustrate that the proposed method made it possible to detect the anomalies in cosmic ray variations timely. We should also note that the Forbush decrease occurrence was determined by the suggested method much earlier (Figure 8 on the left, marked by blue) compared to the global survey method (Figure 8 on the left, marked by a red line). Moreover, the comparison of the results from different stations shows the high sensitivity of the HMTS method and the ability to detect anomalies by the data from each separate station. That indicates high efficiency of the method and the possibility to minimize the number of engaged stations when detecting anomalies in cosmic ray variations. This is an important advantage of the HMTS compared to the global survey method.

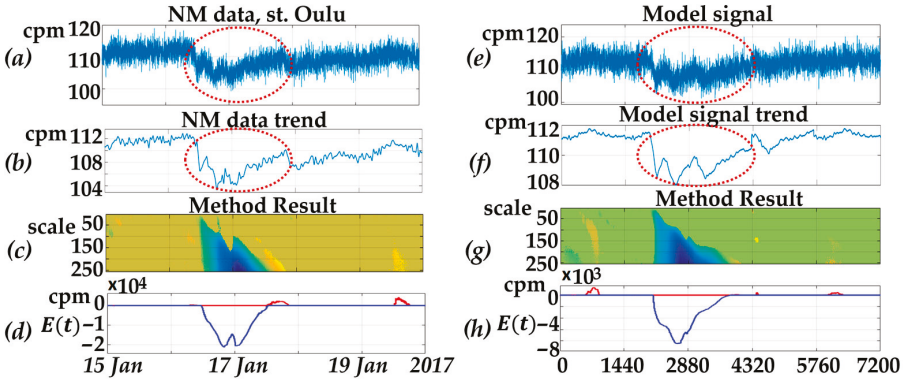


Figure 7. Processing results: (**a**) NM signal at Oulu station; (**b**) trend of NM signal; (**c**,**d**) results of the method with NM data; (**e**) the model signal; (**f**) trend of the model signal; (**g**,**h**) results of the method with the model data.

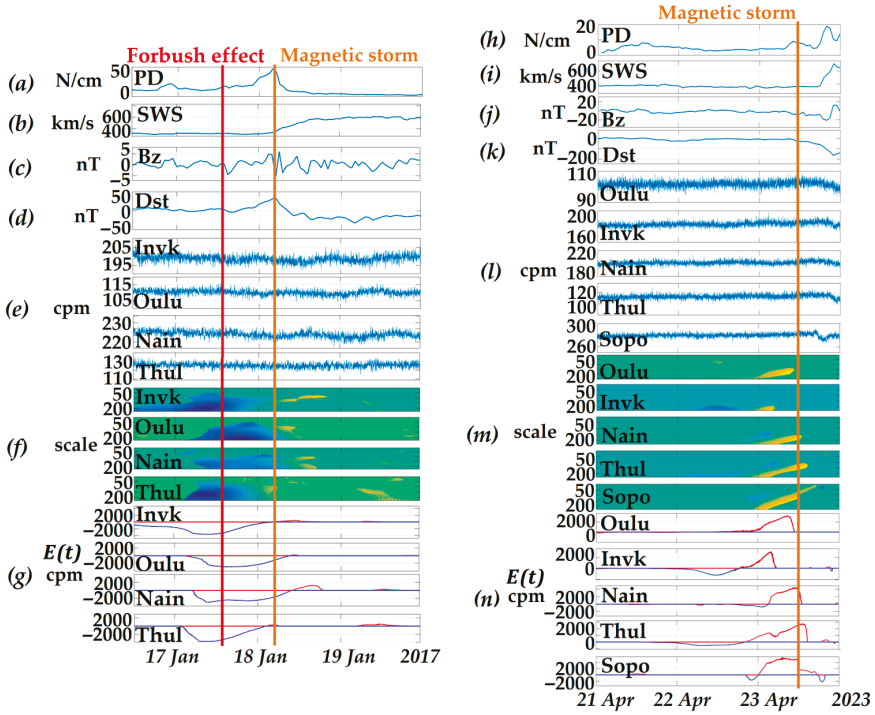


Figure 8. Processing results: (**a**,**h**) proton density; (**b**,**i**) solar wind speed; (**c**,**j**) Bz (GSM) component of the interplanetary magnetic field; (**d**,**k**) geomagnetic activity Dst-index; (**e**,**l**) NM data; (**f**,**g**,**m**,**n**) results of the method with NM data.

5. Conclusions

The paper suggested the hybrid model of a nonstationary time series (HMTS) including the neural network component and the nonlinear adaptive approximating scheme.

The HMTS parameters were estimated using the data of neutron monitor ground stations, recording cosmic ray intensities in the near-Earth space. To estimate the HMTS efficiency, model data were also used. Their structure corresponded to neutron monitor data. The estimates showed the high accuracy of the method in the problem of anomaly detection:

1. The HMTS regular component adequately describes neutron monitor time series during the periods of anomaly absence. The MSE model values are close to 0 and errors are the Gaussian noise.
2. During the anomalous periods, the neural network model errors increase that makes it possible to detect anomalies effectively.
3. The detection probability of the anomaly, lasting for 20 counts, is more than 80% for the signal–noise ratio of 1.5 (at the false alarm rate $\alpha = 0.05$). When the anomaly lasts for more than 60 counts, the detection probability is close to 90% for the signal–noise ratio equal to 1.3.

Comparison of the HMTS with continuous wavelet transform confirmed the HMTS efficiency in the problems of data analysis and anomaly detection. In the result of the correlated noise effect (including the solar-diurnal cycle of the neutron monitor data), it was impossible to detect an anomaly in the data based on the continuous wavelet transform. Application of the method allowed us to detect the anomaly against the background of correlated noise.

The processing results for the data from different neutron monitor stations showed the high sensitivity of the HMTS. Application of the HMTS made it possible to detect the anomalies based on the data from each separate station compared to the widely applied method of global survey. This shows the possibility to minimize the number of engaged stations when detecting anomalies in cosmic ray variations. The continuous operation of neutron monitor stations is not always provided, thus, this result is important.

The developed HMTS can be recommended for anomaly detection in neutron monitor data, and it is effective even for a small number of operating stations. Moreover, the HMTS can be recommended for the modeling and analysis of nonstationary time series including regular components of nonlinear structures and local features of different form and duration.

Author Contributions: Conceptualization, O.M.; methodology, O.M. and B.M.; software, B.M.; validation, B.M.; formal analysis, O.M. and B.M.; writing—review and editing, O.M. and B.M.; project administration, O.M. All authors have read and agreed to the published version of the manuscript.

Funding: The work was supported by IKIR FEB RAS State Task (subject registration No. 124012300245-2).

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors are grateful to the institutes that support the neutron monitor stations (http://www01.nmdb.eu/, http://spaceweather.izmiran.ru/) (accessed on 1 November 2023), the data of which were used in the work.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kuznetsov, V.D. Space weather and risks of space activity. *Space Tech. Technol.* **2014**, *3*, 3–13.
2. Schlickeiser, R. *Cosmic Ray Astrophysics*; Springer GmbH & Co., KG.: Berlin/Heidelberg, Germany, 2002; p. 519.
3. Poluianov, S.V.; Usoskin, I.G.; Mishev, A.L.; Shea, M.A.; Smart, D.F. GLE and Sub-GLE Redefinition in the Light of High-Altitude Polar Neutron Monitors. *Sol. Phys.* **2017**, *292*, 176. [CrossRef]
4. Belov, A.V.; Eroshenko, E.A.; Yanke, V.G.; Oleneva, V.A.; Abunina, M.A.; Abunin, A.A. Global survey method for the world network of neutron monitors. *Geomagn. Aeron.* **2018**, *58*, 356–372. [CrossRef]
5. Australian Space Weather. Available online: www.sws.bom.gov.au (accessed on 1 November 2023).
6. Mavromichalaki, H.; Paschalis, P.; Gerontidou, M.; Tezari, A.; Papailiou, M.C.; Lingri, D.; Livada, M.; Stassinakis, A.; Crosby, N.; Dierckxsens, M. The updated GLE alert system by ANEMOS. In Proceedings of the Hybrid Symposium on Cosmic Ray Studies with Neutron Detectors, Athens, Greece, 26–30 September 2022; pp. 97–103.
7. Veselovsky, I.S.; Yakovchuk, O.S. On the forecast of solar proton events according to the data of ground-based neutron monitors. Astronomical Herald. *Sol. Syst. Explor.* **2011**, *45*, 365–375.

8. Ramos, A.A.; Cheung, M.C.M.; Chifu, I.; Gafeira, R. Machine learning in solar physics. *Living Rev. Sol. Phys.* **2023**, *20*, 4. [CrossRef]
9. Wang, S.; Wang, H.; Perdikaris, P. On the eigenvector bias of Fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. *Comput. Methods Appl. Mech. Eng.* **2021**, *384*, 113938. [CrossRef]
10. Shaik, N.B.; Pedapati, S.R.; Othman, A.R.; Bingi, K.; Dzubir, F.A.A. An intelligent model to predict the life condition of crude oil pipelines using artificial neural networks. *Neural Comput. Appl.* **2021**, *33*, 14771–14792. [CrossRef]
11. Shaik, N.B.; Pedapati, S.R.; Taqvi, S.A.A.; Othman, A.R.; Dzubir, F.A.A. A Feed-Forward Back Propagation Neural Network Approach to Predict the Life Condition of Crude Oil Pipeline. *Processes* **2020**, *8*, 661. [CrossRef]
12. Shaik, N.B.; Jongkittinarukorn, K.; Benjapolakul, W.; Bingi, K. A novel neural network-based framework to estimate oil and gas pipelines life with missing input parameters. *Sci. Rep.* **2024**, *14*, 4511. [CrossRef]
13. Chen, X.; Li, X.; Yu, S.; Lei, Y.; Li, N.; Yang, B. Dynamic Vision Enabled Contactless Cross-Domain Machine Fault Diagnosis with Neuromorphic Computing. *IEEE/CAA J. Autom. Sin.* **2024**, *11*, 788–790. [CrossRef]
14. Koundal, P. Graph neural networks and application for cosmic-ray analysis. In Proceedings of the 5th International Workshop on Deep Learning in Computational Physics, Moscow, Russia, 28–29 June 2021.
15. Kalashev, O.; Pshirkov, M.; Zotov, M. Identifying nearby sources of ultra-high-energy cosmic rays with deep learning. *J. Cosmol. Astropart. Phys.* **2020**, *2020*, 005. [CrossRef]
16. Bojang, P.O.; Yang, T.-C.; Pham, Q.B.; Yu, P.-S. Linking Singular Spectrum Analysis and Machine Learning for Monthly Rainfall Forecasting. *Appl. Sci.* **2020**, *10*, 3224. [CrossRef]
17. Lee, T. EMD and LSTM Hybrid Deep Learning Model for Predicting Sunspot Number Time Series with a Cyclic Pattern. *Sol. Phys.* **2020**, *295*, 82. [CrossRef]
18. Gocheva-Ilieva, S.; Ivanov, A.; Kulina, H.; Stoimenova-Minova, M. Multi-Step Ahead Ex-Ante Forecasting of Air Pollutants Using Machine Learning. *Mathematics* **2023**, *11*, 1566. [CrossRef]
19. Boussaada, Z.; Curea, O.; Remaci, A.; Camblong, H.; Mrabet Bellaaj, N. A Nonlinear Autoregressive Exogenous (NARX) Neural Network Model for the Prediction of the Daily Direct Solar Radiation. *Energies* **2018**, *11*, 620. [CrossRef]
20. Tang, R.; Zeng, F.; Chen, Z.; Wang, J.-S.; Huang, C.-M.; Wu, Z. The Comparison of Predicting Storm-Time Ionospheric TEC by Three Methods: ARIMA, LSTM, and Seq2Seq. *Atmosphere* **2020**, *11*, 316. [CrossRef]
21. Zhao, X.; Lu, X.; Quan, W.; Li, X.; Zhao, H.; Lin, G. An Effective Ionospheric TEC Predicting Approach Using EEMD-PE-Kmeans and Self-Attention LSTM. *Neural Process Lett.* **2023**, *55*, 9225–9245. [CrossRef]
22. Mandrikova, O.; Fetisova, N.; Polozov, Y. Hybrid Model for Time Series of Complex Structure with ARIMA Components. *Mathematics* **2021**, *9*, 1122. [CrossRef]
23. Yang, B.; Lei, Y.; Li, X.; Li, N. Targeted transfer learning through distribution barycenter medium for intelligent fault diagnosis of machines with data decentralization. *Expert Syst. Appl.* **2024**, *244*, 122997. [CrossRef]
24. Mandrikova, O.; Mandrikova, B. Hybrid Method for Detecting Anomalies in Cosmic ray Variations Using Neural Networks Autoencoder. *Symmetry* **2022**, *14*, 744. [CrossRef]
25. Mandrikova, O.; Mandrikova, B.; Rodomanskay, A. Method of Constructing a Nonlinear Approximating Scheme of a Complex Signal: Application Pattern Recognition. *Mathematics* **2021**, *9*, 737. [CrossRef]
26. Mandrikova, O.; Mandrikova, B.; Esikov, O. Detection of Anomalies in Natural Complicated Data Structures Based on a Hybrid Approach. *Mathematics* **2023**, *11*, 2464. [CrossRef]
27. Mallat, S. *A Wavelet Tour of Signal Processing*; Academic Press: London, UK, 1999; p. 620.
28. Pattanayak, S. *Pro Deep Learning with TensorFlow: A Mathematical Approach to Advanced Artificial Intelligence in Python*; Apress: Bangalore, India, 2023; p. 398.
29. Witte, R.S.; Witte, J.S. *Statistics*, 11th ed.; Wiley: New York, NY, USA, 2017; p. 496.
30. Donoho, D.L.; Johnstone, I.M. Ideal spatial adaptation via wavelet shrinkage. *Biometrika* **1994**, *81*, 425–455. [CrossRef]
31. Mandrikova, O.V. Intelligent methods for natural data analysis: Application to space weather. *Comput. Optics.* **2024**, *48*, 139–148. [CrossRef]
32. Levin, B.R. *Theoretical Foundations of Statistical Radio Engineering*; Radio and Communications: Moscow, Russia, 1989.
33. Real Time Data Base for the Measurements of High-Resolution Neutron Monitor. Available online: www.nmdb.eu (accessed on 1 November 2023).
34. IZMIRAN Space Weather Forecast Center. Catalog of Forbush Effects and Interplanetary Disturbances. Available online: http://spaceweather.izmiran.ru/rus/fds2019.html (accessed on 11 November 2023).
35. Institute of Applied Geophysics. Available online: http://ipg.geospace.ru/ (accessed on 11 October 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Predicting Compressive Strength of High-Performance Concrete Using Hybridization of Nature-Inspired Metaheuristic and Gradient Boosting Machine

Nhat-Duc Hoang [1,2], Van-Duc Tran [2,3] and Xuan-Linh Tran [1,2,*]

[1] Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam; hoangnhatduc@duytan.edu.vn
[2] Faculty of Civil Engineering, Duy Tan University, Da Nang 550000, Vietnam; tranvanduc1@dtu.edu.vn
[3] International School, Duy Tan University, Da Nang 550000, Vietnam
* Correspondence: tranxuanlinh@duytan.edu.vn; Tel.: +84-0236-3827111

Abstract: This study proposes a novel integration of the Extreme Gradient Boosting Machine (XGBoost) and Differential Flower Pollination (DFP) for constructing an intelligent method to predict the compressive strength (CS) of high-performance concrete (HPC) mixes. The former is employed to generalize a mapping function between the mechanical property of concrete and its influencing factors. DFP, as a metaheuristic algorithm, is employed to optimize the learning phase of XGBoost and reach a fine balance between the two goals of model building: reducing the prediction error and maximizing the generalization capability. To construct the proposed method, a historical dataset consisting of 400 samples was collected from previous studies. The model's performance is reliably assessed via multiple experiments and Wilcoxon signed-rank tests. The hybrid DFP-XGBoost is able to achieve good predictive outcomes with a root mean square error of 5.27, a mean absolute percentage error of 6.74%, and a coefficient of determination of 0.94. Additionally, quantile regression based on XGBoost is performed to construct interval predictions of the CS of HPC. Notably, an asymmetric error loss is used to diminish overestimations committed by the model. It was found that this loss function successfully reduced the percentage of overestimated CS values from 47.1% to 27.5%. Hence, DFP-XGBoost can be a promising approach for accurately and reliably estimating the CS of untested HPC mixes.

Keywords: high-performance concrete; compressive strength; gradient boosting machine; differential flower pollination; overestimation reduction

MSC: 68Txx; 68Vxx

1. Introduction

Concrete has a vital role in construction engineering because of its distinct combination of compressive strength, durability, moldability, as well as affordability [1]. According to statistical data from previous studies [1,2], the yearly consumption of this construction material may reach as high as 35 billion tons. Among various types of concrete, high-performance concrete (HPC) has been increasingly employed and has become a prominent material, applicable to a wide range of construction projects (e.g., skyscrapers, bridge structures, parking areas, and heavy-duty pavement). As summarized in [3], this advanced material offers various advantages over normal concrete, including high strength, good abrasion resistance, high elastic modulus, low permeation, and high resistance to chemical attack.

HPC is generally different from ordinary concrete because it often employs additional cementitious materials, such as silica fume, ground granulated blast furnace slag (GGBFS), and fly ash [4]. Typically, the content of cementitious material in HPC is high and the

water-to-cement (w/c) ratio is low. The total content of the cementitious materials in a mix was recommended to be as high as 400 to 550 kg/m^3 [5]. The w/c ratio often ranges between 0.25 and 0.40 [6] and can be as low as 0.22 [5]. In addition, the use of silica fume and the w/c ratio often necessitate the inclusion of superplasticizers or water-reducing agents [7].

In recent years, there has been a growing demand for HPC that meets high standards in the construction industry [8]. This demand puts more pressure on the task of mix design and requires a deeper understanding of the relationship between the concrete's mechanical properties and its constituents. Accurate prediction of the mechanical properties of concrete has increasingly become a major concern because it is required by design codes. In addition, the demands of new concrete mixtures have motivated the establishment of reliable models for estimating their mechanical strengths [9]. In this regard, the development of innovative solutions to modeling the mechanical properties of HPC is a pressing need for both researchers and practitioners. Not only can these modeling approaches be useful for the investigation of influential factors in HPC's properties, but they also play a vital role in designing HPC mixtures.

The compressive strength (CS) is considered a pivotal indication of HPC quality [3,10]. It serves as input information for the mix design and structural analysis of concrete elements. Moreover, other mechanical properties of concrete, such as tensile strength and elastic modulus, can be inferred from the CS via correlating equations. In practice, laboratory tests are required to obtain the CS of HPC mixes. These tests are costly and time-consuming. Moreover, the laboratory test results regarding HPC's mechanical behaviors have been continuously reported in academic papers and technical reports. Therefore, making use of the available test outcomes and constructing data-driven models for estimating the CS of HPC is a research direction worth pursuing. Accurate prediction of the CS values not only enhances the efficiency of the mix design process but also greatly supports the scheduling of construction activities involving formwork removal and concrete slab re-shoring [11].

Nevertheless, the task of estimating HPC is still considered challenging. The reason is that HPC materials are sophisticated and nonhomogeneous mixtures that comprise different components [12]. Distinct from normal concrete, HPC often includes mineral and chemical admixtures, which help enhance its mechanical properties. However, the inclusion of these materials affects the hydration reaction between the cementitious materials and water [13]. Accordingly, the mechanical behavior of HPC is much more complicated than that of an ordinary one. For instance, Mazloom et al. [14] experimentally found that the inclusion of silica fume deteriorated the short-term CS of the concrete mixes. Additionally, as stated in [3], the CS of HPC is dependent on multiple factors, including the mix proportions, features of materials, and curing age. Lee et al. [12] pointed out the nonlinear relationship between the CS of HPC and its influencing factors; the authors also demonstrated the need for advanced regression modeling tools to deal with the complex problem of interest.

In previous studies [15,16], an artificial neural network (ANN) was applied to construct predictive models that were superior to conventional regression models. Particularly, the quantities of cement, ground granulated blast furnace slag, water, plasticizer, aggregates, and the age of samples were employed in [16] as predictor variables. This study pointed out the effectiveness of the Levenberg–Marquardt algorithm that helped train the ANN model. However, the authors did not consider the crucial effect of silica fume on the response.

Chou and Pham [3] investigated the capability of ensemble learning approaches and concluded that these types of machine learning methods were particularly suited for the problem at hand. However, one limitation of the paper is that it only makes use of the default settings of the models' parameters. More sophisticated fine-tuning of the models' configurations may result in even better predictive performance. Decision tree ensembles were also studied in [17]; the authors concluded that ensemble models were superior to single decision trees. Similar to the work of [3], Erdal [17] also recommended the use of tree ensembles (e.g., random subspace, bagging, and gradient boosting) for modeling

the mechanical behavior of HPC. Nevertheless, state-of-the-art decision tree ensembles, such as Extreme Gradient Boosting Machines, should be used for enhancing the prediction performance [18,19].

Chithra et al. [20] relied on an ANN to conduct a comparative study on the CS of HPC containing nanosilica and copper slag; the ANN was shown to significantly outperform multiple regression analysis. Yu et al. [21] employed Support Vector Regression (SVR) for predicting the mechanical property of interest; the independent variables were the contents of water, cement, slag, fly ash, superplasticizer, aggregates, and curing age. The authors demonstrated the superiority of SVR over the M5 model tree and adaptive neural fuzzy inference system. Kaloop et al. [22] benchmarked the performance of a gradient tree boosting machine against Gaussian process regression and multivariate adaptive regression splines; the gradient boosting approach achieved the most desired outcome. Nguyen et al. [23] highlighted the advantages of the Extreme Gradient Boosting Machine (XGBoost) over other machine learning approaches in the task of CS estimation. Nevertheless, this study relied on a random search methodology for the selection of hyper-parameters; in addition, the input variables of the model did not include silica fume, which is a crucial component of HPC.

Gradient boosting machines were used in [10] for predicting the CS of concrete mixes containing silica fume, fly ash, and polypropylene fiber. This study demonstrated the good performance of the mentioned machine learning models. However, the previous work did not take into account the effect of GGBFS on the CS of concrete mixes. Lee et al. [12] put forward super-learner models, which combined individual machine learning models for strength estimation. The authors relied on four different datasets with dissimilar sets of predictor variables. These datasets vary in size, and silica fume was not included in three out of four datasets. In addition, information regarding the size of specimens was reported; this fact might cause difficulties in strength conversion among the samples. In addition, the use of metaheuristic optimizers in hyper-parameter settings may further enhance the models' performance and help automate the models' construction phases [24].

As can be seen from the literature, ANN-based models have been extensively studied in CS estimation, and their positive outcomes have been widely reported. Recent review papers [9,25] have confirmed that the ANN is the most commonly applied method for the task of interest. Al Yamani et al. [26] compared the performance of different models and pointed out the superiority of the ANN over random forest and decision tree regressors. In addition, SVR is also a capable function approximator that is suitable for small- and medium-scale datasets [9,27]. Nevertheless, to further enhance the accuracy and reliability of the CS estimation, the capabilities of other state-of-the-art machine learning models should be explored and reported to the research community.

As can be seen from the literature review, although previous works have shown the capability of XGBoost in estimating the CS of HPC [12,28,29], the integration of XGBoost and metaheuristic algorithms for automating the model construction phase has rarely been investigated. The existing models have mainly focused on improving the accuracy of the prediction models. The number of papers dedicated to improving the reliability of the estimated property of HPC mixes is still limited. The current work attempts to fill these gaps in the literature by employing advanced metaheuristic algorithms and data-driven analyses. In detail, the loss function of XGBoost is revised so that the model is able to reduce the proportion of overestimated results as well as derive the prediction intervals.

Based on the review work [27] and recent papers [23,24,30,31], this study relies on the advanced approach of XGBoost to construct a data-driven model for estimating the CS of HPC. Considering the growing tendency to use metaheuristic algorithms for automating the model's construction process [24,32–35], an integration of XGBoost [36] and Differential Flower Pollination (DFP) [37] is put forward. This combined framework is then verified by a historical dataset consisting of 400 data samples. The predictive accuracy of the proposed model is compared with that of benchmark machine learning approaches to confirm its advantages.

2. Research Significance

The current paper aims to propose and verify a data-driven approach for estimating the CS of HPC. The newly proposed method is a hybridization of the XGBoost regressor and the DFP metaheuristic. The main contribution of this paper is multifaceted and can be summarized as follows:

(i) An integrated model, named DFP-XGBoost, is proposed to deal with the nonlinear and multivariate functional mapping between the CS of HPC mixes and their influencing factors. Notably, DFP, as a metaheuristic optimizer, is employed to assist the learning phase of the regressor. An integration of these two computational intelligence approaches for the task of interest has not yet been investigated.

(ii) An innovative objective function is designed for DFP to express the two goals of model building: minimizing the prediction error and maximizing the generalization property. With the help of the DFP metaheuristic, it is able to automate the model's construction phase. Therefore, the newly developed DFP-XGBoost can be easily applied and updated by practitioners without in-depth domain knowledge in machine learning and metaheuristic.

(iii) A dataset, including 400 samples, is collected from laboratory testing results. This dataset is used to train and verify the proposed hybrid method. Besides the curing age, the contents of cement, fly ash, silica fume, ground granulated blast furnace slag, fine aggregate, coarse aggregate, water, and a water-reducing agent are employed as predictor variables. Different from previous studies [16,23], the gathered dataset has taken into account the effect of silica fume, which is a crucial ingredient in HPC [5,13,14].

(iv) In addition to prediction accuracy, reliability is also a major concern in CS estimation [38,39]. Restriction of overestimation is an important task in model building. However, improving the reliability of the estimation has rarely been addressed in the case of HPC. The current work is an attempt to fill this gap in the literature. An asymmetric loss function is utilized in the training phase of the XGBoost regressor to express the bias against overestimations. By doing so, the number of overestimated CS values can be reduced, and the model's reliability is enhanced accordingly. Additionally, XGBoost-based quantile regression is used to derive the prediction interval of the CS values.

(v) To assess the contribution of predictor variables to the model's output, this study resorts to the Fourier Amplitude Sensitivity Test. This is a variance-based global sensitivity analysis approach for computing the effects of the input factors on the CS of HPC. This test helps identify the constituents that considerably affect the response.

(vi) A program based on DFP-XGBoost with a graphical user interface has been developed in Python to ease the implementation of the model. Using this program, the user can obtain a prediction of the CS of untested HPC mixes.

3. Research Method

3.1. Extreme Gradient Boosting Machine Regressor

The Extreme Gradient Boosting Machine (XGBoost), put forward in [36], is widely considered as a powerful ensemble learning approach for nonlinear and multivariate function approximations [23,24]. This regressor gradually improves a single weak regression tree by combining it with other individual models to establish a collectively robust model. Iteratively, the prediction error of the previous weak learners is fitted by the next one. Herein, XGBoost implements a process of adding weak models to its ensemble in which the learning phase is formulated as a gradient descent algorithm over an objective function. The overall structure of an XGBoost model is demonstrated in Figure 1.

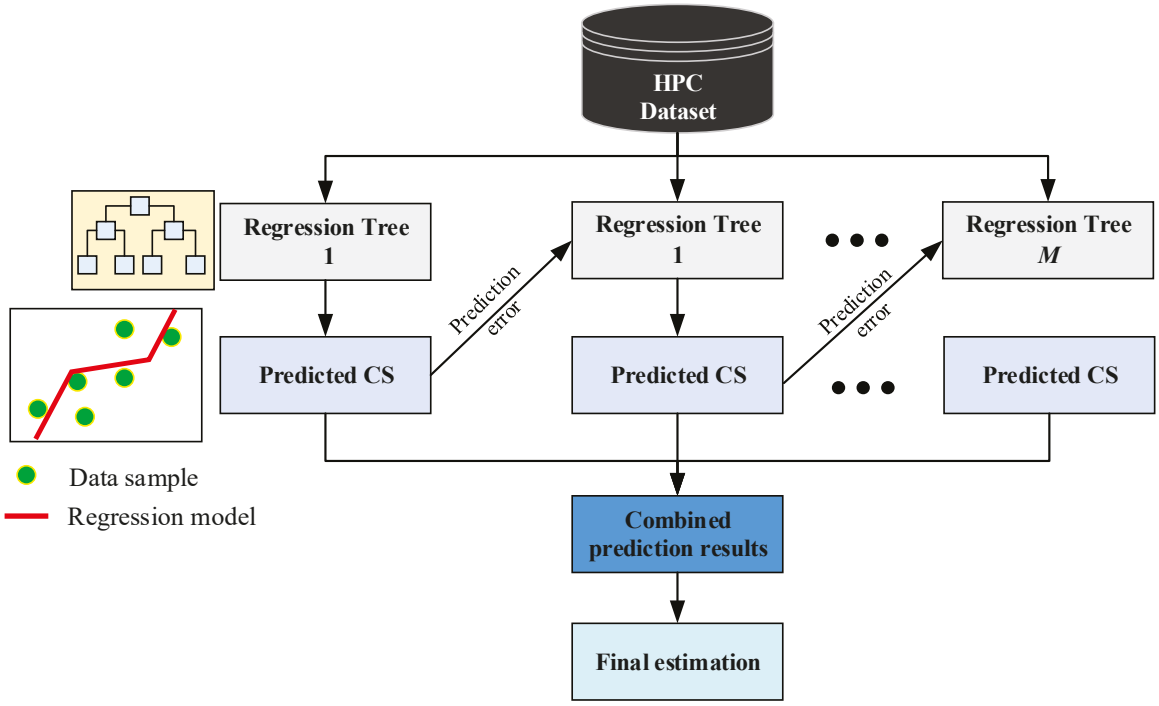


Figure 1. Structure of an XGBoost regressor used for CS estimation.

The objective function of XGBoost can be expressed as follows [18,36]:

$$f_{obj} = \sum_i L(y_i, F(x_i)) + \sum_k \Omega(f_k) \tag{1}$$

where $\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_j^T w_j^2$ is the regularization term; T denotes the leaf number in a regression tree f; w_j is the score of a leaf j; γ represents a threshold of the score function used for splitting regression trees. λ denotes a regularization coefficient.

To compute f_{obj}, it is required to specify a loss function. For regression analysis, the commonly used loss function is the squared error loss (SEL) [40], which is expressed as follows:

$$L(t, y) = (t - y)^2 \tag{2}$$

where t and y denote the actual and forecasted values of the response, which is the CS of HPC mixes.

To find a solution for the aforementioned optimization problem, a Taylor's second-order approximation is employed. In addition, the first- and second-order gradients of the loss function $L(t, y)$ must be computed. Accordingly, the optimal score of each leaf in a decision tree can be expressed as follows [18]:

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \tag{3}$$

where g_i and h_i represents the first- and second-order gradients of $L(t, y)$, respectively, and I_j denotes the data instances that arrive at a leaf.

After being trained, an XGBoost model used for CS estimation is given by:

$$F_{XGBoost} = \sum_{k=1}^{M} f_k(x) \tag{4}$$

where M denotes the number of individual models in the ensemble; x is the feature vector of the model, which describes the characteristics of a HPC mixes; $F_{XGBoost}$ is the CS value of the mix estimated by XGBoost.

Notably, XGBoost offers various advantages over other ensemble approaches in model building. As pointed out in [41], this boosting machine employs a smart tree-splitting mechanism and an efficient randomization method for enhancing the diversity of the collective model. In addition, regularization terms can be incorporated into the objective function of XGBoost to make this model more resilient to overfitting. This boosting machine also implements a level-wise strategy to assess the quality of possible splits across the training set. Therefore, XGBoost models are able to minimize both variance and bias effectively. Last but not least, the training process of XGBoost can be executed parallelly on graphics processing units. This feature makes XGBoost scalable and allows it to cope well with large-scale datasets.

3.2. Differential Flower Pollination

Differential Flower Pollination (DPF), put forward in [37], is a nature-inspired metaheuristic algorithm. DFP is designed to combine the searching operations of Differential Evolution (DE) [42] and the Flower Pollination Algorithm (FPA) [43]. This metaheuristic relies on DE for exploitation of the search space and the FPA for enhancing the exploration capability. In the current work, DFP is used for automating the model selection phase of the XGBoost regressor. This task is formulated as a global optimization problem in which the hyper-parameters of XGBoost are modeled as decision variables. An objective function expressing the model's predictive capability is used to guide the population of the metaheuristic approach to the optimal solution. DFP inherits effective searching operators from DE, including mutation and cross-over. Moreover, it employs a Levy flight-based global search as described in the FPA [43]. DFP is particularly suitable for optimizing the performance of machine learning models because it is a gradient-free method and possesses good search capability [44]. For more details regarding the implementation of DFP, readers are guided to the previous works of [37,44].

3.3. The Collected Dataset

Since the CS estimation is modeled as a supervised learning task, the preparation of data samples containing actual laboratory test results is crucial. These data samples provide the features and expected values of the response. Herein, the features consist of the mixes' constituents and testing age. The response is the testing outcome of the mixes' CS. To train the XGBoost regressor, this study has conducted a comprehensive literature survey and gathered data samples from the current literature. The independent variables in the study are selected via the findings of previous works [8,13,14,45] as well the availability of data. Those variables are used as inputs for the machine learning model to estimate the CS of HPC measured in MPa. Due to the availability and compatibility of data, other influencing factors, such as curing conditions and the size distribution of aggregates, have not been considered in the current work.

Moreover, the selection of the data samples is based on the recommendations of the w/c ratio as well as the total content of the cementitious materials in [5,6]. To ensure the quality of the laboratory testing results, the data samples are collected from reliable academic articles. After the literature review process, a historical dataset, consisting of 400 data points, is established. The sources of the data samples are reported in Table 1. It is noted that the specimen sizes in different data sources are inconsistent. Therefore, this study relies on correlation factors [38] to standardize the CS values. The correlation factors are provided in Table 2. The statistical descriptions of the variables in the dataset are summarized in Table 3.

Table 1. Data sources.

Data Source	Number of Laboratory Tests	Proportion (%)	Type of Specimens	Reference
1	100	25.00	150 × 150 × 150 mm cube	[16]
2	4	1.00	100 × 200 mm cylinder	[46]
3	16	4.00	100 × 100 × 100 mm cube	[47]
4	52	13.00	100 × 100 × 100 mm cube	[10]
5	28	7.00	100 × 100 × 100 mm cube	[14]
6	200	50.00	150 × 300 mm cylinder	[6]

Table 2. Correlation factors used for standardizing the compressive strength values.

Specimen	Cube	Cube	Cylinder	Cylinder
Dimension (mm)	150 × 150 × 150	100 × 100 × 100	100 × 200	150 × 300
Correlation factor	1.119	1.000	1.020	1.063

Table 3. Statistical description of the variables.

Variables	Unit	Notation	Min	Average	Std	Skewness	Max
Cement content	kg/m^3	X_1	80.00	378.16	103.28	−1.38	527.20
Fly ash content	kg/m^3	X_2	0.00	6.42	22.56	3.51	123.50
Silica fume content	kg/m^3	X_3	0.00	19.93	21.63	0.71	75.00
GGBFS content	kg/m^3	X_4	0.00	96.40	84.87	1.19	360.00
Fine-aggregate content	kg/m^3	X_5	488.00	649.20	79.25	0.07	804.96
Coarse-aggregate content	kg/m^3	X_6	915.20	1129.35	48.23	−1.33	1203.00
Water content	kg/m^3	X_7	120.00	161.13	14.45	−1.52	180.00
WRA content	kg/m^3	X_8	1.30	9.42	4.18	−0.07	18.00
Age	day	X_9	1.00	48.68	91.37	2.93	400.00
Compressive strength	MPa	Y	22.54	69.66	22.13	−0.06	120.86

Note: Std denotes the standard deviation.

In the collected dataset, the contents of cement, fly ash, silica fume, GGBFS, fine aggregate, coarse aggregate, water, water-reducing agent (WRA), and curing age are employed as influencing variables. The use of Supplementary Cementing Materials, including silica fume, fly ash, and GGBFS, is crucial for HPC mixes to modify their mechanical performance. Silica fume is a highly pozzolanic material that is capable of improving the strength and durability of concrete. Fly ash and GGBFS are by-products of industrial processes that are also widely used to enhance the mechanical properties and sustainability of concrete [48–51]. In addition, since HPC generally requires low w/c ratios, a WRA (e.g., high-range water reducers and superplasticizers) is commonly employed to reduce the water content and ensure the workability of concrete mixes [5,6].

3.4. Benchmark Methods

3.4.1. Artificial Neural Network

The artificial neural network (ANN) is a machine learning approach inspired by actual biological brains in the natural world [52]. ANNs have been widely used for CS estimation, and positive results obtained from this method were reported in various studies [9,27]. A typical ANN model consists of an input, a hidden layer, and an output layer. The first layer receives input signals that contain the characteristics of a concrete mixture. These input signals are then transmitted to the neurons in an ANN's hidden layer, where various data transformation and feature engineering processes are performed. To deal with nonlinear functional mappings, the ANN relies on activation functions (e.g., the sigmoid function) to process the input data of a neuron. The output layer yields the estimated CS values based on the signals arriving at the input layer. To construct an ANN model for predicting

the CS of HPC, its structure, specified by the synaptic weights, must be trained. Based on previous studies [16,53], the current paper resorts to the Levenberg–Marquardt algorithm for constructing the ANN model.

3.4.2. Support Vector Regression

Support vector regression (SVR) is also a popular machine learning approach used for CS estimation [21,27]. This method constructs a mapping function by identifying a set of crucial data samples within the training set. These crucial data samples help locate the inference model and are called support vectors. Notably, only a small fraction of the training samples are selected to construct an SVR model. Therefore, SVR can achieve both the objectives of low prediction error and good sparseness. The reason is that a sparse model is less affected by noise. In addition, SVR employs the epsilon-insensitive loss function during its training process. Using this loss function, prediction errors that are less than a certain limit (ε_{SVR}) are not penalized by the model. The training process of an SVR model is formulated as a quadratic programming problem that can be solved by a nonlinear programming solver. To deal with nonlinear mapping, SVR relies on kernel functions to perform data transformations. The radial basis function is often employed for regression analysis [54].

4. Result and Discussion

4.1. Experimental setting

As stated in the previous section, the dataset, consisting of 400 samples and nine independent variables, is employed to train and validate the machine learning methods. This study randomly divides the collected data samples into two sets: the training set (90%) and the testing set (10%). The former is used to construct the proposed approach, named DFP-XGBoost. The latter, which includes 40 samples, is reserved to inspect the model's generalization property. Since the model optimization of DFP-XGBoost requires the fine-tuning of various hyper-parameters, this study allocates 90% of the samples in the training set to provide sufficient data for the model construction phase.

Due to the effect of randomness in data sampling, one experiment with the model's training and testing may not be sufficient for evaluating its forecasting capability. Hence, this study has performed the experiment with DFP-XGBoost 20 times. In each experiment, 10% of the collected samples are randomly drawn to serve as untested concrete mixes. The rest of the dataset is used for model construction. Since the variables in the dataset have different ranges, the variables with large magnitudes may dominate the others. To avoid this situation, a data normalization technique should be employed to pre-process the dataset. This study relies on the Z-score normalization method for standardizing the ranges of variables. The Z-score normalization equation is given by:

$$X_Z = \frac{X_O - \mu_X}{\sigma_X} \tag{5}$$

where X_Z and X_O denote the normalized and original variables, respectively. It is noted that the variables include all the input features as well as the CS of HPC. μ_X and σ_X are the mean and standard deviation (std) of the original variables, respectively.

Moreover, to assess the performance of the prediction models, the root mean square error (RMSE), mean absolute percentage error (MAPE), and coefficient of determination (R^2) are used in this section. RMSE is essentially the quadratic mean of the discrepancy between the actual compressive strength values (measured from destructive testing) and the estimated ones (yielded by the machine learning models). RMSE is non-negative; the closer it is to 0, the better the prediction accuracy. MAPE computes the average magnitude of errors estimated by the machine learning model. This index varies from 0 to 100%. As pointed out by [24], MAPE less than 10% indicates excellent performance; values in the range of [10%, 20%] show good performance. Therefore, it is expected that a decent prediction model for the CS estimation of HPC mixes should have MAPE <20%. In addition,

R^2 exhibits the proportion of the variation in the CS that can be explained by the machine learning approach; it ranges from 0 to 1. The larger the value of R2, the more predictive the model is.

These indices were selected in this study because they are commonly used for performance measurement and comparison in previous works related to the compressive strength estimation of concrete mixes [24,26,30]. The metrics of RMSE, MAPE, and R^2 are given by:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - t_i)^2} \tag{6}$$

$$MAPE = \frac{100}{N} \times \sum_{i=1}^{N}\frac{|y_i - t_i|}{y_i} \tag{7}$$

$$R^2 = 1 - \frac{\sum\limits_{i=1}^{N}(t_i - y_i)^2}{\sum\limits_{i=1}^{N}(t_i - \bar{t})^2} \tag{8}$$

where t_i and y_i are the actual and forecasted CS of an HPC mix, respectively. N denotes the number of data samples.

It is noted that the proposed method is a combination of DFP and XGBoost. The DFP metaheuristic is employed to optimize the performance of the XGBoost regressor by identifying the most suitable set of its hyper-parameters. The predictive capability of XGBoost strongly depends on the number of individual trees (M) in the ensemble, the learning rate, the maximum tree depth (MTD), the L_2-regularization parameter (λ), and the L_1-regularization parameter (α). The number of individual trees and MTD significantly control the predictive power of XGBoost. The larger the values of M and MTD are, the more degrees of freedom the model has.

Sophisticated models are capable of explaining the complex mapping relationship between the predictor variables and the response. However, an excessively sophisticated model tends to suffer from overfitting. An overfitted model may achieve a desired goodness of fit in the training phase, but it may perform unsatisfactorily on unseen data samples. To avoid overfitting, the regularization parameters (λ and α) should be used. The larger those regularization parameters are, the more restricted the complexity of the model is. This study relies on DFP to automatically search for the most appropriate values of the five hyper-parameters of XGBoost.

To implement DFP, an objective function that expresses the model's quality must be defined. Accordingly, the following objective function is used during the optimization process of DFP:

$$f = \sum_{k=1}^{K}\left(\frac{RMSE_{Tr} + RMSE_{Te}}{K}\right) + \chi \times (M + MTD) + \rho \times \left(\frac{1}{1+\lambda} + \frac{1}{1+\alpha}\right) \tag{9}$$

where $\sum\limits_{k=1}^{K}\left(\frac{RMSE_{Tr}+RMSE_{Te}}{K}\right)$ denotes the model performance obtained from a cross-validation process with K = 5; $RMSE_{Tr}$ and $RMSE_{Te}$ are the prediction errors in the training and testing phases, respectively; χ and λ are weighting coefficients.

The objective function described in Equation (9) guides DFP to search for a model's configuration that features a fine balance between low prediction error and a high generalization property. Since these two goals are often conflicting in nature, the use of DFP is particularly helpful for the model-building process. The weighting coefficients in Equation (9) can be identified experimentally. Via several trial-and-error runs, the suitable values of χ and λ were found to be 0.01 and 0.001, respectively. It is noted that the lower boundaries of M, the learning rate, MTD, λ, and α are 1, 0.01, 1, 0, and 0, respectively. Their upper

boundaries are 100, 10, 10, 100, and 100. The DFP metaheuristic is coded in Python by the author. Meanwhile, the XGBoost regressor is built using the toolbox provided in [55].

To demonstrate the advantages of DFP-XGBoost, an ANN and SVR, two commonly employed models for CS estimation, are used. The ANN model is trained by the Levenberg–Marquardt algorithm and built in MATLAB's neural network toolbox [56]. The SVR model is constructed with the built-in functions provided in the scikit-learn machine learning library [57]. The hyper-parameters of the ANN and SVR are set via five-fold cross-validation processes [58]. In detail, the number of neurons in the hidden layer of the ANN model and its learning rate are selected to be 18 and 0.001, respectively. This model was trained for 300 epochs. In addition, the hyper-parameters of SVR, including the penalty coefficient (C_{SVR}), the kernel function parameter (K_{SVR}), and the radius of the epsilon-insensitive tube (ε_{SVR}), are set to be 400, 0.1, and 0.05, respectively. It is noted that experiments with the machine learning models in this paper were conducted on a Dell G15 5511 (Core i7-11800H and 16 GB RAM).

4.2. Prediction Results and Performance Comparison

As stated earlier, the hybrid model employs the DFP metaheuristic to fine-tune its hyper-parameters (i.e., the number of individual trees, learning rate, MTD, and regularization coefficients). DFP was implemented with 20 members in its population and executed for 100 generations. In the first generation, the population of DFP is generated randomly within its boundaries. This algorithm uses mutation, crossover, and Lévy flight-based global search operations to exploit and explore the search space. The evolutionary process of DFP used for optimizing the performance of the XGBoost regressor is demonstrated in Figure 2. The optimized model's hyper-parameters are found as follows: M = 12, learning rate = 1.42, MTD = 4, λ = 98.62, and α = 0.013.

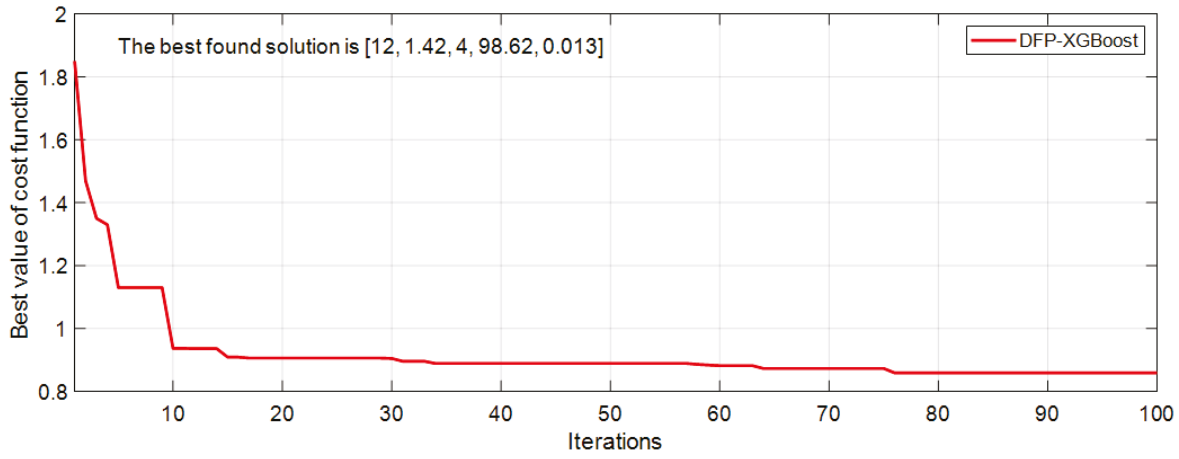


Figure 2. Optimization progress.

Table 4 summarizes the prediction outcomes of DFP-XGBoost and other benchmark models in terms of all performance measurement metrics. It is observable that the proposed method has achieved the best results with an RMSE of 5.270, an MAPE of 6.740%, and an R^2 of 0.942. The ANN is the second-best method, with an RMSE of 8.823, an MAPE of 11.836%, and an R^2 of 0.820. The result of SVR (RMSE = 10.461, MAPE = 15.212%, and an R^2 of 0.763) is inferior to that of DFP-XGBoost and the ANN. To quantify the correlation between the actual and estimated results, the R^2 index can be used. This index is also an indicator of the proportion of the variation in the CS of HPC that can be estimated from DFP-XGBoost and its influencing variables. With an R^2 of 0.94, the prediction results are satisfactory because the hybrid model is capable of explaining 94% of the variation in the output variable. Moreover, the ANN and SVR are only capable of explaining 82% and 76% of the variation in the modeled variable. Figure 3 graphically shows the improvements in

results obtained by DFP-XGBoost. The newly developed method helps decrease roughly 40% and 43% of the prediction errors in terms of RMSE and MAPE, respectively. In terms of R^2, DFP-XGBoost achieves a 14.88% improvement compared with other methods.

Table 4. Performance of the machine learning models.

Phase	Performance Measurement Indices	DFP-XGBoost		ANN		SVR	
		Mean	Std	Mean	Std	Mean	Std
Training	RMSE	3.710	0.148	6.746	2.649	7.424	0.106
	MAPE (%)	4.780	0.183	8.857	4.040	9.852	0.164
	R^2	0.972	0.003	0.893	0.076	0.887	0.004
Testing	RMSE	5.270	0.834	8.823	1.990	10.461	1.204
	MAPE (%)	6.740	1.360	11.836	3.350	15.212	2.037
	R^2	0.942	0.020	0.820	0.096	0.763	0.053

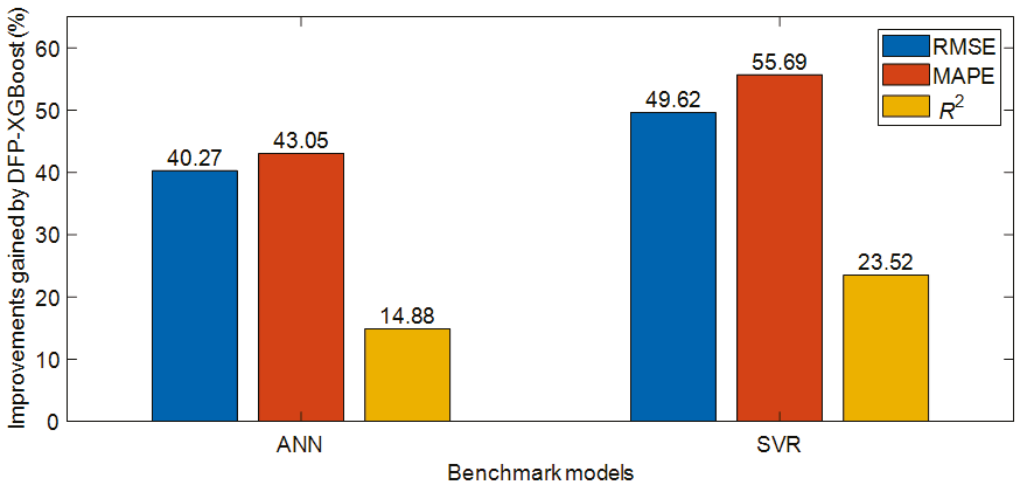


Figure 3. Improvements in performance measurement metrics.

Considering the aspect of computational cost, the DFP-based optimization requires roughly 236 s to complete its evolutionary process. However, the DFP-XGBoost's prediction phase can be executed very fast in roughly 0.5 s. The cross-validation processes used by the ANN and SVR require a computation time of 68.6 s and 13.3 s. The prediction phases of the ANN and SVR can be accomplished in 0.06 s and 0.70 s, respectively. It can be seen that DFP-XGBoost employs the DFP metaheuristic for model optimization. Hence, the computation time of its model construction phase is significantly longer than that of the benchmark approaches. The reason is that DFP is a population-based evolutionary algorithm; it necessitates a large number of cost function evaluations during the optimization process.

In addition, Figure 4 provides more insights regarding the distribution of the residual range (r) of the prediction models. It is noted that the range of residuals has been normalized by the magnitude of the actual CS values. Herein, the values of r obtained from each model are divided into five categories: (i) $r \leq 5\%$, (ii) $5\% < r \leq 10\%$, (iii) $10\% < r \leq 15\%$, (iv) $15\% < r \leq 20\%$, and (v) $r > 20\%$. As pointed out in [24], a relative error of less than 20% can be considered satisfactory for the task of CS estimation. Meanwhile, the relative error less than 10% indicates excellent prediction outcomes. Moreover, comparing the distribution of prediction errors can bring about better understanding of the predictive capability of the machine learning model [40].

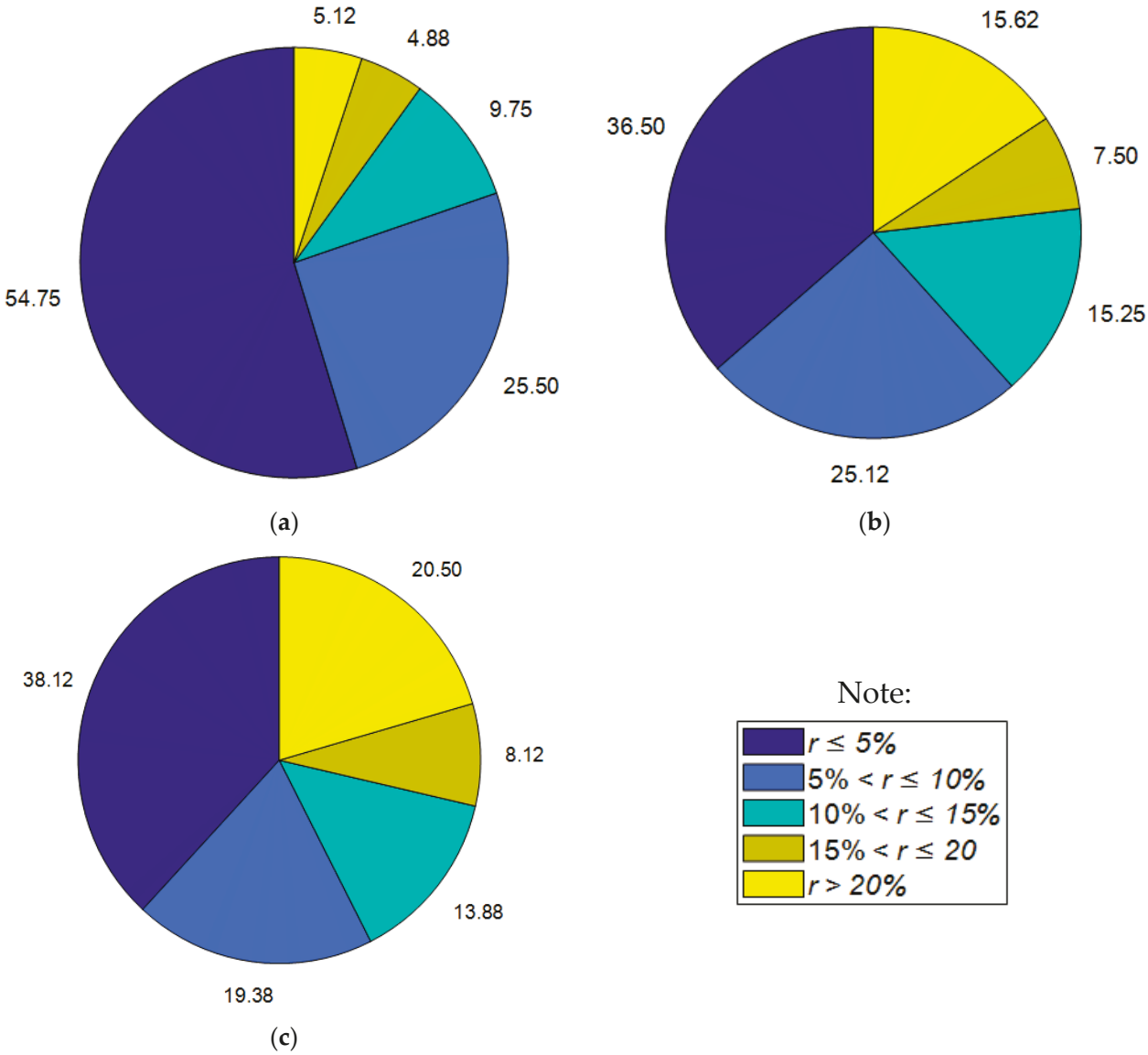


Figure 4. Distribution of residual ranges: (**a**) DFP-XGBoost, (**b**) ANN, and (**c**) SVR.

As can be seen from Figure 4, DFP-XGBoost achieves the highest proportion of the data samples in the first category (54.75%). This outcome is better than SVR (38.12%) and ANN (36.5%) by a large margin. The proportions of residual ranges in the second group of DFP-XGBoost and the ANN are relatively equal. Meanwhile, the number of residual ranges in (10%, 15%] yielded by the ANN is larger than that obtained from other models. Notably, 80.25% of the results predicted by DFP-XGBoost have an error range of less than 10%. The proportion of interest for the ANN and SVR models is 61.62% and 57.50%, respectively. Data samples in the last category only occupy 5.12% in the case of DFP-XGBoost. Meanwhile, this proportion of data samples yielded from the benchmark methods is at least three times larger than that of the proposed method. These analyses point out the highly satisfactory performance of DFP-XGBoost regarding the distribution of the residual range.

Furthermore, the boxplots in Figure 5 illustrate the minimum, first quartile, median, third quartile, and maximum of the RSME values obtained from the prediction models. A red line in each distribution denotes the median of the prediction errors. The median of prediction errors for DFP-XGBoost is 5.11 MPa, which is significantly below that of the ANN (8.47 MPa) and SVR (10.26 MPa). In addition, Wilcoxon signed-rank tests [59] are

used to compare the performance of the models used for predicting the CS of HPC. The threshold of the *p*-value of the tests is set to be 0.05. The hypothesis tests of DFP-XGBoost vs. the ANN and DFP-XGBoost vs. SVR both yield *p*-values of 0.0001, which is lower than the selected threshold. Therefore, there is sufficient confidence to reject the null hypothesis of equal performance and confirm the superiority of the proposed DFP-XGBoost over the benchmark approaches.

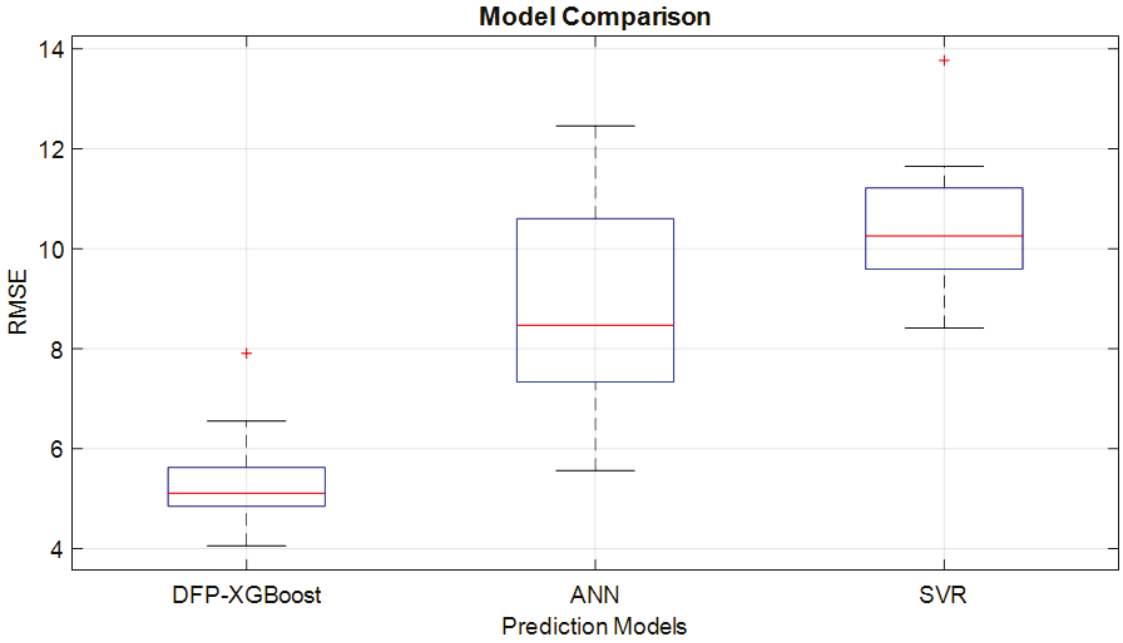


Figure 5. Performance comparison.

4.3. Sensitivity Analysis-Based Assessment of Feature Importance

For CS estimation, the relative influence of the independent variables on the response is helpful for the mixture design process. To clarify the influence of the predictor variables on the DFP-XGBoost's output, this study relies on the global analysis of the Fourier Amplitude Sensitivity Test (FAST) [60] as well as the SHAPley Additive exPlanations (SHAP) [61]. FAST is a popular method for revealing the relative influence of a model's input variables over its prediction accuracy [38]. Based on the test's result, the sensitivity values of the variables can be calculated to express their relative importance. In this study, the Python toolbox provided in [62] is used to perform the sensitivity test. Meanwhile, SHAP is the approach for elucidating the model's structure. This method is often used to express the effect of each feature on the predicted outcome. The SHAP approach is performed with the Python library developed by Lundberg [63].

The outcome of the analyses is presented in Figure 6. Based on FAST, four out of nine variables have significant effects on the CS of HPC, namely X_8 (quantity of WRA), X_4 (quantity of GGBFS), X_5 (quantity of fine aggregate), and X_6 (quantity of coarse aggregate). Concrete age (X_9), cement content (X_1), and silica fume (X_3) demonstrate moderate impacts on the response. Meanwhile, the relative importance of fly ash content and water content is lower than other variables. Based on SHAP, we are able to categorize the features into three groups according to their order of importance. The first group includes the curing age, quantity of WRA, quantity of GGBFS, and quantity of fine aggregate. The dosages of coarse aggregate, cement, and silica fume can be categorized into the second group. Meanwhile, the third group consists of water content and fly ash content. Thus, the results of SHAP are, to some degree, in agreement with those of FAST. Both methods emphasize the importance of the WRA in HPC mix design.

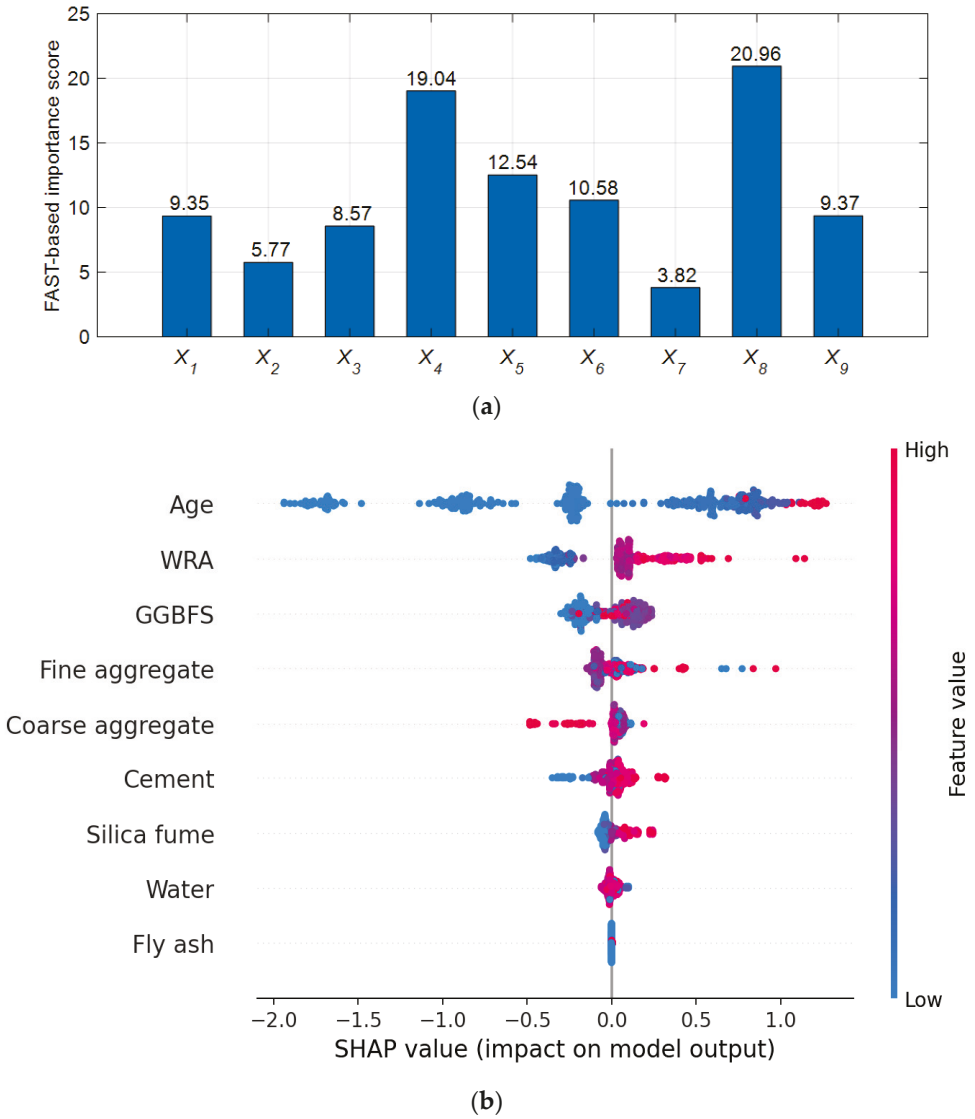


Figure 6. Results of feature importance analysis: (**a**) FAST and (**b**) SHAP.

4.4. Interval Prediction of Compressive Strength Based on Quantile Regression

The standard XGBoost regressor is capable of providing point estimates of the CS values. Although the previous section demonstrates that this model can achieve outstanding prediction results with an RMSE of 5.72 and an MAPE of 6.74%, it cannot yield information about the dispersion of the predictions around the CS value. This study resorts to quantile regression [64] for expressing a functional relationship between the independent variables (i.e., the mix's constituents and curing age) for all possible portions of a probability distribution. Quantile regression can be used to derive prediction intervals from a base model [65]. For instance, a 95% prediction interval for the estimated CS can be expressed as follows:

$$PI(x) = [Q_{0.025}(x), Q_{0.975}(x)] \tag{10}$$

where $Q_{0.025}(x)$ and $Q_{0.975}(x)$ are the two XGBoost models trained with the quantile regression parameters of 0.025 and 0.975, respectively.

Hence, to derive the interval estimation of the CS of HPC, it is required to train two separate XGBoost regressors that employ two quantile regression parameters. By doing so, DFP-XGBoost is able to yield a range estimation of the CS values in addition to their point

estimation. To construct these two models, the first-order derivative of the loss function of the model is revised as follows [66]:

$$\frac{\partial L}{\partial y} = \begin{cases} \zeta_{QR} \ \ if \ \ \varepsilon > 0 \\ \zeta_{QR} - 1 \ \ if \ \ \varepsilon < 0 \\ 0 \ \ if \ \ \varepsilon = 0 \end{cases} \tag{11}$$

where ζ_{QR} denotes the quantile regression's parameter; ε is the model's residual; y denotes the predicted CS value.

In addition, to evaluate the quality of prediction intervals, prediction interval coverage probability (PICP) can be computed [67]. PICP basically measures the proportion of the predicted CS values that lie within the lower and upper boundaries. In addition, the mean width of the prediction interval (MWPI) is an index that characterizes the width of the interval estimation. The PICP is expected to be as close to the specified confidence level as possible. Meanwhile, the MWPI should be reasonably narrow to convey useful information about the range estimation of the CS. This study constructs 95% interval estimations of the CS values by integrating XGBoost and the quantile regression method. Herein, the XGBoost model used for deriving the lower bound of the estimations is built with ζ_{QR} = 0.025. Meanwhile, the one that yields the upper bound of the estimations is trained with ζ_{QR} = 0.975. The prediction intervals for one experiment with 40 testing samples are illustrated in Figure 7. Considering the estimation results obtained from 20 runs, the average PICP and MWPI in the training phases are 96% and 28.49 MPa, respectively. These values obtained from the testing phases are 84% and 28.25 MPa. Hence, the MWPIs in the training and testing phases are relatively equal to each other. Although the PICP in the training phase is satisfactory, the result in the testing phase could not meet expectations. This result can be explained by the fact that the uncertainty of the response in the testing phase is much higher than that in the training phase. Therefore, the model has failed to reach a PICP of 95% for novel data samples.

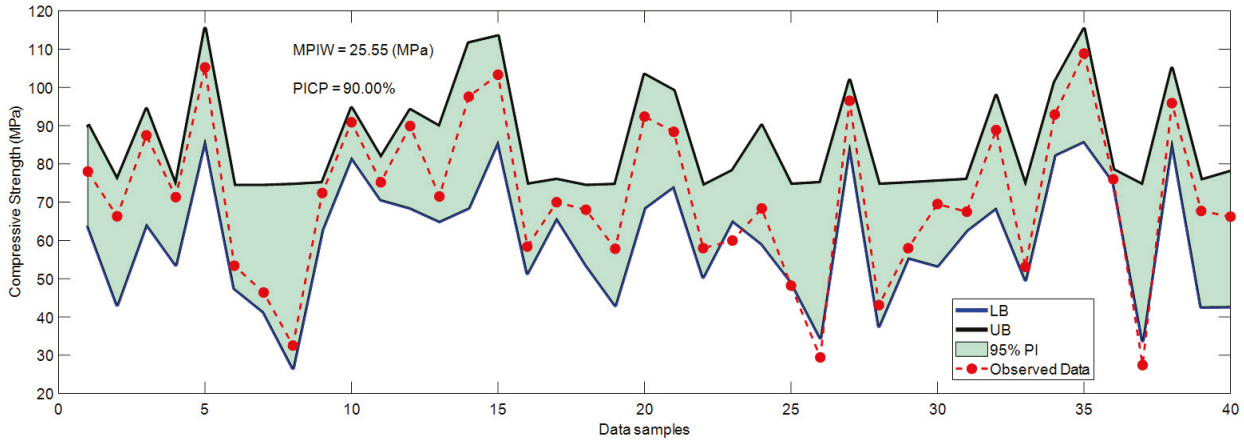


Figure 7. Prediction intervals constructed by XGBoost and quantile regression method.

4.5. Reduction in Overestimations Committed by XGBoost Regressor

In CS estimation, besides the goal of minimizing the prediction errors or residuals, enhancing the model's reliability is also of practical concern. The standard loss function used for training an XGBoost regressor (i.e., the SEL) is symmetric in nature. Therefore, it treats negative and positive errors similarly during the training process. Notably, the model residual (ε) is calculated as $t - y$, where t and y denote the actual and predicted CS values, respectively. If the actual CS is less than the estimated one, the residual is negative, and the model overestimates the target variable. This situation is undesirable because it leads to unsafe prediction outcomes. Therefore, negative residuals or overestimations should be avoided [39].

Because SEL is symmetric, the proportions of negative and positive residuals of the model trained by the XGBoost regressor using SEL are expected to be similar. By inspecting the actual and estimated CS values in multiple experiments, the percentage of negative residuals is found to be 47.12%. Therefore, the numbers of overestimated and underestimated results are relatively equal to each other. The distribution of the residuals obtained from the model using SEL is demonstrated in Figure 8. The mean, standard deviation, and skewness of the distribution are found to be 0.07, 5.34, and -0.87, respectively. Hence, the average of the errors is close to 0. In addition, a negative skewness of -0.87 indicates that the left tail of the distribution is longer than the right one. This fact is also confirmed by visually inspecting the histogram in Figure 8. To enhance the reliability of the model, it is desired to shorten the left tail of the distribution because data samples in this region are associated with overestimated results. Figure 9 shows more details of the model's residuals; the average magnitude of the negative residuals (4.13 MPa) is slightly higher than that of the positive ones (3.80 MPa).

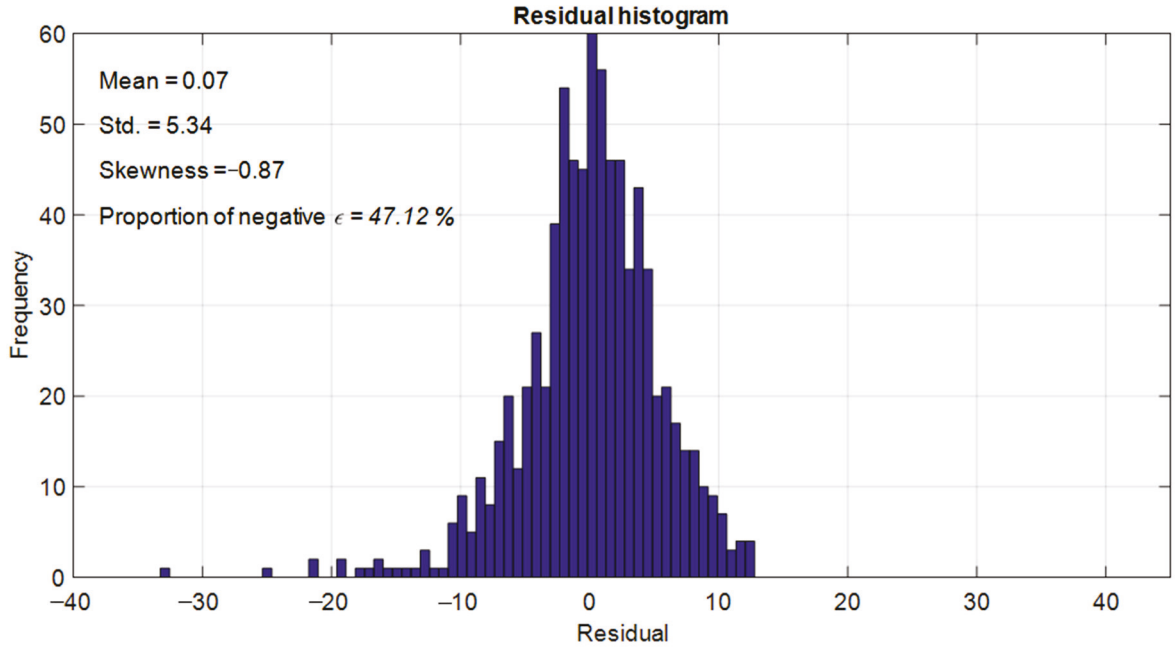


Figure 8. Distribution of residuals obtained from the model using SEL.

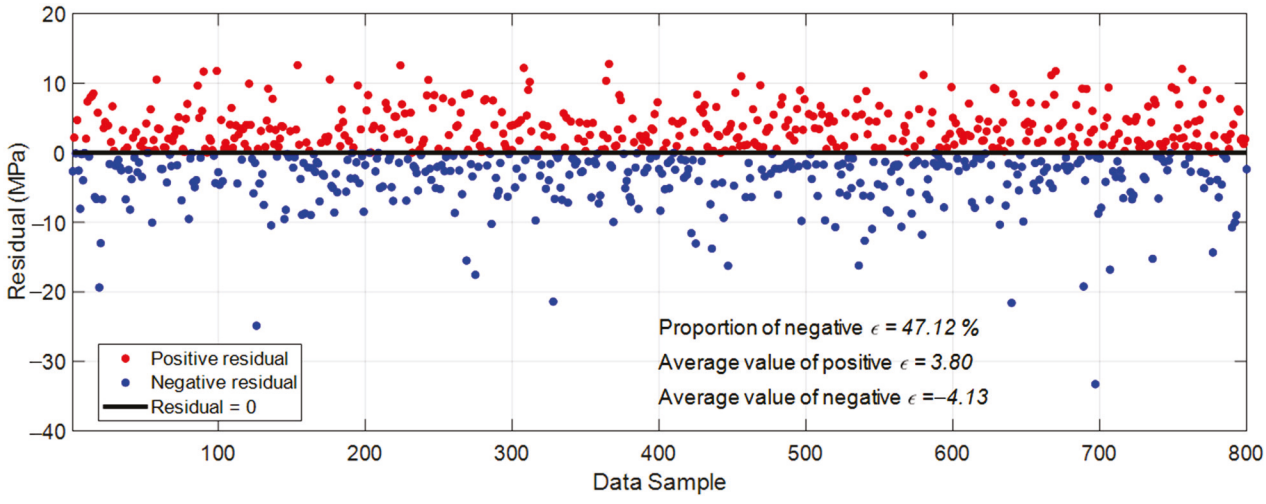


Figure 9. Magnitude of positive and negative residuals obtained from the model using SEL.

To express the bias against overestimations, the asymmetric squared error loss (ASEL) should be used for training XGBoost [40]. The ASEL function, denoted as L_{ASEL}, is given by:

$$L_{ASEL}(t,y) = \begin{cases} (t-y)^2, & if\ (t-y) \geq 0 \\ \rho \times (t-y)^2, & otherwise \end{cases} \tag{12}$$

where ρ is a tuning parameter of the function for controlling the degree of asymmetry. The larger the value of ρ, the more severe overestimations are penalized.

To implement the ASEL with the XGBoost regressor, its first-order and second-order derivatives of the loss function with respect to the predicted result (y) are revised in the following manner [40]:

First-order derivative:

$$L'(t,y) = \begin{cases} 2 \times (t-y), & if\ (t-y) \geq 0 \\ 2 \times \rho \times (t-y), & otherwise. \end{cases} \tag{13}$$

Second-order derivative:

$$L''(t,y) = \begin{cases} 2, & if\ (t-y) \geq 0 \\ 2 \times \rho, & otherwise \end{cases} \tag{14}$$

The ASEL generally imposes a large penalty on overestimated results. Therefore, the model is trained to maximize the number of positive residuals. However, fine-tuning of ρ is necessary because excessively large values of this parameter may significantly impair the overall prediction accuracy of the model. The experimental results with different values of ρ are reported in Figure 10. As ρ increases from 0 to 15, the proportion of underestimations gradually rises. However, the overall predictive capability of the model, expressed in terms of R^2, tends to be damaged. Observable from the figure, with ρ = 10, the percentage of positive residuals reaches 72.5% and the model can capture up to 85% of the variability in the CS of HPC. When the parameter of the ASEL surpasses 10, the gain in proportion of positive residuals is insignificant; meanwhile, the overall performance of the model deteriorates considerably. Therefore, ρ = 10 is selected for the XGBoost regressor in this study.

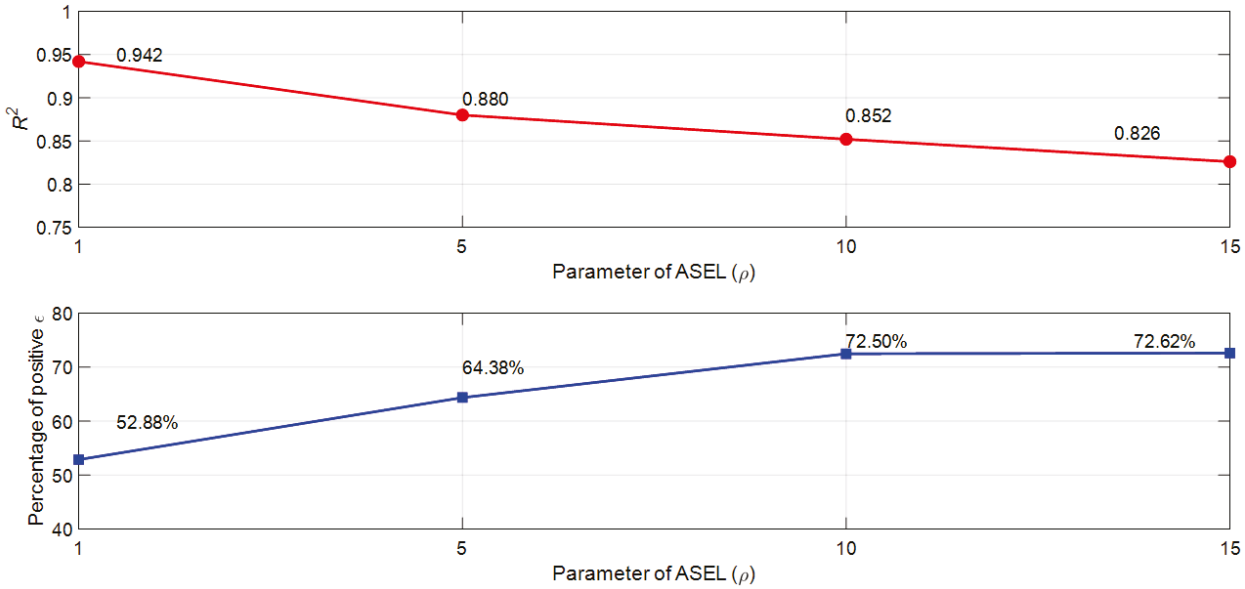


Figure 10. Model performance with respect to ρ.

Using the ASEL with $\rho = 10$, the performance of DFP-XGBoost regarding the characteristics of its residuals is significantly improved. As shown in Figure 11, the average value of the residuals increases from roughly 0.7 to 4.12. The bulk of the distribution shifts to the right side of the horizontal axis. Therefore, the number of overestimations is considerably reduced from 47.12% to 27.5%. In other words, underestimations account for 72.5% of the prediction. The skewness of the distribution also changes sign; this fact indicates that its right tale is longer than the left one. It is highly desirable for CS estimation because the magnitude of overestimation has been reduced. However, the use of the ASEL also causes a certain increase in the range of positive residuals. As demonstrated in Figure 12, the average range of positive residuals increases from 3.80 MPa to 7.12 MPa. On the contrary, the average range of negative ones decreases from 4.13 MPa to 3.80 MPa.

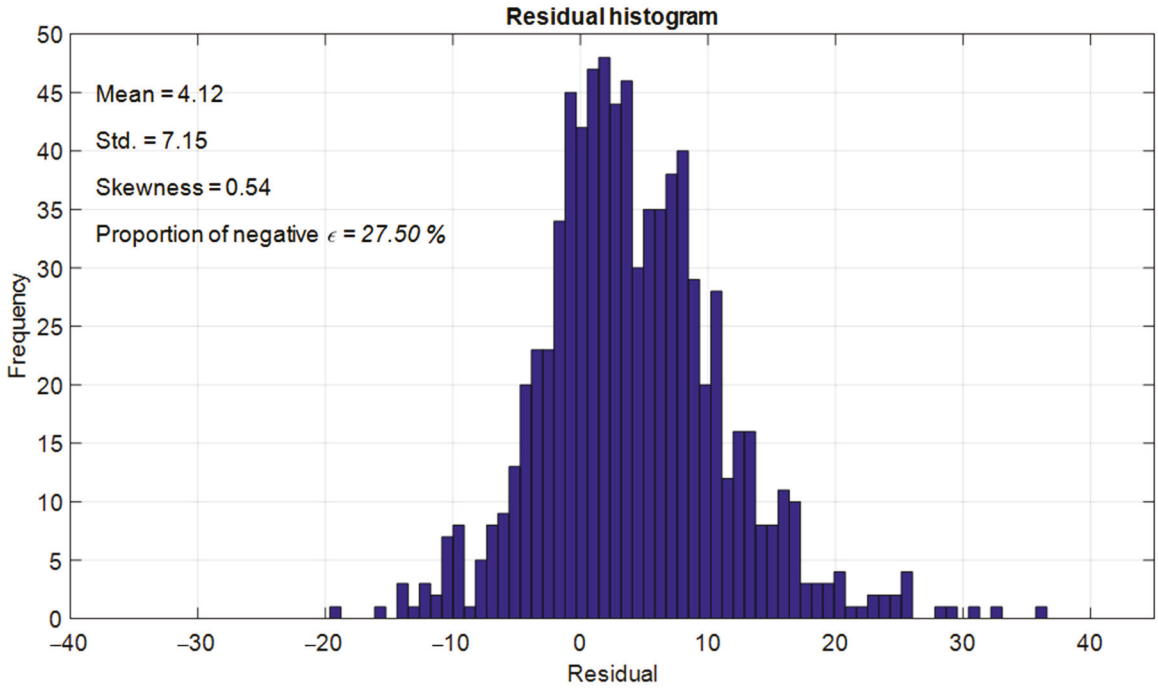


Figure 11. Distribution of residuals obtained from the model using ASEL.

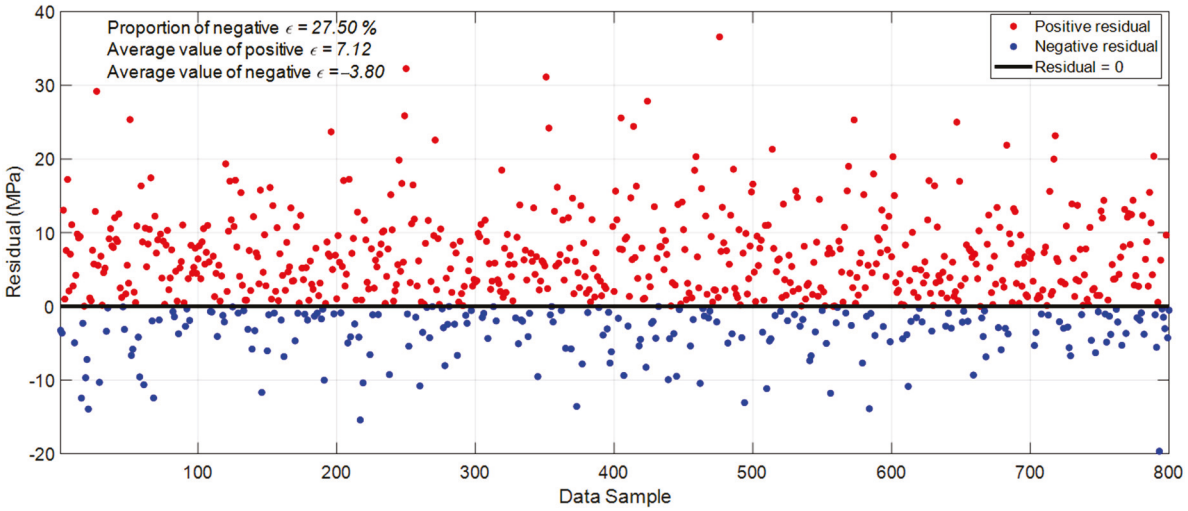


Figure 12. Magnitude of positive and negative residuals obtained from the model using ASEL.

The prediction results of DFP-XGBoost trained by the ASEL (denoted as DFP-XGBoost-ASEL) are demonstrated in Figure 13. As observed in the figure, the majority of the data points are located below the line of best fit. This fact accounts for the high proportion of underestimations. DFP-XGBoost-ASEL is able to achieve an RMSE of 8.13, an MAPE of 9.57%, and an R^2 of 0.852. Notably, the MAPE of the model is still less than 10%, which indicates good prediction performance in CS estimation [24]. Detailed performance of the models using the two loss functions are provided in Table 5. Based on the experimental results, the use of the ASEL brings about a reduction in the model's performance in both the training and testing phases. The Wilcoxon signed-rank test yields a p-value of $0.00009 < 0.05$, confirming that the overall predictive accuracy of DFP-XGBoost-ASEL is statistically lower than that of the one using the SEL. Nevertheless, the former also helps significantly restrict the occurrence of overestimations. Therefore, there is an inevitable trade-off between overestimation reduction and prediction error minimization.

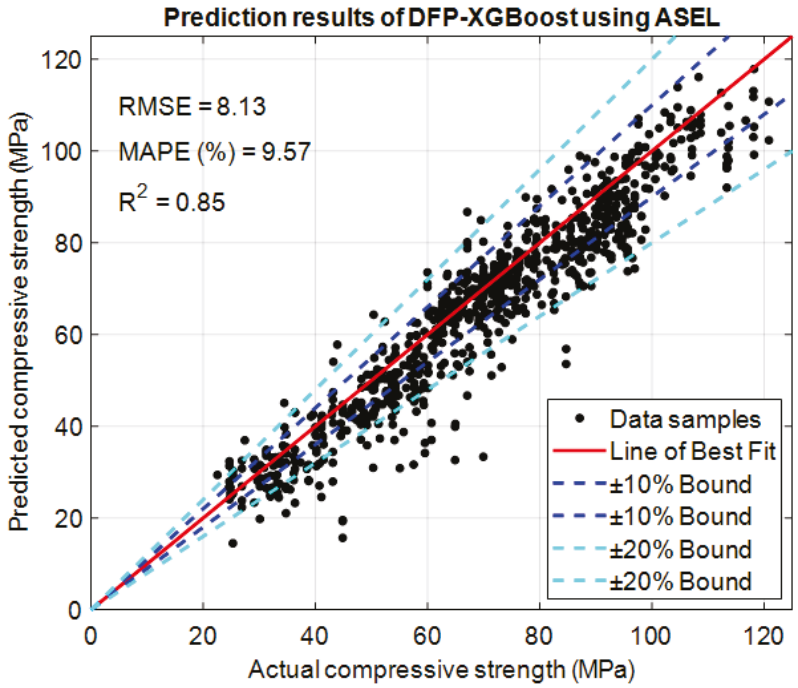


Figure 13. Prediction outcomes of DFP-XGBoost trained with ASEL.

Table 5. Comparison of the model's performance using SEL and ASEL.

Phase	Performance Measurement Indices	DFP-XGBoost Using SEL		DFP-XGBoost Using ASEL	
		Mean	Std	Mean	Std
Training	RMSE	3.710	0.148	6.280	0.648
	MAPE (%)	4.780	0.183	7.290	0.855
	R^2	0.972	0.003	0.919	0.017
Testing	RMSE	5.270	0.834	8.130	1.360
	MAPE (%)	6.740	1.360	9.570	1.630
	R^2	0.942	0.020	0.852	0.054

One notable observation is that although DFP-XGBoost-ASEL is inferior to the standard XGBoost regressor, it still outperforms other benchmark methods in terms of all measurement indices. Therefore, the proposed method is a promising alternative for assisting researchers and construction engineers in HPC mix design and prediction of the mix's mechanical properties. Furthermore, to ease the implementation of DFP-XGBoost, a graphical user interface (GUI) has been developed in Python (v. 3.10). The model's GUI

is provided in the Appendix A of this paper and the Github repository. It is noted that the program is constructed with the assistant of the tkinter package [68]. The developed program is able to provide the user with estimated CS values of untested HPC mixes. The required input information includes the mix's constituents and curing age. Both the SEL and ASEL can be selected in the GUI via the setting of the radio buttons.

5. Conclusions

The estimation of the CS of HPC is crucial for the tasks of mix design. This study has proposed and verified a hybrid data-driven approach named DFP-XGBoost for enhancing the accuracy as well as the reliability of the CS estimation model. In this integrated framework, XGBoost is employed as a regressor for generalizing a functional mapping between the CS of HPC mixes and their influencing variables. The quantities of cement, Supplementary Cementitious Materials (i.e., silica fume, GGBFS, fly ash), water, fine aggregate, coarse aggregate, and WRA are utilized as independent variables regarding the mixes' constituents. It is noted that forecasts of the CS at different curing times are also important for studying the developmental progress of the concrete's properties. In addition, this capability is also helpful for scheduling construction activities related to the removal of formwork systems and the re-shoring of reinforced concrete slabs. Hence, this study also considers the concrete age as an input feature of the machine learning model.

To automate the model construction phase, this study relied on the DFP metaheuristic algorithm. DFP helps carry out the model selection phase of XGBoost in an entirely data-driven manner. Based on the model's performance on the training dataset, DFP guides a population of the model's hyper-parameters to a good solution that minimizes the objective function. In this manner, the model's hyper-parameters, including the number of trees, learning rate, maximum tree depth, L_2-regularization coefficient, and L_1-regularization coefficient, can be adapted entirely by the collected data without the need for human intervention. A historical dataset consisting of 400 samples was gathered from previous works to construct DFP-XGBoost. After being trained, the model is able to yield estimated CS values for untested HPC mixes. Additionally, a program with a simple GUI is developed in Python to ease the implementation of the proposed approach.

Experimental results show that the hybrid model has achieved an outstanding result with an RMSE of 5.27, an MAPE of 6.74%, and an R^2 of 0.94. This performance is highly desirable because DFP-XGBoost is capable of explaining up to 94% of the variability in the CS of HPC. To reduce the number of overestimations, this study resorts to the ASEL function for training the XGBoost regressor. As can be seen from the experimental results, the use of the ASEL helps decrease the percentage of overestimated CS values by 19.62%. However, this reduction in overestimation is achieved by a certain compromise in the overall prediction accuracy. Moreover, XGBoost-based quantile regression for estimating the CS of HPC is also investigated in this study. By modifying the loss function of XGBoost, this model is capable of yielding prediction intervals of CS values at a certain confidence level. Accordingly, the proposed hybrid approach can be a capable and reliable tool for assisting with the tasks of mixture design and optimization. The newly developed DFP-XGBoost can be implemented easily and can be applied to model other important mechanical properties of HPC, such as tensile strength and modulus of elasticity.

Nevertheless, the current work also has several limitations. First, the number of mixtures containing fly ash in the dataset is limited. Only 8% of the samples use fly ash as one of the Supplementary Cementitious Materials. Second, the current paper has not considered the use of nanomaterials (e.g., nanosilica) and recycled aggregates in HPC. Third, the total number of samples in the current dataset is still limited. Fourth, other crucial factors, such as the size distribution of aggregates and curing conditions, have not been taken into consideration. Therefore, future work should focus on extending the current dataset by collecting more data samples containing fly ash and other potential factors. Fifth, although the range estimation of the training phase of DFP-XGBoost is qualified with a PICP of 96% for a confidence level of 95%, the prediction interval constructed in the testing

phase can only cover roughly 84% of the actual CS values. It is expected that by increasing the number of laboratory testing results in the database, the generalization capability of the model can be further improved. In addition, other advanced techniques for model optimization and range estimation can be explored to ameliorate the current performance. Finally, the capability of other potential approaches for estimating the CS of concrete mixes, such as the Group Method of Data Handling (GMDH), fuzzy logic, and deep learning, should be investigated to enhance the prediction accuracy.

Author Contributions: Conceptualization, N.-D.H. and X.-L.T.; methodology, N.-D.H., V.-D.T. and X.-L.T.; software, N.-D.H. and X.-L.T.; validation, X.-L.T., V.-D.T. and N.-D.H.; data curation, X.-L.T. and N.-D.H.; writing—original draft preparation, N.-D.H., V.-D.T. and X.-L.T.; writing—review and editing, X.-L.T. and N.-D.H.; visualization, X.-L.T.; supervision, V.-D.T. and N.-D.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are openly available in the Github repository at: https://github.com/NHDDTUEDU/HPC_DFP_XGBoost.

Conflicts of Interest: The authors have no competing interests to declare that are relevant to the content of this article.

Appendix A

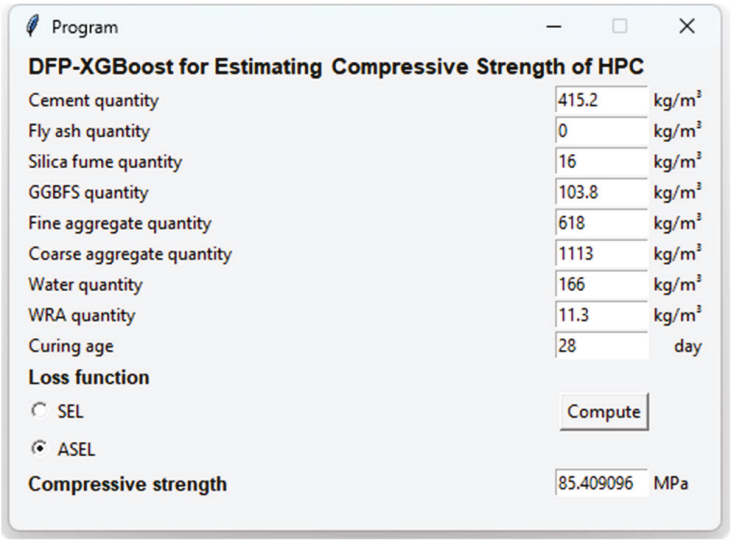


Figure A1. Graphical user interface of the proposed method for estimating compressive strength of high-performance concrete.

References

1. Van Damme, H. Concrete material science: Past, present, and future innovations. *Cem. Concr. Res.* **2018**, *112*, 5–24. [CrossRef]
2. Gagg, C.R. Cement and concrete as an engineering material: An historic appraisal and case study analysis. *Eng. Fail. Anal.* **2014**, *40*, 114–140. [CrossRef]
3. Chou, J.-S.; Pham, A.-D. Enhanced artificial intelligence for ensemble approach to predicting high performance concrete compressive strength. *Constr. Build. Mater.* **2013**, *49*, 554–563. [CrossRef]
4. Akhnoukh, A.K.; Buckhalter, C. Ultra-high-performance concrete: Constituents, mechanical properties, applications and current challenges. *Case Stud. Constr. Mater.* **2021**, *15*, e00559. [CrossRef]
5. Neville, A.; Aïtcin, P.-C. High performance concrete—An overview. *Mater. Struct.* **1998**, *31*, 111–117. [CrossRef]
6. Videla, C.; Gaedicke, C. Modeling portland blast-furnace slag cement high-performance concrete. *ACI Mater. J.* **2004**, *101*, 365–375.
7. Wu, L.; Farzadnia, N.; Shi, C.; Zhang, Z.; Wang, H. Autogenous shrinkage of high performance concrete: A review. *Constr. Build. Mater.* **2017**, *149*, 62–75. [CrossRef]
8. Xue, J.; Briseghella, B.; Huang, F.; Nuti, C.; Tabatabai, H.; Chen, B. Review of ultra-high performance concrete and its application in bridge engineering. *Constr. Build. Mater.* **2020**, *260*, 119844. [CrossRef]

9. Ben Chaabene, W.; Flah, M.; Nehdi, M.L. Machine learning prediction of mechanical properties of concrete: Critical review. *Constr. Build. Mater.* **2020**, *260*, 119889. [CrossRef]
10. Li, Q.-F.; Song, Z.-M. High-performance concrete strength prediction based on ensemble learning. *Constr. Build. Mater.* **2022**, *324*, 126694. [CrossRef]
11. Azkune, M.; Puente, I.; Santilli, A. Shore overloads during shoring removal. *Eng. Struct.* **2010**, *32*, 3629–3638. [CrossRef]
12. Lee, S.; Nguyen, N.-H.; Karamanli, A.; Lee, J.; Vo, T.P. Super learner machine-learning algorithms for compressive strength prediction of high performance concrete. *Struct. Concr.* **2023**, *24*, 2208–2228. [CrossRef]
13. Smarzewski, P. Influence of silica fume on mechanical and fracture properties of high performance concrete. *Procedia Struct. Integr.* **2019**, *17*, 5–12. [CrossRef]
14. Mazloom, M.; Ramezanianpour, A.A.; Brooks, J.J. Effect of silica fume on mechanical properties of high-strength concrete. *Cem. Concr. Compos.* **2004**, *26*, 347–357. [CrossRef]
15. Yeh, I.C. Modeling of strength of high-performance concrete using artificial neural networks. *Cem. Concr. Res.* **1998**, *28*, 1797–1808. [CrossRef]
16. Bilim, C.; Atiş, C.D.; Tanyildizi, H.; Karahan, O. Predicting the compressive strength of ground granulated blast furnace slag concrete using artificial neural network. *Adv. Eng. Softw.* **2009**, *40*, 334–340. [CrossRef]
17. Erdal, H.I. Two-level and hybrid ensembles of decision trees for high performance concrete compressive strength prediction. *Eng. Appl. Artif. Intell.* **2013**, *26*, 1689–1697. [CrossRef]
18. González, S.; García, S.; Del Ser, J.; Rokach, L.; Herrera, F. A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities. *Inf. Fusion* **2020**, *64*, 205–237. [CrossRef]
19. Gómez-Ríos, A.; Luengo, J.; Herrera, F. A Study on the Noise Label Influence in Boosting Algorithms: AdaBoost, GBM and XGBoost. In *Hybrid Artificial Intelligent Systems*; Springer International Publishing: Cham, Switzerland, 2017; pp. 268–280.
20. Chithra, S.; Kumar, S.R.R.S.; Chinnaraju, K.; Alfin Ashmita, F. A comparative study on the compressive strength prediction models for High Performance Concrete containing nano silica and copper slag using regression analysis and Artificial Neural Networks. *Constr. Build. Mater.* **2016**, *114*, 528–535. [CrossRef]
21. Yu, Y.; Li, W.; Li, J.; Nguyen, T.N. A novel optimised self-learning method for compressive strength prediction of high performance concrete. *Constr. Build. Mater.* **2018**, *184*, 229–247. [CrossRef]
22. Kaloop, M.R.; Kumar, D.; Samui, P.; Hu, J.W.; Kim, D. Compressive strength prediction of high-performance concrete using gradient tree boosting machine. *Constr. Build. Mater.* **2020**, *264*, 120198. [CrossRef]
23. Nguyen, H.; Vu, T.; Vo, T.P.; Thai, H.-T. Efficient machine learning models for prediction of concrete strengths. *Constr. Build. Mater.* **2021**, *266*, 120950. [CrossRef]
24. Chou, J.-S.; Chen, L.-Y.; Liu, C.-Y. Forensic-based investigation-optimized extreme gradient boosting system for predicting compressive strength of ready-mixed concrete. *J. Comput. Des. Eng.* **2022**, *10*, 425–445. [CrossRef]
25. Li, Z.; Yoon, J.; Zhang, R.; Rajabipour, F.; Srubar Iii, W.V.; Dabo, I.; Radlińska, A. Machine learning in concrete science: Applications, challenges, and best practices. *npj Comput. Mater.* **2022**, *8*, 127. [CrossRef]
26. Al Yamani, W.H.; Ghunimat, D.M.; Bisharah, M.M. Modeling and predicting the sensitivity of high-performance concrete compressive strength using machine learning methods. *Asian J. Civ. Eng.* **2023**, *24*, 1943–1955. [CrossRef]
27. Nguyen, T.-D.; Cherif, R.; Mahieux, P.-Y.; Lux, J.; Aït-Mokhtar, A.; Bastidas-Arteaga, E. Artificial intelligence algorithms for prediction and sensitivity analysis of mechanical properties of recycled aggregate concrete: A review. *J. Build. Eng.* **2023**, *66*, 105929. [CrossRef]
28. Singh, S.; Patro, S.K.; Parhi, S.K. Evolutionary optimization of machine learning algorithm hyperparameters for strength prediction of high-performance concrete. *Asian J. Civ. Eng.* **2023**, *24*, 3121–3143. [CrossRef]
29. Rathakrishnan, V.; Beddu, S.B.; Ahmed, A.N. Predicting compressive strength of high-performance concrete with high volume ground granulated blast-furnace slag replacement using boosting machine learning algorithms. *Sci. Rep.* **2022**, *12*, 9539. [CrossRef]
30. Hoang, N.D.; Tran, D.V. Machine learning-based estimation of concrete compressive strength: A multi-model and multi-dataset study. *Civ. Eng. Infrastruct. J.* **2023**. [CrossRef]
31. Al Adwan, J.; Alzubi, Y.; Alkhdour, A.; Alqawasmeh, H. Predicting Compressive Strength of Concrete Using Histogram-Based Gradient Boosting Approach for Rapid Design of Mixtures. *Civ. Eng. Infrastruct. J.* **2023**, *56*, 159–172. [CrossRef]
32. Ranjbar, I.; Toufigh, V.; Boroushaki, M. A combination of deep learning and genetic algorithm for predicting the compressive strength of high-performance concrete. *Struct. Concr.* **2022**, *23*, 2405–2418. [CrossRef]
33. Ben Seghier, M.E.A.; Golafshani, E.M.; Jafari-Asl, J.; Arashpour, M. Metaheuristic-based machine learning modeling of the compressive strength of concrete containing waste glass. *Struct. Concr.* **2023**, *24*, 5417–5440. [CrossRef]
34. Huang, Y.; Lei, Y.; Luo, X.; Fu, C. Prediction of compressive strength of rice husk ash concrete: A comparison of different metaheuristic algorithms for optimizing support vector regression. *Case Stud. Constr. Mater.* **2023**, *18*, e02201. [CrossRef]
35. Mohammadizadeh, M.R.; Esfandnia, F. Predicting Compression Strength of Reinforced Concrete Columns Confined by FRP Using Meta-Heuristic Methods. *Civ. Eng. Infrastruct. J.* **2022**, *55*, 1–17. [CrossRef]
36. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794. [CrossRef]

37. Hoang, N.-D.; Tien Bui, D.; Liao, K.-W. Groutability estimation of grouting processes with cement grouts using Differential Flower Pollination Optimized Support Vector Machine. *Appl. Soft Comput.* **2016**, *45*, 173–186. [CrossRef]
38. Hoang, N.-D. Compressive Strength Estimation of Rice Husk Ash-Blended Concrete Using Deep Neural Network Regression with an Asymmetric Loss Function. *Iran. J. Sci. Technol. Trans. Civ. Eng.* **2022**, *47*, 1547–1565. [CrossRef]
39. Pham, A.-D.; Hoang, N.-D.; Nguyen, Q.-T. Predicting Compressive Strength of High-Performance Concrete Using Metaheuristic-Optimized Least Squares Support Vector Regression. *J. Comput. Civ. Eng.* **2016**, *30*, 06015002. [CrossRef]
40. Hoang, N.-D. A novel ant colony-optimized extreme gradient boosting machine for estimating compressive strength of recycled aggregate concrete. *Multiscale Multidiscip. Model. Exp. Design.* **2023**, *7*, 375–394. [CrossRef]
41. Bentéjac, C.; Csörgő, A.; Martínez-Muñoz, G. A Comparative Analysis of XGBoost. *arXiv* **2019**, arXiv:1911.01914.
42. Storn, R.; Price, K. Differential Evolution—A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
43. Yang, X.-S. Flower Pollination Algorithm for Global Optimization. In *Unconventional Computation and Natural Computation*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 240–249.
44. Tien Bui, D.; Hoang, N.-D.; Samui, P. Spatial pattern analysis and prediction of forest fire using new machine learning approach of Multivariate Adaptive Regression Splines and Differential Flower Pollination optimization: A case study at Lao Cai province (Viet Nam). *J. Environ. Manag.* **2019**, *237*, 476–487. [CrossRef]
45. Pavlů, T.; Fořtová, K.; Mariaková, D.; Řepka, J.; Vlach, T.; Hájek, P. High-performance concrete with fine recycled concrete aggregate: Experimental assessment. *Struct. Concr.* **2023**, *24*, 1868–1878. [CrossRef]
46. Fallah-Valukolaee, S.; Mousavi, R.; Arjomandi, A.; Nematzadeh, M.; Kazemi, M. A comparative study of mechanical properties and life cycle assessment of high-strength concrete containing silica fume and nanosilica as a partial cement replacement. *Structures* **2022**, *46*, 838–851. [CrossRef]
47. Li, J.; Tian, P. Effect of slag and silica fume on mechanical properties of high strength concrete. *Cem. Concr. Res.* **1997**, *27*, 833–837. [CrossRef]
48. Wang, L.; Yu, Z.; Liu, B.; Zhao, F.; Tang, S.; Jin, M. Effects of Fly Ash Dosage on Shrinkage, Crack Resistance and Fractal Characteristics of Face Slab Concrete. *Fractal Fract.* **2022**, *6*, 335. [CrossRef]
49. Sun, Y.; Liu, Z.; Ghorbani, S.; Ye, G.; De Schutter, G. Fresh and hardened properties of alkali-activated slag concrete: The effect of fly ash as a supplementary precursor. *J. Clean. Prod.* **2022**, *370*, 133362. [CrossRef] [PubMed]
50. Reddy, P.V.R.K.; Ravi Prasad, D. A study on workability, strength and microstructure characteristics of graphene oxide and fly ash based concrete. *Mater. Today Proc.* **2022**, *62*, 2919–2925. [CrossRef]
51. Sun, J.; Kong, K.H.; Lye, C.Q.; Quek, S.T. Effect of ground granulated blast furnace slag on cement hydration and autogenous healing of concrete. *Constr. Build. Mater.* **2022**, *315*, 125365. [CrossRef]
52. Aggarwal, C.C. *Neural Networks and Deep Learning*; Springer: Berlin/Heidelberg, Germany, 2018; ISBN 978-3-319-94463-0.
53. Ly, H.-B.; Nguyen, M.H.; Pham, B.T. Metaheuristic optimization of Levenberg–Marquardt-based artificial neural network using particle swarm optimization for prediction of foamed concrete compressive strength. *Neural Comput. Appl.* **2021**, *33*, 17331–17351. [CrossRef]
54. Nazari, A.; Sanjayan, J.G. Modelling of compressive strength of geopolymer paste, mortar and concrete by optimized support vector machine. *Ceram. Int.* **2015**, *41 Pt B*, 12164–12177. [CrossRef]
55. XGBoost XGBoost Documentation. Available online: https://xgboostreadthedocsio/en/stable/indexhtml (accessed on 30 December 2021).
56. Beale, M.H.; Hagan, M.T.; Demuth, H.B. Neural Network Toolbox User's Guide. The MathWorks, Inc. 2018. Available online: https://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf (accessed on 28 April 2018).
57. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
58. Wong, T.; Yeh, P. Reliable Accuracy Estimates from k-Fold Cross Validation. *IEEE Trans. Knowl. Data Eng.* **2020**, *32*, 1586–1594. [CrossRef]
59. Conover, W.J. *Practical Nonparametric Statistics*; John Wiley & Sons, Inc.: New York, NY, USA, 1999; ISBN 0-471-16068-7.
60. McRae, G.J.; Tilden, J.W.; Seinfeld, J.H. Global sensitivity analysis—A computational implementation of the Fourier Amplitude Sensitivity Test (FAST). *Comput. Chem. Eng.* **1982**, *6*, 15–25. [CrossRef]
61. Lundberg, S.M.; Lee, S.-I. A unified approach to interpreting model predictions. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
62. Herman, J.; Usher, W. SALib: An open-source Python library for sensitivity analysis. *J. Open Source Softw.* **2017**, *2*, 97. [CrossRef]
63. Lundberg, S. An Introduction to Explainable AI with Shapley Values. 2018. Available online: https://shapreadthedocsio/en/latest/indexhtml (accessed on 27 February 2024).
64. Koenker, R.; Bassett, G. Regression Quantiles. *Econometrica* **1978**, *46*, 33–50. [CrossRef]
65. Meinshausen, N. Quantile Regression Forests. *J. Mach. Learn. Res.* **2006**, *7*, 983–999.
66. Fahrmeir, L.; Kneib, T.; Lang, S.; Marx, B. *Regression: Models, Methods and Applications*; Springer: Berlin/Heidelberg, Germany, 2013. [CrossRef]

67. Khosravi, A.; Nahavandi, S.; Creighton, D. Construction of Optimal Prediction Intervals for Load Forecasting Problems. *IEEE Trans. Power Syst.* **2010**, *25*, 1496–1503. [CrossRef]
68. Python Tkinter—Python Interface to Tcl/Tk. 2023. Available online: https://docs.python.org/3/library/tkinter.html (accessed on 7 May 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Wind Energy Production in Italy: A Forecasting Approach Based on Fractional Brownian Motion and Generative Adversarial Networks

Luca Di Persio [1], Nicola Fraccarolo [2,*] and Andrea Veronese [1]

[1] Department of Computer Science, University of Verona, 37134 Verona, Italy; luca.dipersio@univr.it (L.D.P.); andrea.veronese@univr.it (A.V.)
[2] Department of Mathematics, University of Trento, 38123 Trento, Italy
* Correspondence: nicola.fraccarolo@unitn.it

Abstract: This paper focuses on developing a predictive model for wind energy production in Italy, aligning with the ambitious goals of the European Green Deal. In particular, by utilising real data from the *SUD* (South) Italian electricity zone over seven years, the model employs stochastic differential equations driven by (fractional) Brownian motion-based dynamic and generative adversarial networks to forecast wind energy production up to one week ahead accurately. Numerical simulations demonstrate the model's effectiveness in capturing the complexities of wind energy prediction.

Keywords: energy forecasting; generative adversarial networks; machine learning; renewable energies; stochastic differential equations

MSC: 60G18; 60G22; 60H10; 60H35; 60J65; 62M10; 65C20; 65C30; 68T07

1. Introduction

In recent years, the global pursuit of sustainable energy solutions has accelerated, driven by growing concerns over climate change and the reduction of fossil fuel resources. Within this context, renewable energy sources have emerged as pivotal components in the transition towards a more sustainable future. Among these, wind energy has garnered significant attention due to its inherent sustainability, non-polluting nature, and renewable abundance [1]. In alignment with international sustainability goals, e.g., the European Green Deal, which aims to achieve climate neutrality by 2050, wind-based power generation has become increasingly imperative [2].

In addition to the generation of renewable energy, accurately predicting energy consumption and effectively managing the electric network system are critical aspects of ensuring a reliable and efficient energy supply [3–5]. Precise forecasting of wind power generation can aid in balancing supply and demand, optimising grid operations, and minimising energy waste.

In Italy, wind energy production has experienced constant growth in the last decade, reflecting the broader global trend towards renewable energy adoption. According to Terna (https://www.terna.it/en/electric-system/dispatching/renewable-sources, accessed on 18 May 2024), the Italian electricity transmission system operator, wind energy has seen a significant increase, with a rise of 3 GW in installed capacity over the last 10 years. Moreover, production is predominantly concentrated in the southern regions of Italy, due to favourable climate conditions (Figure 1).

Wind's inherently stochastic, intermittent, and volatile nature poses significant obstacles to accurately forecasting its energy production [6]. As such, developing robust prediction models for wind energy production has become a crucial area of research and innovation.

Predictive modelling typically falls into one of four categories: point forecasting, interval forecasting, probabilistic forecasting, and scenario generation. Each approach offers distinct advantages and challenges.

Various techniques can be classified into [7]: physical methods, based on numerical weather prediction [8]; statistical methods, such as the class of autoregressive moving average (ARMA) and its modifications [9–11]; machine-learning-based methods, that can deal with irregular and nonlinear features of wind power series [12–22]; and hybrid approaches, which combine different technologies to improve the prediction accuracy [23–27].

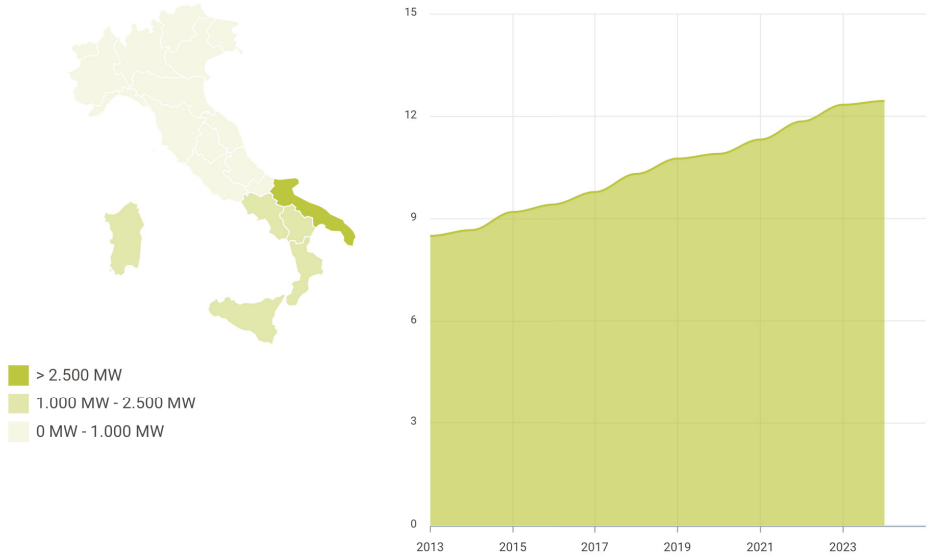


Figure 1. (**Left**): distribution of wind energy production in Italy. (**Right**): wind power capacity (GW) from 2013 to 2024. Source: Terna.

This paper focuses on point forecasting, while incorporating elements of probabilistic modelling. Specifically, the time series of wind energy production is considered a realisation of a stochastic process representing the solution of a stochastic differential Equation (SDE) driven by Brownian motion. This solution, a probability distribution on the path space of possible realisations of wind energy production, is then numerically approximated by a generative adversarial network (GAN) [28]. Subsequently, a generalisation based on fractional Brownian motion (fBm) is examined to determine whether the time series of wind energy production exhibits self-similarity properties.

GANs [29] are utilised in various fields, including image-to-image translation [30], natural language processing [31], and many others. A detailed introduction to GANs is provided in Section 2.1.3. Several notable variations of the original GAN architecture are reviewed, each introducing unique modifications and innovations to address specific challenges and enhance generative modelling capabilities.

The Wasserstein GAN (WGAN) [32] introduces the Wasserstein distance for measuring the difference between the generated and real data distributions. By minimising this distance, WGAN aims to stabilise the training process and produce higher-quality samples than traditional GANs. The Conditional GAN (CGAN) [32] extends the GAN framework by conditioning the generator and discriminator networks on additional information, such as class labels or auxiliary data. This enables the generation of samples conditioned on specific attributes, leading to more controlled and targeted synthesis. Ref. [33] provides probabilistic estimates of the generated data. Ref. [34] improves the original WGAN by introducing a gradient penalty term to enforce smoothness in the learned distribution. This regularisation technique helps to alleviate mode collapse and stabilises the training process. Combining the elements of both GANs and Variational Autoencoders (VAEs) to leverage the strengths of both frameworks, ref. [35] offers improved stability and control

over the generated data distribution. Among the hybrid methods involving GANs, in [36], a CGAN is followed by Single-objective Optimisation; this approach consists of training a CGAN to generate conditional samples, followed by a single-objective optimisation step to refine the generated samples further. Also, Ref. [37] is worth mentioning, where a WGAN is combined with a convolutional neural network (CNN) classifier to improve the discriminative capabilities of the discriminator network. The Conditional WGAN-GP can also be integrated with a Support Vector Classifier to enhance discrimination, as in [38]. Finally, the Sequence GAN (SeqGAN) [39] adopts LSTM to capture the temporal correlation, and then uses GANs coupled with reinforcement learning.

Given the objective of adopting a stochastic approach based on SDEs for modelling wind power generation, GANs provide a robust and flexible framework. Wind power generation is characterised by high volatility and inherent randomness, which makes traditional deterministic models insufficient. GANs are particularly well-suited for this task due to their ability to capture and generate diverse, complex data patterns through adversarial training. By using a GAN, the probability distribution of wind power generation trajectories can be effectively approximated, allowing us to incorporate the stochastic nature of the process [37,40]. This work is inspired by [28].

The paper's main contributions are as follows:

- A stochastic approach to wind energy production forecasting in Italy, with a forward prediction horizon of one week, has been adopted. The time series is treated as an outcome of a stochastic process, a solution of an SDE.
- The solution distribution is modelled by a GAN whose generator is driven by a Brownian motion. In addition to [28], the generator is also allowed to be driven by a fractional Brownian motion (fBm) with an experimentally estimated Hurst exponent. The results obtained by the two frameworks are then compared. Similar results are found in [41], where both models show a very close complexity.

The paper is structured as follows: in Section 2, after a brief recall of fBm and SDEs basics, the architecture is presented; Section 3 presents the data preprocessing, the training procedure and the results obtained; and finally, Section 4 concludes the paper.

2. Methodology

2.1. Preliminaries

This section discusses the preliminaries underlying fBm, SDEs, and GANs, as well as notably weak and strong solutions and GANs. By incorporating fBm into the model, the aim is to capture and leverage the self-similarity structure of the wind energy production time series.

2.1.1. Fractional Brownian Motion

Fractional Brownian motion (fBm) is a stochastic process characterised by long-range dependence and self-similarity. Unlike traditional Brownian motion, which exhibits independent and identically distributed increments, fBm displays correlated increments with a power-law decay in autocovariance.

Key properties of fBm include its non-stationary behaviour, continuous sample paths, and self-similarity across different time scales. The Hurst exponent quantifies the self-similarity property, denoted by H, which measures the degree of dependence between distant points in the time series. A Hurst exponent $H > 0.5$ indicates persistent behaviour, while $H < 0.5$ suggests anti-persistent behaviour.

2.1.2. Definition of SDE

Let $(\Omega, \mathcal{F}, P)$ be a probability space, and let $\{W_t^H\}_{t\geq 0}$ be a fBm on $\mathbb{R}$, adapted to its natural filtration $\mathcal{F}_t := \sigma(W_s^H : s \leq t)$.

Consider a one-dimensional SDE of Itô type:

$$dX_t = b(t, X_t)dt + \sigma(t, X_t)dW_t^H, \quad X_0 = x_0, \tag{1}$$

where $\{X_t\}_{t\geq 0}$ is a continuous-time stochastic process on $\mathbb{R}$ adapted to $\mathcal{F}_t$; $b(t, X_t)$ and $\sigma(t, X_t)$ are $\mathcal{F}_t$-measurable processes. One could rewrite the SDE equivalently in its integral form, namely

$$X_t = x_0 + \int_0^t b(s, X_s)ds + \int_0^t \sigma(s, X_s)dW_s^H. \tag{2}$$

A realisation of the process $\{X_t\}$ is called a path. Let us remark that a path is completely defined once the fBm $\{W_t^H\}$ has taken a realisation on the respective time interval. Given an fBm $\{W_t^H\}$, $\{X_t\}$ is is said to be a strong solution if $\{X_t\}$ is a continuous-time stochastic process adapted to $\mathcal{F}_t$, such that it solves Equation (2), whereas $\{X_t\}$ is said to be a weak solution if there exists a fBm $\{W_t^H\}$ such that $\{X_t\}$ is $\mathcal{F}_t$-adapted and solves (2). Path-wise uniqueness holds if any two strong solutions $\{X_t\}$, $\{X_t'\}$ on $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}, P, \{W_t\})$ are indistinguishable, namely

$$P(X_t = X_t' \quad \text{for every } t) = 1,$$

i.e., the paths corresponding to the solution are P-a.s. indistinguishable. On the other hand, weak uniqueness holds if any two strong $\{X_t\}$, $\{X_t'\}$ have the same law, namely

$$P(X_t = X_t') = 1 \quad \text{for a.e. } t.$$

For the sake of completeness, let us recall that the previous problem admits a unique, strong solution if, e.g., $b(t, X_t)$ and $\sigma(t, X_t)$ satisfy Lipschitz conditions, with X_0 being independent of $\mathcal{F}_t$. The process $\{X_t\}$ is square-integrable for all t; see, e.g., ref. [42] for details. In the following, the focus is restricted to SDEs that admit a strong solution.

2.1.3. Generative Adversarial Networks

Generally speaking, generative neural network (NN) structures are among the most relevant methods to generate synthetic data, including wind energy production datasets, mainly because of their capability to model complex data distributions. These architectures learn the underlying probability distribution of the data and generate samples that resemble real observations. Mathematically, a generative NN aims to learn a mapping function $G : \mathcal{Z} \to \mathcal{X}$, where $\mathcal{Z}$ is the input noise space and $\mathcal{X}$ is the data space. GANs have shown remarkable success in generating high-quality synthetic data within this context.

A GAN is an artificial intelligence model designed to generate new data that closely resembles a given dataset. It was introduced by Ian Goodfellow and their colleagues in 2014. GANs belong to the broader category of generative models, which aim to learn a given dataset's underlying patterns and structures to generate new, realistic samples from that data distribution. Accordingly, a generative NN is trained to minimise the discrepancy between the empirical distribution of the training data and the generated distribution.

The key idea behind GANs is to train two neural networks, a generator G and a discriminator D, in a competitive framework. The generator takes random noise as input and generates synthetic data samples. The generator's goal is to create indistinguishable data from real data. The discriminator evaluates input data and tries to distinguish between real data from the dataset and synthetic data produced by the generator. The discriminator's objective is to classify whether a given sample is real or generated correctly. The training process involves a back-and-forth competition between the generator and the discriminator: the generator tries to improve its ability to produce realistic data to fool the discriminator. Conversely, the discriminator aims to enhance its ability to differentiate between real and generated data.

Suppose the generator G_θ is parametrised by $\theta \in \Theta$ and the discriminator D_ϕ is parametrised by $\phi \in \Phi$. Then, consider a noise distribution μ on a space $\mathcal{X}$ and a target distribution ν on a space $\mathcal{Y}$. The generator is learned function $G_\theta \colon \mathcal{X} \to \mathcal{Y}$ trained so

that the (pushforward) distribution $G_\theta(\mu)$ approximates ν. The generator is optimised concerning

$$\min_\theta \mathbb{E}[D_\phi(G_\theta(\mu))],$$

i.e., it minimises a loss function that encourages the generated distribution to be similar to the target distribution. The discriminator is optimised concerning

$$\max_\phi \Big(\mathbb{E}[D_\phi(G_\theta(\mu))] - \mathbb{E}[D_\phi(\nu)]\Big),$$

i.e., it maximises a score function that measures its ability. The loss function of the GAN is, therefore,

$$V(G_\theta, D_\phi) = \mathbb{E}[D_\phi(G_\theta(\mu))] - \mathbb{E}[D_\phi(\nu)],$$

and an equilibrium is reached when the data produced by the generator cannot be distinguished from real data. The discriminator is unable to reliably tell the difference, which translates into the following two-player zero-sum minimax game:

$$\inf_\theta \sup_\phi V(G_\theta, D_\phi).$$

The training process is performed via stochastic gradient descent techniques, and it involves iteratively updating the parameters of the generator and discriminator networks to reach an equilibrium, where the generator produces samples that are indistinguishable from real data according to the discriminator.

2.1.4. SDEs as GANs

The strong solution to an SDE may be thought of as the unique function F, such that $F(X_0, W^H) = X$ almost surely (see Chapter V, Definition 10.9 in [43]). In other words, this entails that SDEs take a noise distribution W^H (Wiener measure, the distribution of fBm) and return some solution distribution, which is a probability distribution on path space.

In this way, the strong solution of an SDE and the functioning of a GAN share conceptual similarities. In both cases, there is a mapping from a source of randomness (noise) to a more complex structure, and both are concerned with capturing or replicating certain probability distributions.

Having established this linkage, the next objective is to combine these two methods.

2.2. Architecture

2.2.1. Generator

Let Y_{true} be a random variable on d_y-dimensional path space, i.e., the space of continuous functions $f : [0, T] \to \mathbb{R}^{d_y}$ for some fixed time horizon $T > 0$. Let $W^H : [0, T] \to \mathbb{R}^{d_w}$ be a d_w-dimensional fBm, with $d_w = 1$ whenever $H \neq \frac{1}{2}$, and $V \sim \mathcal{N}(0, I_{d_v})$ be drawn from a d_v-dimensional standard multivariate normal. The values d_w, d_v are hyperparameters describing the noise size. Let

$$\begin{aligned}
&\zeta_\theta : \mathbb{R}^{d_v} \to \mathbb{R}^{d_x},\\
&\mu_\theta : [0, T] \times \mathbb{R}^{d_x} \to \mathbb{R}^{d_x},\\
&\sigma_\theta : [0, T] \times \mathbb{R}^{d_x} \to \mathbb{R}^{d_x \times d_w},\\
&\alpha_\theta \in \mathbb{R}^{d_y \times d_x},\\
&\beta_\theta \in \mathbb{R}^{d_y},
\end{aligned}$$

where ζ_θ, μ_θ, and σ_θ are (Lipschitz) feedforward neural networks (FFNNs). See Section 2.2.4 for a formal definition of a Lipschitz neural network.

They are parametrised by θ. The hyperparameter d_x describes the size of the hidden state. By plugging the latter nets into an SDE in place of the corresponding functions and parameters, a neural SDE is defined:

$$\begin{aligned} X_0 &= \zeta_\theta(V), \\ dX_t &= \mu_\theta(t, X_t)dt + \sigma_\theta(t, X_t)dW_t^H, \quad (3) \\ Y_t &= \alpha_\theta X_t + \beta_\theta, \quad (4) \end{aligned}$$

for $t \in [0, T]$, with $X : [0, T] \to \mathbb{R}^x$ being the (strong) solution to the SDE, with the goal that $Y \approx Y_{\text{true}}$, according to a metric that will be defined later. Assuming Lipschitz continuity for μ_θ and σ_θ, solution X exists, and is also uniquely defined. As an SDE solution, X is a Markov process, even though the real data might not satisfy a Markov property. That is why X is thought of as a hidden state and transformed into Y, representing the real data process. To generate a path, an initial noise is sampled from V and from the fBm W^H; then, Equation (3) is solved with the Euler–Maruyama method.

Neural differential equations represent a powerful framework that integrates neural networks with both deterministic and stochastic processes. The reader is referred to [44–51] for more insights on neural differential equations and applications.

2.2.2. Discriminator

The discriminator evaluates input data, comparing real data from the dataset to synthetic data produced by the generator, aiming at correctly classifying whether a given sample is real or generated.

In this neural SDE-based approach, the previous concept translates into writing an application that takes as input paths $Y\colon [0, T] \to \mathbb{R}^{d_y}$, and is defined as neural SDE, allowing the model to be defined continuously. Let

$$\begin{aligned} \xi_\phi &: \mathbb{R}^{d_y} \to \mathbb{R}^{d_h}, \\ f_\phi &: [0, T] \times \mathbb{R}^{d_h} \to \mathbb{R}^{d_h}, \\ g_\phi &: [0, T] \times \mathbb{R}^{d_h} \to \mathbb{R}^{d_h \times d_y}, \\ m_\phi &\in \mathbb{R}^{d_h} \end{aligned} \quad (5)$$

where ξ_ϕ, f_ϕ, and g_ϕ are (Lipschitz) FFNNs. They are all parametrised by ϕ. The dimension h is a hyperparameter describing the size of the hidden state. This allows us to write the discriminator as an SDE of the form

$$\begin{aligned} H_0 &= \xi_\phi(Y_0), \\ dH_t &= f_\phi(t, H_t)dt + g_\phi(t, H_t)dY_t, \quad (6) \\ D &= m_\phi \cdot H_T, \end{aligned}$$

for $t \in [0, T]$, with $H : [0, T] \to \mathbb{R}^h$ being the (strong) solution to this SDE, which exists given Lipschitz continuity of f_ϕ, g_ϕ. The value $D \in \mathbb{R}$, a function of the terminal hidden state H_T, is the discriminator's score for real versus fake. The discriminator follows the formulation of a neural controlled differential equation (CDE) [46,52] concerning the control Y. A CDE allows the process H to naturally adapt to incoming data, as changes in Y change the local dynamics of the system.

2.2.3. Training Loss

In WGANs, the training loss is defined using the Wasserstein distance (also known as the Earth Mover's distance) between the distributions of real and generated samples,

aiming at measuring the difference between the average discriminator scores for real and generated samples as follows:

$$\mathbb{E}_{x\sim P_{\text{data}}}[D_\phi(x)] - \mathbb{E}_{z\sim P_{\text{noise}}}[D_\phi(G_\theta(z))],$$

where G_θ represents the overall action of the generator, and D_ϕ represents the overall action of the discriminator [53]. Such minimisation encourages the generator to produce samples that are indistinguishable from real samples according to the critic's evaluation. Training is performed via stochastic gradient descent techniques.

2.2.4. Lipschitz Regularisation

Since the Wasserstein distance estimate requires the discriminator's functional to be a 1-Lipschitz function, the Lipschitz constraint must be enforced. To achieve this, the Gradient Penalty technique is used [54], penalising the critic's gradient norm deviation from 1 at certain input points.

3. Experiments

This section details the experimental procedures carried out to develop and validate the wind energy production forecasting model. The entire workflow, from raw data to the forecast of new wind energy data, is outlined in the subsequent subsections and illustrated in the comprehensive flowchart of Figure 2.

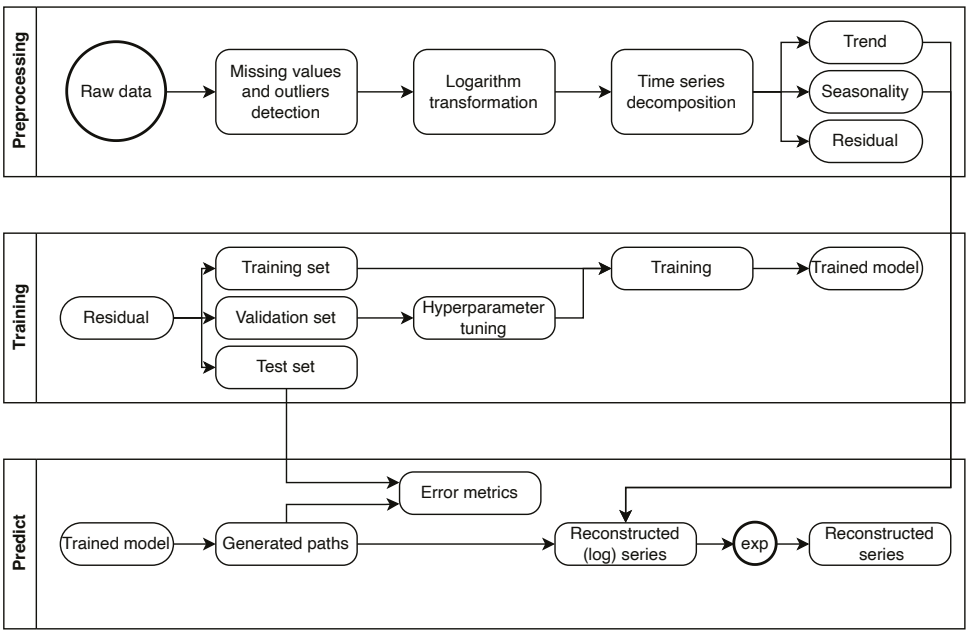


Figure 2. Flowchart representing the steps from raw data to the forecasting of wind energy.

3.1. Data Preprocessing

The dataset comprises a daily time series in which each observation represents the daily average wind energy production in the Italian electricity zone *SUD*. The dataset spans from 1 January 2015, to 31 December 2021, and it is shown in Figure 3.

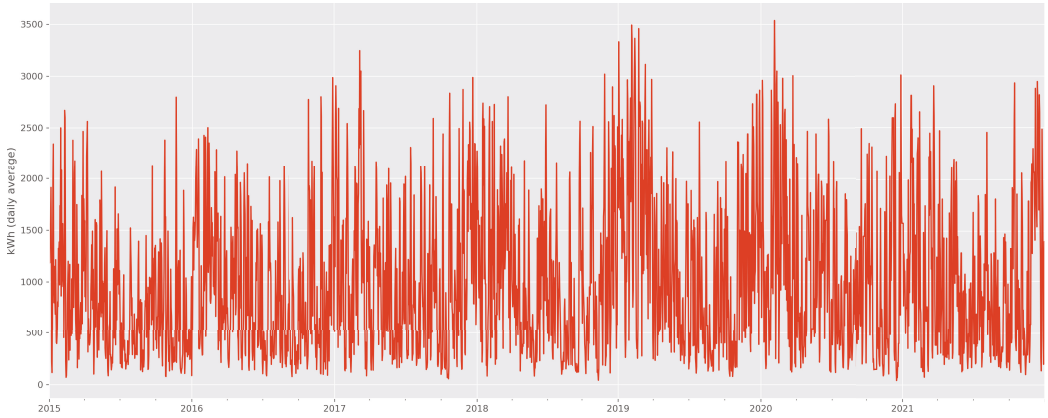


Figure 3. Time series of the daily average wind energy production in SUD zone.

The following data preprocessing procedures have been conducted:

- Missing data and potential outliers have been detected and corrected;
- The logarithmic transformations have been applied;
- The series has been decomposed into trends, seasonal and residual components. More specifically, the trend has been obtained via a rolling window smoothing of the series; the analysis of the periodogram allowed to detect the main seasonal component, i.e., annual periodicity, that was successively isolated with a Fourier decomposition; and the residual component has been tested for stationarity; both the Augmented Dickey–Fuller (ADF) test and the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test confirmed the stationary nature of the residual component.

The residual component was then used to construct the training and test sets for the GAN model. Specifically, each sample point within the data set represents a 7-day-long contiguous segment of the underlying time series. The data from 2015-01-01 to 2020-12-31 are used for training; from 2021-01-01 to 2021-09-30 for validation and hyperparameter tuning; and from 2021-10-01 to 2021-12-31 for testing.

3.2. Model Specification

3.2.1. Software

The PyTorch framework was adopted. The library `torchsde` [28,55] was used to solve the SDEs, and the library `torchcde` [52] for its interpolation schemes and to solve the neural CDEs used in the classification and prediction metrics.

3.2.2. Computing Infrastructure

Training was performed on GPU NVIDIA A100 80 GB.

3.2.3. Normalisation

All data were normalised to have zero mean and unit variance.

3.2.4. Architectures

Hyperparameters were selected according to hyperparameter tuning. The Python library `ray-tune` was used. Every neural network was parametrised as an FFNN with one hidden layer, width 16, and SiLU activations. Every model's drift, diffusion and vector fields also had a sigmoid nonlinearity as their final operation. The neural SDE's generator has a hidden state of size x, and the discriminator has a hidden state of size h. These were both taken as $x = h = 16$. The Latent ODE, likewise, has an evolving hidden state, which was also taken as size 16. The Latent ODE samples noise from a customarily distributed initial condition, which has been taken to have three dimensions. For the model with fBm, all the hidden sizes and noises were assumed to have dimension one.

3.2.5. Optimisers

The Latent ODE was trained with Adam [56] with a learning rate of 10^{-3}. The generator and discriminator of the neural SDE were trained with Adadelta, with a learning rate of 0.5×10^{-5}. The learning rates of the model with fBm were chosen to be equal to 10^{-1} and 0.5×10^{-5}, respectively.

3.2.6. Hurst Index Estimation

One of the pioneering methods for Hurst index estimation is the rescaled range analysis, commonly known as Range over Standard Deviation (R/S). In this section, we will describe the approach mentioned above, and estimate the Hurst index; see [57] for further details.

Generally speaking, this method involves calculating the rescaled range of a time series, which is the difference between the maximum and minimum cumulative sums of data divided by the standard deviation. On the one hand, we remark that the key advantage of R/S analysis lies in its simplicity, making it suitable for initial exploratory analyses; on the other, sensitivity to data size should be taken into account, and adjustments may be necessary for optimal results.

3.3. Algorithm to Estimate H

The approach relies on the representation of the Hurst exponent H in terms of the asymptotic behaviour of the rescaled range as a function of the time span of a time series. Formally:

$$\mathbb{E}\left[\frac{R(n)}{S(n)}\right] = C \cdot n^H \quad \text{as} \quad n \to \infty \tag{7}$$

where:

1. n represents the period of the observations, denoting the number of data points within a time series of length N under consideration;
2. $R(n)$ denotes the range of the initial n cumulative deviations from the mean;
3. $S(n)$ indicates the range of the series comprising the first n cumulative standard deviations;
4. C is a positive constant.

Therefore, taking into account the cumulative sums of residuals, the following procedure is implemented:

1. For each specified length $n \in \{n_1, \ldots, n_k = N\}$, the residuals $u = (u_i,\, i \in \{1, \ldots, n\})$ are extracted from the seasonal decomposition of the observed time series, and the mean $\overline{u}$ is subsequently computed;
2. The cumulative deviations are computed as $\widehat{R}(n) = \sum_{i=1}^{n}(u_i - \overline{u})$;
3. The range is computed as $\max(\widehat{R}(n)) - \min(\widehat{R}(n))$;
4. $S(n)$ is computed as the standard deviation of the cumulative deviations of $\widehat{R}(n)$;
5. The rescaled ranges are obtained as $\frac{R(n)}{S(n)}$;
6. A linear regression of the rescaled ranges is fitted against n on a log scale;
7. $\hat{H}$ is estimated as the slope of the regression analysis conducted over all the different k lengths considered.

3.4. Data Size Analysis

An R/S analysis was conducted on the residuals, with the optimal data size determined as a result of subsequent analysis:

- The rolling standard deviation is computed on a yearly basis;
- Both the minimum and maximum values of the rolling standard deviations are utilised to delineate a grid of five appropriate diffusion coefficients for the fractional Brownian motion. The range spans from the minimum value of 0.7423 to the maximum value of 0.8887, with equal spacing between each coefficient;

- A grid of nine appropriate values for H is also considered, and it spaces evenly from 0.1 to 0.9;
- For each combination of H and the diffusion coefficient, we simulate 1000 trajectories for the fBm, over 5 years of daily projections;
- The calibration algorithm for H is implemented over the trajectories generated using different data size, with RMSE computation realised for each data size selected;
- The minimum RMSE for H highlights the optimal data size.

The analysis focuses on a specific set of values chosen for their ability to provide deep insights into the model's dynamics while maintaining simplicity. Although this set may appear limited, each value was meticulously selected to cover a representative spectrum of scenarios and conditions. Expanding the dataset to include more values would not significantly enhance our understanding of the model's behaviour. Additional values would likely fall within ranges already covered by the selected set, contributing redundant information without introducing meaningful new insights. Furthermore, a denser dataset could obscure significant trends or patterns and overwhelm the reader with excessive data points.

Emphasising a concise yet comprehensive set of values allows for effective assessment of the model's performance across key metrics, facilitating clear conclusions without unnecessary complexity. This approach maximises the efficiency of our analysis, enabling extraction of valuable insights with minimal noise.

It is noteworthy that the preceding algorithm confirms the optimal window size as the maximum one.

In this specific case, we consider daily residuals across the Italian geographical area of interest, spanning from 1 January 2015, to 31 December 2020.

The algorithm described above produces an the value of H of 0.3522, as showed in Table 1.

Table 1. Estimation results for the Hurst exponent.

Zone	$\hat{H}$
SUD	0.3522

3.5. Results

In the simulation methodology, two primary error metrics have been used to assess the performance of the predictive model: Dynamic Time Warping (DTW) and Maximum–Minimum Discrepancy (MMD). These metrics offer distinct perspectives on the accuracy and fidelity of the wind energy production forecasts.

DTW is a technique commonly employed in time series analysis to measure the similarity between two sequences, accounting for variations in the time axis. By aligning the sequences non-linearly, DTW accommodates temporal distortions and fluctuations, making it particularly well-suited for evaluating the performance of time series forecasting models. In other words, DTW measures the degree of correspondence between the predicted and observed energy production trajectories, accounting for potential time shifts and variations.

MMD is a statistical measure used to quantify the dissimilarity between probability distributions. By comparing the maximum difference and minimum difference between the distributions of predicted and observed wind energy production, MMD captures discrepancies in both the mean and variance of the distributions. This metric offers a comprehensive assessment of the model's predictive accuracy, highlighting deviations in both central tendency and variability.

The implementation of DTW and MMD together provides complementary perspectives on the models' performance, encompassing both the temporal dynamics and the distributional characteristics of the generated data. Latter metrics allow us to effectively evaluate the quality, accuracy, and consistency of the wind energy production predictions

produced by the mentioned models, thus providing a comprehensive assessment of their comparative performance.

The experimental errors are shown in Table 2; they indicate comparable performance for the two models, both with and without the inclusion of fBm.

Table 2. Experimental results.

Metric	Neural SDE	Neural SDE with fBm
DTW	3.3078	3.0743
MMD	0.1042	0.1091

To better visualise the level of agreement, in Figures 4a and 5a, 50 samples from the true distribution are plotted against 50 from the learned distribution.

The level of agreement between the real and generated distributions is visually assessed by examining the overlap and spread of the trajectories. Both the real and generated trajectories exhibit a comparable spread as they evolve over time, suggesting that the model has effectively learned the variability and stochastic nature of the wind energy production, maintaining a consistent level of divergence comparable to the real data. In addition, the paths of the real and generated trajectories display similar patterns and fluctuations, implying that the model has accurately captured the underlying dynamics and trends of the wind energy production.

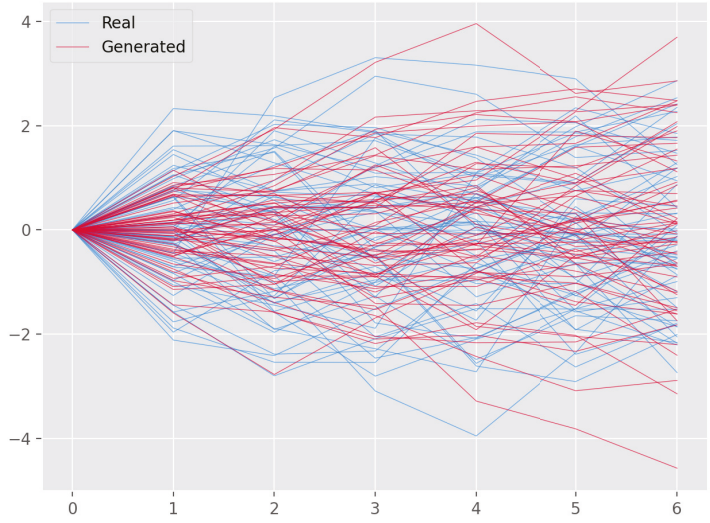
In addition to the trajectory comparison, the marginal distributions of the real and generated paths at specific time steps ($t = 1, 2, 3, 4, 5$) are presented in Figures 4b and 5b. These histograms provide a more granular view of the model's performance at each individual time point. The overlap between the real and generated distributions at each time step indicates that the model not only captures the general trend, but also the specific probabilistic characteristics of the data. This overlap reinforces the conclusion that the generated data are statistically similar to the real data. The close agreement between the real and generated distributions at these individual time points underscores the model's effectiveness in predicting wind energy production.

Since the model has been trained on the residual component of the wind energy production time series, the predictions generated by the model correspond to this residual part. To reconstruct the actual time series of wind energy production, it is necessary to reintroduce the trend and seasonal components that were isolated during the preprocessing steps. Figure 6 shows the recovered time series from the predicted residuals after adding the trend and seasonal components.

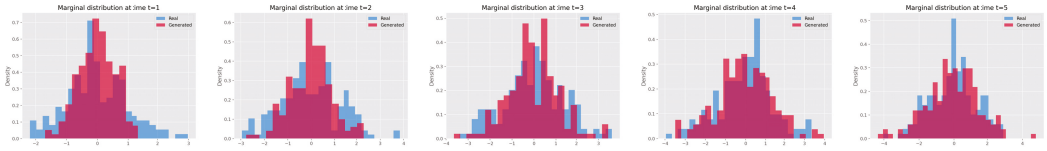
Then, the Mean Absolute Percentage Error (MAPE) has been calculated on the recovered time series (in logarithmic form) to assess the accuracy of the model's predictions. The resulting MAPE value is approximately 13–14%, as shown in Table 3. Although this error rate is not the lowest achievable, it is still within a satisfactory range, indicating that the model performs reasonably well in capturing and predicting the wind energy production dynamics.

Table 3. MAPE (%) computed on the reconstructed time series after adding trend and seasonality.

Metric	Neural SDE	Neural SDE with fBm
MAPE	13.40	14.29

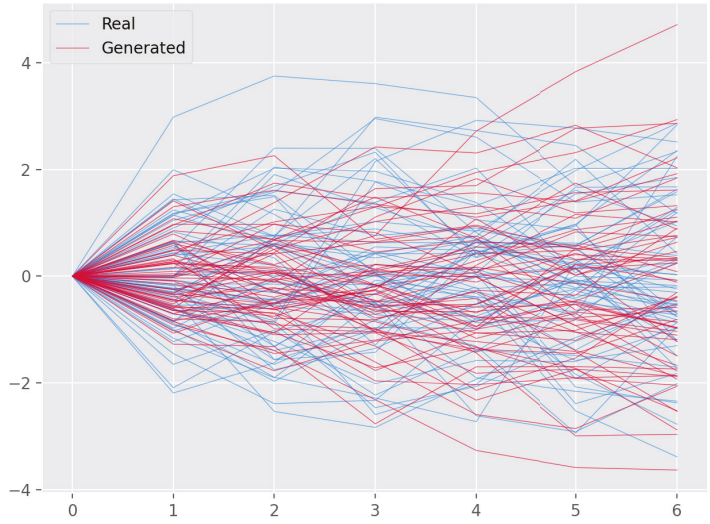


(**a**)

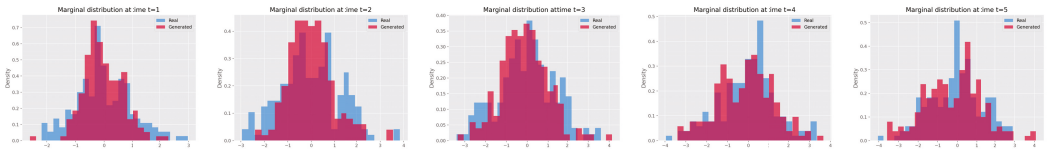


(**b**)

Figure 4. (**a**) Sample paths from the real and the generated processes. (**b**) Left to right: marginal distributions at $t = 1, 2, 3, 4, 5$.



(**a**)



(**b**)

Figure 5. (**a**) Sample paths from the real and the generated processes for the model with fBm. (**b**) Left to right: marginal distributions at $t = 1, 2, 3, 4, 5$ for the model with fBm.

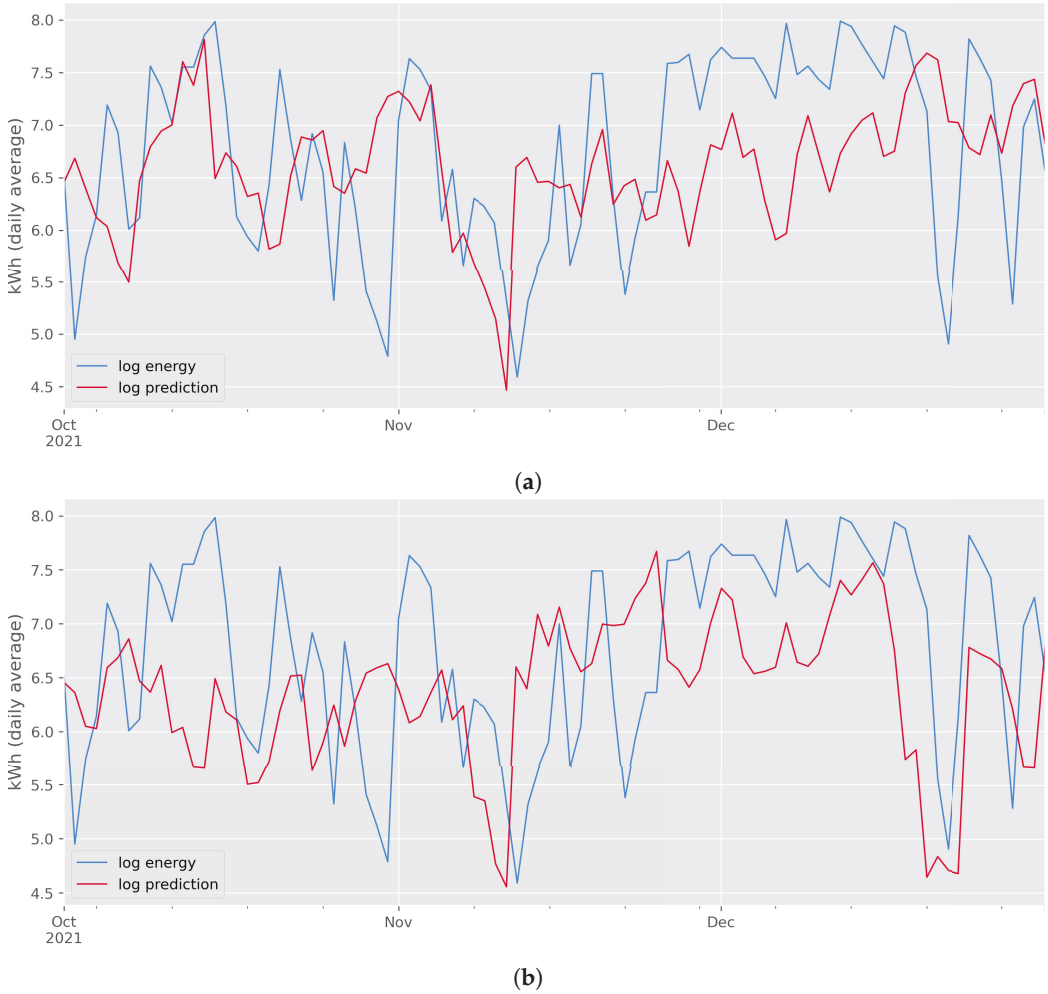


Figure 6. (**a**) Actual versus predicted wind energy production (in log scale). (**b**) Actual versus predicted wind energy production with fBm (in log scale).

4. Conclusions

In this study, predictive models for wind energy production in Italy were applied and evaluated, implementing GANs to capture the stochastic nature of wind behaviour. The investigation aimed to address the challenges associated with precise wind energy prediction, which is crucial for sustainable energy planning and management.

Through extensive numerical simulations and analyses, the effectiveness of the proposed approach in accurately forecasting wind energy production has been demonstrated. By treating the wind energy production time series as a realisation of a stochastic process, the complex dynamics and uncertainties inherent in wind generation have been successfully modelled.

Despite the introduction of fBm, comparable errors were observed between the two models.

This can be attributed to the linear transformation (4) that may mitigate the contribution of the fBm, resulting in similar error profiles between the models. The observed limited impact may be attributed to the consideration of fBm with dimensionality restricted to 1. Future research could explore the generalisation of the generation algorithm for fBm to encompass higher dimensions, as well as other models for characterising long-range dependency, such as the fractional Lévy stable motion (fLsm). With two parameters, α and H, the latter could characterise local irregularity and global persistence separately.

Author Contributions: Conceptualisation, L.D.P., N.F. and A.V.; methodology, L.D.P., N.F. and A.V.; software, N.F. and A.V.; validation, L.D.P., N.F. and A.V.; formal analysis, L.D.P., N.F. and A.V.; investigation, L.D.P., N.F. and A.V.; resources, A.V.; data curation, N.F. and A.V.; writing—original draft preparation, L.D.P., N.F. and A.V.; writing—review and editing, L.D.P., N.F. and A.V.; visualisation, L.D.P., N.F. and A.V.; supervision, L.D.P.; project administration, L.D.P., N.F. and A.V.; funding acquisition, N.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was realised with co-financing of the European Union—NOP Research and Innovation 2014–2020.

Data Availability Statement: Data sharing not applicable. No new data were created or analysed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ARMA	Autoregressive moving average
CGAN	Conditional GAN
CNN	Convolutional neural network
CSE	Controlled differential equation
DTW	Dynamic Time Warping
fBm	Fractional Brownian motion
FFNN	Feedforward neural network
fLsm	Fractional Lévy stable motion
GAN	Generative adversarial network
GAN-GP	GAN with gradient penalty
MAPE	Mean absolute percentage error
MMD	Maximum–Minimum Discrepancy
NN	Neural networks
SDE	Stochastic differential equation
SeqGAN	Sequence GAN
VAE	Variational autoencoder
WGAN	Wasserstein GAN

References

1. Zhang, X.; Ma, C.; Song, X.; Zhou, Y.; Chen, W. The impacts of wind technology advancement on future global energy. *Appl. Energy* **2016**, *184*, 1033–1037. [CrossRef]
2. Commission, E. Clean Energy. 2019. Available online: https://ec.europa.eu/commission/presscorner/detail/en/fs_19_6723 (accessed on 18 May 2024).
3. Weron, R. *Modeling and Forecasting Electricity Loads and Prices: A Statistical Approach*; John Wiley & Sons: Hoboken, NJ, USA, 2006; Volume 396.
4. Di Persio, L.; Fraccarolo, N. Energy consumption forecasts by gradient boosting regression trees. *Mathematics* **2023**, *11*, 1068. [CrossRef]
5. Di Persio, L.; Fraccarolo, N. Investment and Bidding Strategies for Optimal Transmission Management Dynamics: The Italian Case. *Energies* **2023**, *16*, 5950. [CrossRef]
6. Foley, A.M.; Leahy, P.G.; Marvuglia, A.; McKeogh, E.J. Current methods and advances in forecasting of wind power generation. *Renew. Energy* **2012**, *37*, 1–8. [CrossRef]
7. Hu, J.; Heng, J.; Wen, J.; Zhao, W. Deterministic and probabilistic wind speed forecasting with de-noising-reconstruction strategy and quantile regression based algorithm. *Renew. Energy* **2020**, *162*, 1208–1226. [CrossRef]
8. Wang, Y.; Li, Y.; Zou, R.; Foley, A.M.; Al Kez, D.; Song, D.; Hu, Q.; Srinivasan, D. Sparse heteroscedastic multiple spline regression models for wind turbine power curve modeling. *IEEE Trans. Sustain. Energy* **2020**, *12*, 191–201. [CrossRef]
9. Yunus, K.; Thiringer, T.; Chen, P. ARIMA-based frequency-decomposed modeling of wind speed time series. *IEEE Trans. Power Syst.* **2015**, *31*, 2546–2556. [CrossRef]
10. Kavasseri, R.G.; Seetharaman, K. Day-ahead wind speed forecasting using f-ARIMA models. *Renew. Energy* **2009**, *34*, 1388–1393. [CrossRef]
11. Maatallah, O.A.; Achuthan, A.; Janoyan, K.; Marzocca, P. Recursive wind speed forecasting based on Hammerstein Auto-Regressive model. *Appl. Energy* **2015**, *145*, 191–197. [CrossRef]

12. Wang, J.; Wang, S.; Yang, W. A novel non-linear combination system for short-term wind speed forecast. *Renew. Energy* **2019**, *143*, 1172–1192. [CrossRef]
13. Wang, Y.; Wang, J.; Wei, X. A hybrid wind speed forecasting model based on phase space reconstruction theory and Markov model: A case study of wind farms in northwest China. *Energy* **2015**, *91*, 556–572. [CrossRef]
14. Liu, H.; Mi, X.; Li, Y. Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, LSTM network and ELM. *Energy Convers. Manag.* **2018**, *159*, 54–64. [CrossRef]
15. Song, J.; Wang, J.; Lu, H. A novel combined model based on advanced optimization algorithm for short-term wind speed forecasting. *Appl. Energy* **2018**, *215*, 643–658. [CrossRef]
16. He, X.; Nie, Y.; Guo, H.; Wang, J. Research on a Novel Combination System on the Basis of Deep Learning and Swarm Intelligence Optimization Algorithm for Wind Speed Forecasting. *IEEE Access* **2020**, *8*, 51482–51499. [CrossRef]
17. Mezaache, H.; Bouzgou, H. Auto-Encoder with Neural Networks for Wind Speed Forecasting. In Proceedings of the 2018 International Conference on Communications and Electrical Engineering (ICCEE), El Oued, Algeria, 17–18 December 2018; pp. 1–5. [CrossRef]
18. Fu, W.; Wang, K.; Tan, J.; Zhang, K. A composite framework coupling multiple feature selection, compound prediction models and novel hybrid swarm optimizer-based synchronization optimization strategy for multi-step ahead short-term wind speed forecasting. *Energy Convers. Manag.* **2020**, *205*, 112461. [CrossRef]
19. Niu, Z.; Yu, Z.; Tang, W.; Wu, Q.; Reformat, M. Wind power forecasting using attention-based gated recurrent unit network. *Energy* **2020**, *196*, 117081. [CrossRef]
20. Yu, Y.; Han, X.; Yang, M.; Yang, J. Probabilistic Prediction of Regional Wind Power Based on Spatiotemporal Quantile Regression. In Proceedings of the 2019 IEEE Industry Applications Society Annual Meeting, Baltimore, MD, USA, 29 September–3 October 2019; pp. 1–16. [CrossRef]
21. Kosovic, B.; Haupt, S.E.; Adriaansen, D.; Alessandrini, S.; Wiener, G.; Delle Monache, L.; Liu, Y.; Linden, S.; Jensen, T.; Cheng, W.; et al. A comprehensive wind power forecasting system integrating artificial intelligence and numerical weather prediction. *Energies* **2020**, *13*, 1372. [CrossRef]
22. Liu, T.; Huang, Z.; Tian, L.; Zhu, Y.; Wang, H.; Feng, S. Enhancing wind turbine power forecast via convolutional neural network. *Electronics* **2021**, *10*, 261. [CrossRef]
23. Shi, J.; Guo, J.; Zheng, S. Evaluation of hybrid forecasting approaches for wind speed and power generation time series. *Renew. Sustain. Energy Rev.* **2012**, *16*, 3471–3480. [CrossRef]
24. Chen, C.; Liu, H. Dynamic ensemble wind speed prediction model based on hybrid deep reinforcement learning. *Adv. Eng. Informatics* **2021**, *48*, 101290. [CrossRef]
25. Chen, Y.; Zhang, S.; Zhang, W.; Peng, J.; Cai, Y. Multifactor spatio-temporal correlation model based on a combination of convolutional neural network and long short-term memory neural network for wind speed forecasting. *Energy Convers. Manag.* **2019**, *185*, 783–799. [CrossRef]
26. Chen, J.; Zeng, G.Q.; Zhou, W.; Du, W.; Lu, K.D. Wind speed forecasting using nonlinear-learning ensemble of deep learning time series prediction and extremal optimization. *Energy Convers. Manag.* **2018**, *165*, 681–695. [CrossRef]
27. Wang, G.; Jia, R.; Liu, J.; Zhang, H. A hybrid wind power forecasting approach based on Bayesian model averaging and ensemble learning. *Renew. Energy* **2020**, *145*, 2426–2434. [CrossRef]
28. Kidger, P.; Foster, J.; Li, X.; Lyons, T.J. Neural SDEs as Infinite-Dimensional GANs. In Proceedings of the Proceedings of the 38th International Conference on Machine Learning, Virtual, 18–24 July 2021; Volume 139.
29. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 2672–2680.
30. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2242–2251. [CrossRef]
31. Yuan, R.; Wang, B.; Mao, Z.; Watada, J. Multi-objective wind power scenario forecasting based on PG-GAN. *Energy* **2021**, *226*, 120379. [CrossRef]
32. Chen, Y.; Wang, Y.; Kirschen, D.; Zhang, B. Model-Free Renewable Scenario Generation Using Generative Adversarial Networks. *IEEE Trans. Power Syst.* **2018**, *33*, 3265–3275. [CrossRef]
33. Chen, Y.; Li, P.; Zhang, B. Bayesian renewables scenario generation via deep generative networks. In Proceedings of the 2018 52nd Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 13–15 March 2018; pp. 1–6. [CrossRef]
34. Jiang, C.; Mao, Y.; Chai, Y.; Yu, M.; Tao, S. Scenario Generation for Wind Power Using Improved Generative Adversarial Networks. *IEEE Access* **2018**, *6*, 62193–62203. [CrossRef]
35. Wei, H.; Hongxuan, Z.; Yu, D.; Yiting, W.; Ling, D.; Ming, X. Short-term optimal operation of hydro-wind-solar hybrid system with improved generative adversarial networks. *Appl. Energy* **2019**, *250*, 389–403. [CrossRef]
36. Chen, Y.; Wang, X.; Zhang, B. An Unsupervised Deep Learning Approach for Scenario Forecasts. In Proceedings of the 2018 Power Systems Computation Conference (PSCC), Dublin, Ireland, 11–15 June 2018; pp. 1–7. [CrossRef]
37. Wang, F.; Zhang, Z.; Liu, C.; Yu, Y.; Pang, S.; Duić, N.; Shafie-khah, M.; Catalão, J.P. Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting. *Energy Convers. Manag.* **2019**, *181*, 443–462. [CrossRef]

38. Zhang, Y.; Ai, Q.; Xiao, F.; Hao, R.; Lu, T. Typical wind power scenario generation for multiple wind farms using conditional improved Wasserstein generative adversarial network. *Int. J. Electr. Power Energy Syst.* **2020**, *114*, 105388. [CrossRef]
39. Liang, J.; Tang, W. Sequence Generative Adversarial Networks for Wind Power Scenario Generation. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 110–118. [CrossRef]
40. Sun, Z.; El-Laham, Y.; Vyetrenko, S. Neural Stochastic Differential Equations with Change Points: A Generative Adversarial Approach. In Proceedings of the ICASSP 2024–2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Republic of Korea, 14–19 April 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 6965–6969.
41. Allouche, M.; Girard, S.; Gobet, E. A generative model for fBm with deep ReLU neural networks. *J. Complex.* **2022**, *73*, 101667. [CrossRef]
42. Oksendal, B. *Stochastic Differential Equations: An Introduction with Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
43. Rogers, L.C.G.; Williams, D. *Diffusions, Markov Processes and Martingales: Volume 2, Itô Calculus*; Cambridge University Press: Cambridge, UK, 2000; Volume 2.
44. Chen, R.T.; Rubanova, Y.; Bettencourt, J.; Duvenaud, D.K. Neural ordinary differential equations. *Adv. Neural Inf. Process. Syst.* **2018**, *31*.
45. Kidger, P.; Morrill, J.; Foster, J.; Lyons, T. Neural Controlled Differential Equations for Irregular Time Series. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–12 December 2020; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H., Eds.; Curran Associates, Inc.: Glasgow, UK, 2020; Volume 33, pp. 6696–6707.
46. Kidger, P. On neural differential equations. *arXiv* **2022**, arXiv:2202.02435.
47. Yang, L.; Gao, T.; Lu, Y.; Duan, J.; Liu, T. Neural network stochastic differential equation models with applications to financial data forecasting. *Appl. Math. Model.* **2022**, *115*, 279–299. [CrossRef]
48. Kong, L.; Sun, J.; Zhang, C. Sde-net: Equipping deep neural networks with uncertainty estimates. *arXiv* **2020**, arXiv:2008.10546.
49. Gierjatowicz, P.; Sabaté-Vidales, M.; Šiška, D.; Szpruch, Ł.; Zuric, Z. Robust pricing and hedging via neural stochastic differential equations. *J. Comput. Financ.* **2022**, *16*. [CrossRef]
50. Veeravalli, T.; Raginsky, M. Nonlinear controllability and function representation by neural stochastic differential equations. *arXiv* **2022**, arXiv:2212.00896.
51. Yang, L.; Zhang, D.; Karniadakis, G. Physics-Informed Generative Adversarial Networks for Stochastic Differential Equations. *SIAM J. Sci. Comput.* **2018**, *42*, A292–A317. [CrossRef]
52. Kidger, P.; Morrill, J.; Foster, J.; Lyons, T. Neural controlled differential equations for irregular time series. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 6696–6707.
53. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 214–223.
54. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A.C. Improved training of wasserstein gans. *Adv. Neural Inf. Process. Syst.* **2017**, *30*.
55. Li, X.; Wong, T.K.L.; Chen, R.T.Q.; Duvenaud, D. Scalable gradients for stochastic differential equations. In Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (AISTATS), Online, 26–28 August 2020.
56. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2017**, arXiv:1412.6980
57. Qian, B.; Rasheed, K. Hurst exponent and financial market predictability. In Proceedings of the IASTED Conference on Financial Engineering and Applications, Cambridge, MA, USA, 8–10 November 2007; pp. 203–209.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Forecasting and Multilevel Early Warning of Wind Speed Using an Adaptive Kernel Estimator and Optimized Gated Recurrent Units

Pengjiao Wang [1], Qiuliang Long [1,2], Hu Zhang [2], Xu Chen [2], Ran Yu [2] and Fengqi Guo [1,*]

[1] School of Civil Engineering, Central South University, Changsha 410075, China; w2592570@gmail.com (P.W.); longqiuliang@csu.edu.cn (Q.L.)
[2] Hunan Harbor Engineering Corporation Limited, Changsha 410021, China; zhanghu07@126.com (H.Z.); chenxu13@126.com (X.C.); yuran0812@163.com (R.Y.)
* Correspondence: fengqiguo@csu.edu.cn

Abstract: Accurately predicting wind speeds is of great significance in various engineering applications, such as the operation of high-speed trains. Machine learning models are effective in this field. However, existing studies generally provide deterministic predictions and utilize decomposition techniques in advance to enhance predictive performance, which may encounter data leakage and fail to capture the stochastic nature of wind data. This work proposes an advanced framework for the prediction and early warning of wind speeds by combining the optimized gated recurrent unit (GRU) and adaptive kernel density estimator (AKDE). Firstly, 12 samples (26,280 points each) were collected from an extensive open database. Three representative metaheuristic algorithms were then employed to optimize the parameters of diverse models, including extreme learning machines, a transformer model, and recurrent networks. The results yielded an optimal selection using the GRU and the crested porcupine optimizer. Afterwards, by using the AKDE, the joint probability density and cumulative distribution function of wind predictions and related predicting errors could be obtained. It was then applicable to calculate the conditional probability that actual wind speed exceeds the critical value, thereby providing probabilistic-based predictions in a multilevel manner. A comparison of the predictive performance of various methods and accuracy of subsequent decisions validated the proposed framework.

Keywords: wind speed forecasting; gated recurrent unit; metaheuristic optimization; machine learning; kernel density estimation; cumulative distribution function

MSC: 68T20; 60G35; 62M45; 68T05

1. Introduction

In recent years, the coverage of high-speed railway (HSR) networks has been expanding at an accelerated rate; however, at the same time, the greater complexity of the operating environment poses increasing challenges. One challenge is that high-speed railway trains often face strong winds. Excessive wind speed can significantly affect the dynamic performance of high-speed trains, such as the overturning coefficient [1] and car body rolling motion [2], thereby endangering the operational safety of trains. At present, the commonly adopted method worldwide involves monitoring the windy environment along the HSR in real time, setting a wind speed threshold and establishing an early warning system [3–5], which relies on an accurate and timely prediction model for the wind speed. However, the capability of the existing models still needs to be improved, and the delay in the delivery of warnings also needs to be considered. For example, in China, although a relatively perfect monitoring and warning system has been established for strong winds along HSR lines, there is still a delay of 2–3 min in the transfer of warning information [6]. Hence, it is highly

important to establish a very short-term (i.e., a few seconds to 30 min ahead [7]) prediction model for wind speed to ensure safety and improve the operational efficiency of HSRs.

The prediction of short-term wind speed has become a popular topic in the fields of traffic safety and disaster prevention. Generally, the widely used wind speed prediction models can be classified into (i) physical models; (ii) statistical models, including traditional statistical methods and artificial intelligence (AI)-based models; and (iii) hybrid models. Among them, physical models usually consider various meteorological features, including barometric pressure, temperature, and humidity, for wind speed prediction; there are models such as numerical weather prediction and weather researcher forecasting, but they are more suitable for medium- and long-term forecasts [8].

In contrast, statistical models represented by autoregressive integrated moving average (ARIMA) and its derivatives have shown good performance in short-term wind speed (and wind energy) prediction [9,10]. A recent investigation into the Hammerstein autoregressive [11] model suggested its superior ability over ARIMA in capturing diverse wind speed characteristics, including asymmetric wind speed distributions, nonstationary time series profiles, and chaotic dynamics. It even outperforms artificial neural network models in terms of mathematical metrics. It should be noted that new AI-based statistical models have received the most attention at present. As they are good at capturing the implicit features in complex nonlinear problems, many excellent algorithms in machine learning [12–14] and deep learning [15,16], such as the long short-term memory (LSTM) model and deep belief network, have been widely used in wind speed prediction and have achieved good results. Nevertheless, concurrent studies [17] have highlighted that predicting wind speed sequences without noise reduction or other preprocessing steps may lead to significant errors, particularly in ultra-short-term predictions. This is attributed to the pronounced nonstationary stochastic nature of wind field data, which also demonstrates the importance of employing hybrid models.

The basic idea of hybrid models is to combine a prediction model (physical or statistical) with techniques that are able to improve model performance, mainly signal processing techniques and intelligent optimization algorithms. Signal processing techniques in wind speed prediction include wavelet decomposition [18], empirical modal decomposition [19], and variational mode decomposition [20] and their derivatives, which can efficiently enhance the signal-to-noise ratio of wind signals and reduce the influence of nonstationary features on the prediction process. Relevant studies have shown [12] that the use of front-loaded signal processing techniques can reduce the prediction error of a model by approximately 30–50% on a given dataset. The use of intelligent optimization algorithms, on the other hand, is to search for the optimal parameter configurations to increase the robustness of the prediction models. For example, studies have used genetic algorithms, cuckoo searches, conjugate gradient algorithms, and improved atomic search algorithms to determine the optimal parameters for prediction models [21,22]. Notably, due to variations in datasets, different studies often yield various optimal algorithms and corresponding parameter settings. However, overall, the use of hybrid models has now reached a consensus.

Table 1 summarizes the methodologies in related works on wind speed forecasting and subsequent warning systems. As it shows, for the prediction part, current models can obtain fine accuracy by using decomposition techniques in advance. This is generally because the decomposed components are smoother when compared with the original signal, and more importantly, a decomposition process of the overall signal may leak future information (i.e., the test set) that should have been unknown. Moreover, solutions [23] to this issue (step-by-step decomposition) will inevitably increase the training cost. Given this, using optimization algorithms may be more intuitive and effective. It is also worth noting that existing methods are dominated by deterministic point prediction; hence, they cannot reflect the effects of strong randomness and intermittency of the monitored data or the uncertainty of model parameters on the predicted results, making it difficult to provide direct warnings and assist decision making. In addition, while there has been

much progress in modelling medium- and long-term wind speed predictions, the relevant advances in (ultra) short-term predictions remain relatively slow.

Table 1. Related works on wind speed predictions and warning systems.

Literature	Predicting Model				Warning System	
	Base Model	Decomposition Techniques	Optimization Algorithm	Data Interval	Method	Level
Ref. [24]	ANN	—	—	10 min	Deterministic	Multiple
Ref. [19]	ARIMA	EMD	—	1 min	Deterministic	Multiple
Ref. [25]	ANN	EEMD	GA	10 min	Deterministic	Single
Ref. [26]	ELM/ARIMA	ICEEMDAN	—	10 min	Deterministic	Single
Ref. [27]	DBM	EEMD	—	1 h	—	—
Ref. [28]	RNN	—	HNN	15 min	Probabilistic	Single
Ref. [29]	ELM	—	AdaBoost	10/20/30 min	—	—
Ref. [30]	RNN/ELM	EEMD	GA	5min	—	—
Ref. [31]	LSTM/SVM	CEEMDAN	PSO/IWOA	5 min	—	—
Ref. [22]	SVM	WD	PSO/IASO	1 h		
Ref. [32]	GRU	VMD	PSR/IWOA	10 min		
Ref. [6]	LSTM	—	—	1/40 s	Probabilistic	Multiple

In view of this situation, it is proposed to use hybrid models achieved by parametric optimization for wind speed prediction and employ probabilistic prediction methods [33] based on conditional probability. Based on the results of existing studies, the use of hybrid models can achieve better performance than physical and statistical models, and the implementation of optimization algorithms can avoid data leakage. More importantly, with extensive historic data and predicting results, the predictive errors can be used to enhance final warning decision, which is a totally different perspective compared with using various features in model training. Specifically, the proposed framework consists of two main phases: the use of an optimized network to obtain the predicted wind speed and the use of a kernel estimator to provide probabilistic results for subsequent multilevel warnings. To achieve this goal, we first predict the wind speed via an optimized recurrent network and then form the joint kernel density of the wind speed predictions and the prediction errors. Afterwards, the proposed framework can not only output predictions at specific timestamps but also estimate the conditional probability that the predictions fall within the speed limit intervals. In this scenario, a warning system is proposed in which the predicted values and their conditional probabilities are considered. Table 2 summarizes the abbreviations used in this work.

Table 2. Abbreviations used in this work.

Term	Description	Term	Description
HSR	high-speed railway	ORELM	online recurrent extreme learning machine
CEC	congress on evolutionary computation	MAE	mean absolute error
CPO	crested porcupine optimizer	MAPE	mean absolute percentage error
GWO	gray wolf optimizer	RMSE	root mean square error
WOA	whale optimization algorithm	R^2	coefficient of determination
ARIMA	autoregressive integrated moving average	KDE	kernel density estimation
AI	artificial intelligence	AKDE	adaptive kernel density estimation
RNN	recurrent neural network	CDF	cumulative distribution function
LSTM	long short-term memory	MISE	mean integrated square error
BiLSTM	bidirectional LSTM	AMISE	asymptotic mean integrated square error
GRU	long short-term memory	T/F-P/N	true/false positive/negative
ELM	extreme learning machine	T/F-PR	true/false positive rate

2. Materials and Methods

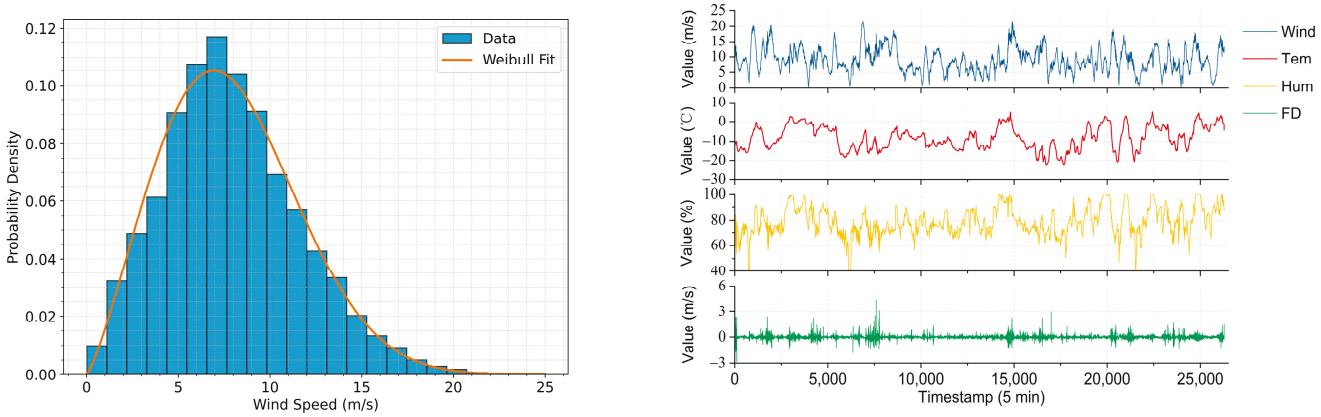
2.1. Data Collection

The quality of the dataset is essential for ensuring the validity and predictive performance of machine learning models. To provide a solid basis for this work, we collected an informative dataset of wind speed from a comprehensive worldwide wind database

named WRDB [34]. Data at 45.00° N, 82.00° W, for the years 2014, 2011, and 2008 were collected and denoted as D1, D2, and D3, respectively. Each dataset contains 105,120 data points of wind speed in one year with a sampling interval of 5 min. Inspection of these datasets revealed no unusual local outliers or missing values. In addition, each dataset is divided into four seasons, with each containing approximately 90–92 days, thereby forming 12 samples for model training and testing, denoted as D1/2/3-1/2/3/4 (26,280 points each). The database contains the wind speed data and related environmental features. In this work, the temperature and humidity are chosen as the input features. As an example, the distributions of Dataset 1 are drawn in Figure 1a, along with a curve fitted by the Weibull function, one of the most commonly used functions for practical wind data; see Equation (1).

$$f(x;\lambda,k) = \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-(\frac{x}{\lambda})^k} (x \geq 0) \quad (1)$$

where $f(x;\lambda,k)$ is the probability density function of the Weibull distribution and where k and λ are also known as the shape and scale parameters, respectively.



(**a**) (**b**)

Figure 1. Distribution of the wind speed and environmental data. (**a**) Histogram and Weibull fitting ($k = 2.27$, $\lambda = 8.94$) of the wind speed; (**b**) distribution of the features and target. Note that Wind, Tem, Hum, and FD refer to the wind speed, temperature, humidity, and first difference in the wind speed, respectively.

As shown in Figure 1, the distribution shape of the wind data is consistent with the probability density function of the Weibull function, which is also proven by a relatively low (approximately 0.02) Kolmogorov–Smirnov test statistic. The fitted shape parameter k is 2.27, which is close to the empirical value of 2 in practical research.

2.2. Methodologies

2.2.1. Crested Porcupine Optimizer

Stochastic optimization algorithms are widely used in machine learning to improve the robustness of models. Recently, meta-inspired optimization algorithms have gained the attention of researchers and are widely used to address challenging optimization problems. Most of these algorithms are based on the principles of biology, physics, behavioral science, or group intelligence and aim to simulate various characteristics of natural systems [35]. The crested porcupine optimizer (CPO) [36] is a new representative of such algorithms. It simulates the defense strategies of crowned porcupines using four strategies, namely, visual, sound, odor, and physical attacks, which are ranked by aggressiveness. These strategies map to four defense regions in the search space, which are activated sequentially on the basis of the threat level posed by the predator. Statistical analysis of the results shows that the CPO can achieve better performance in Congress on Evolutionary Computation (known as CEC) benchmarks [37] than its competitors, such as the well-known gray wolf optimizer

(GWO) [38], whale optimization algorithm (WOA) [39], and salp swarm algorithm [40], with improvement rates of up to 83% for CEC2014 and 100% for six real-world engineering problems. In this work, CPO is implemented to optimize general prediction models, and the key components are summarized as follows:

1. Population and fitness initialization:

The purpose of this step is to generate an initial population of candidate solutions within the specified bounds and evaluate the fitness of each solution by using an objective function for practical problems. Typically, Equations (2) and (3) are employed.

$$X_i^j = lb_j + \tau_0 \cdot \left(ub_j - lb_j\right) \tag{2}$$

$$f_i = f(X_i) \tag{3}$$

where X_i^j is the $j-th$ dimension of the $i-th$ candidate solution, τ_0 is a random value in [0, 1], and lb_j and ub_j are the lower and upper bounds of the $j-th$ dimension, respectively. In addition, f_i is the fitness of X_j.

2. Four defensive strategies in CPO

- Exploration phase (strategies 1 and 2)

When the porcupine becomes aware of the predator, it starts flapping its spines to expand its size. The predator has two options: move towards or move away from the porcupine (the distance between them decreases or increases). Equations (4) and (5) are thereby presented on the basis of the average position of randomly selected solutions. For the second strategy, the porcupine's voice becomes louder as the predator approaches it. Gaussian perturbations are used in Equation (6) to update positions to simulate a sound attack.

$$x_i^{j+1} = x_i^j + \tau_1 \cdot \left| 2\tau_2 \cdot x_{CP}^j - y_i^j \right| \tag{4}$$

$$y_i^j = \left(x_i^j + x_r^j\right)/2 \tag{5}$$

$$x_i^{j+1} = (1 - U_1) \cdot x_i^j + U_1 \cdot \left(y + \tau_3 \cdot \left(x_{r1}^j - x_{r2}^j\right)\right) \tag{6}$$

where x_{CP}^j is the global best solution vector; τ_1 is a Gaussian random variable; τ_2 is a random value in $[0, 1]$; y_i^j is a vector between the current porcupine and a porcupine selected randomly from the population whose solution is x_r^j; and U_1 is a binary vector, where x_{r1}^j and x_{r2}^j are other randomly selected solutions.

- Exploitation phase (strategies 3 and 4)

In this phase, the porcupine first secretes a fetid odor that spreads in the region around it to prevent the predator from approaching it. To achieve this, fitness-based scaling is used for updating the positions; see Equations (7)–(10). For the final strategy, the porcupine resorts to physical attack when a predator is very close and strikes it with short, thick quills. During a physical attack, the two bodies are strongly fused, representing an inelastic collision in one dimension; see Equations (11) and (12).

$$x_i^{j+1} = (1 - U_1) \cdot x_i^j + U_1 \cdot \left(x_{r1}^j + S_i^j \cdot \left(x_{r2}^j - x_{r3}^j\right) - \tau_4 \cdot \delta \cdot \gamma_t \cdot S_i^j\right) \tag{7}$$

$$\delta = \begin{cases} 1 & if\ rand \leq 0.5 \\ -1 & Else \end{cases} \tag{8}$$

$$\gamma_t = 2 \cdot rand \cdot \left(1 - \frac{j}{j_{max}}\right)^{\frac{j}{j_{max}}} \tag{9}$$

$$S_i^j = exp\left(\frac{f\left(x_i^j\right)}{\sum_{k=1}^{N} f\left(x_k^j\right) + \epsilon}\right) \tag{10}$$

$$x_i^{j+1} = x_{CP}^j + (\alpha(1-\tau_5) + \tau_5)\cdot\left(\delta\cdot x_{CP}^j - x_i^j\right) - \tau_6\cdot\delta\cdot\gamma_t\cdot F_i^j \tag{11}$$

$$F_i^j = \tau_7\cdot\frac{S_i\cdot\left(x_i^{j+1} - x_r^j\right)}{\Delta j} \tag{12}$$

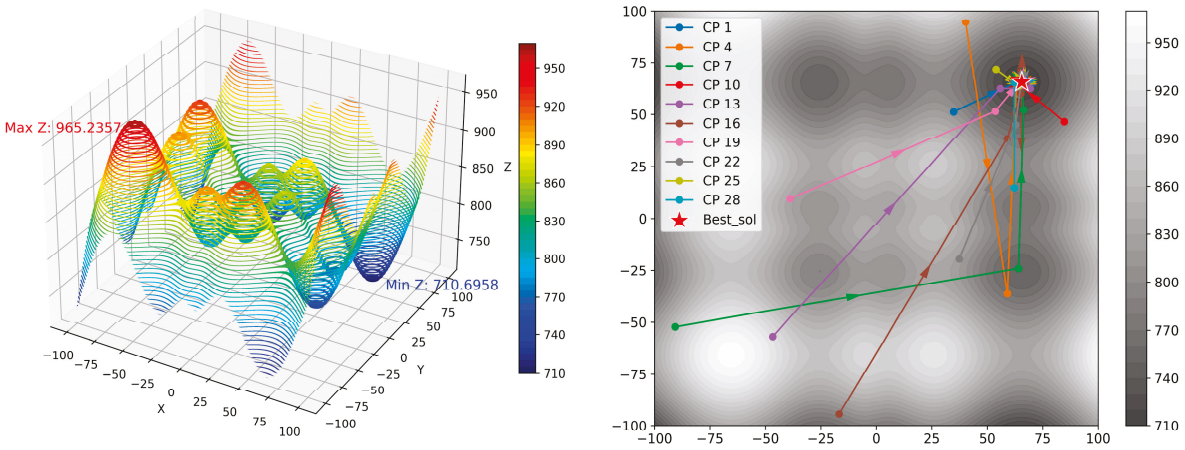
where x_{r3}^j is another randomly selected solution; δ is a parameter used to control the search direction; γ_t is a factor; S_i^j is the odor diffusion factor; $f\left(x_i^j\right)$ represents the objective function value of the $i-th$ individual; ϵ is a small value to avoid division by zero; $\tau_{4,5,6,7}$ are random values within [0, 1]; and α is a factor related to convergence speed.

3. Simple case using the CPO

Below is the presentation of a straightforward optimization task aimed at finding the minimum value of a function given by Equation (13). Given the bounds of $(-100, 100)$ and a dimension of 2, the minimum value is 710.6958 at (64.55, 64.55), as shown in Figure 2a. Figure 2b shows the search trajectories of 10 porcupines under 100 iterations when CPO is used in this case.

$$f(x) = 418.9829\cdot d - \sum_{i=1}^{d} x_i sin\left(\sqrt{|x_i|}\right) \tag{13}$$

where d is a natural number representing the dimension of the function.



(**a**) (**b**)

Figure 2. Results of seeking the minimum value via the CPO. (**a**) Shape of the original function; (**b**) searching trajectories of the porcupines (10 out of 30).

2.2.2. Gated Recurrent Unit

Wind speed prediction is a very specific but meaningful task in time series forecasting in engineering applications. Generally, time series data differ from other types of data because of their temporal nature and ordered sequence of observations. Recurrent neural networks (RNNs) can capture temporal dependencies in data by maintaining an internal state or memory, enabling them to process time-dependent sequences. Frameworks, including gated recurrent units (GRUs) and long short-term memory (LSTM) networks, were subsequently developed to address the problem of vanishing gradients when RNNs are used. Research has indicated that the LSTM and GRU architectures achieve high accuracy in time series prediction. However, the GRU exhibits comparable or even better performance than LSTM in many tasks while requiring fewer parameters, thereby consuming fewer computational resources [41–43]. In this work, a GRU is employed to predict wind speed

series. The general equations of a single-layer GRU cell are shown in Equations (14)–(17) and Figure 3 for illustration.

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{14}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{15}$$

$$\widetilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{16}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h}_t \tag{17}$$

where x_t is the input at time step t, h_{t-1} is the hidden state from step $t-1$, and σ denotes the sigmoid activation function, where z_t and r_t are the update and reset gates at step t, respectively. In addition, $\widetilde{h}_t$ and h_t are the candidate hidden state and hidden state at step t, respectively; tanh denotes the hyperbolic tangent activation function; $\odot$ is the element–wire operator; $W_{z,\ r,\ h}$ and $U_{z,r,h}$ are weight matrices; and $b_{z,\ r,\ h}$ are bias vectors.

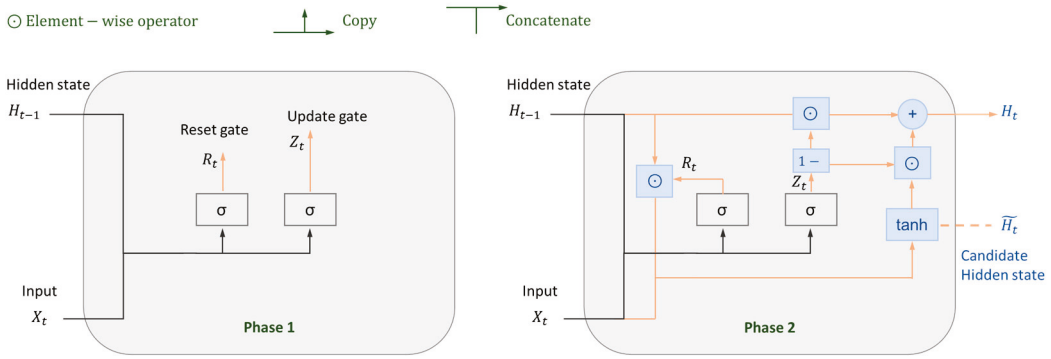


Figure 3. Architecture of a single-layer GRU cell.

2.2.3. Interval Forecasts via Kernel Density Estimation

Extensive studies have shown the effectiveness and accuracy of neural networks in time series forecasting tasks. However, these studies generally present a deterministic point prediction for these tasks and hence cannot capture the uncertainties in measured wind speed data as well as the difference between the predicted and true values. To address this, we propose employing kernel density estimation (KDE) to obtain interval values of prediction. It contains two main steps: (i) estimate the kernel density and (ii) generate confidence intervals via bootstrap resampling.

The main challenge of using this technique is to find an optimized bandwidth for the KDE. The standard univariate KDE of a discrete sequence $D = \{x_1,\ x_2,\ \ldots, x_n\}$ from an unknown function $f(x)$ is shown in Equation (18). Notably, the asymptotic convergence of $\hat{f}(x|D,h)$ toward the underlying $f(x)$ is significantly affected by the choice of bandwidth (h) rather than the kernel function (K).

$$\hat{f}(x|D,h) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{18}$$

where K is a kernel function satisfying $\int K(x)dx = 1$ and h is the bandwidth.

A plausible way to choose the bandwidth is to minimize the mean integrated square error (MISE); see Equation (19). Under an integrability assumption on $f(x)$, we can further define the asymptotic mean integrated square error (AMISE), as shown in Equation (20). Afterwards, the value of the bandwidth that minimizes the AMISE is given by Equation (21). The computationally simplest method for choosing a global bandwidth is based on replacing the unknown $R(f'')$ in Equation (21) with its value for a parametric family [44]. For example, Scott's rule [45] is widely adopted, as shown in Equation (22). Additionally, a natural extension of the standard KDE uses adaptive KDE (AKDE), which is obtained when

h is no longer a global constant. Considering the effectiveness and efficiency of various algorithms, an AKDE method [46] employing a localized MISE and a scaled parameter is used in this work. A simple case below presents the results of KDE using various rules for comparison, as shown in Figure 4a.

$$\begin{aligned} \mathrm{MISE}\left(\hat{f}_h\right) &= E\left\{\int\left(\hat{f}_h(y) - f(y)\right)^2 dy\right\} \\ &= \int \mathrm{Bias}\left(\hat{f}_h(y)\right)^2 dy + \int \mathrm{Var}\left(\hat{f}_h(y)\right) dy \end{aligned} \tag{19}$$

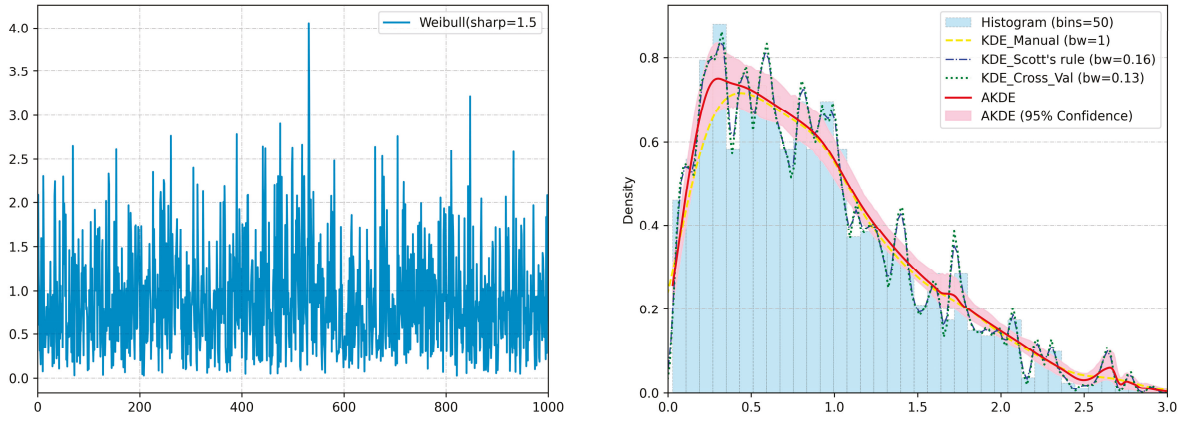
where $\mathrm{Bias}\left(\hat{f}_h(y)\right) = \frac{h^2}{2}\mu_2(K)f''(x) + o\left(h^2\right)$ and $\mathrm{Var}\left(\hat{f}_h(y)\right) = \frac{1}{nh}R(K)f(x) + o\left(\frac{1}{nh}\right)$ are the bias and variance operators, respectively and where $\mu_2(K) = \int y^2 K(y)dy > 0$ and $R(K) = \int K^2(y)dy$.

$$\mathrm{AMISE}\left(\hat{f}_h\right) = \frac{1}{nh}R(K) + \frac{h^4}{4}\mu_2(K)^2 R(f'') \tag{20}$$

$$h_{\mathrm{AMISE}} = \left[\frac{R(K)}{\mu_2(K)^2 R(f'')}\right]^{1/5} n^{-1/5} \tag{21}$$

$$h_{\mathrm{AMISE}} = 1.06\sigma n^{-1/5} \tag{22}$$

where $R(f'') = \int [f''(y)]^2 dy$ and where σ and n are the sample standard deviation and number of points (1-dimensional), respectively. Note that a more robust estimate should consider the sample interquartile; see Ref. [47].



(**a**) (**b**)

Figure 4. Results of estimating an artificial sequence via KDE and AKDE. (**a**) Generated sequence; (**b**) kernel density estimations.

2.3. Proposed Framework

The main objective of this work is to establish a multilevel warning system by integrating various methods. By implementing prediction models, an optimization algorithm, and an adaptive estimator, we can naturally propose a framework for wind speed forecasting and establish a multilevel early warning system. The proposed framework comprises three main parts corresponding to the utilized algorithms: (i) using an optimized GRU model to predict the wind speed, (ii) performing AKDE and obtaining various confidence intervals, and finally, (iii) deciding the warning level. Figure 5 illustrates the main idea of this framework.

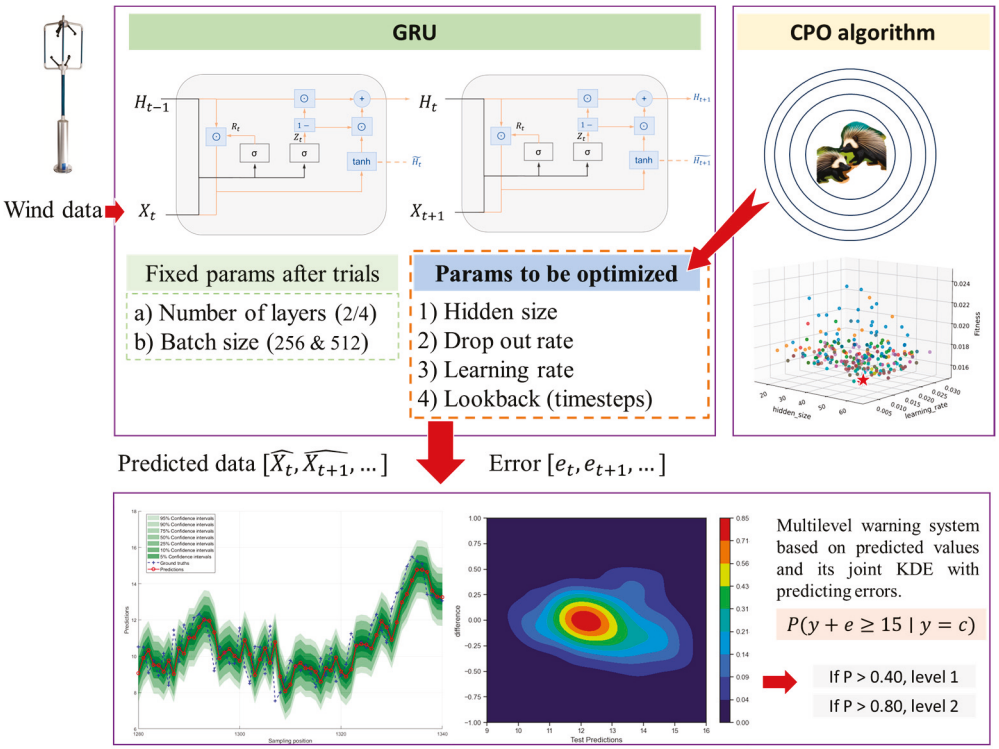


Figure 5. Framework of the multilevel warning system based on the optimized GRU and AKDE.

The warning system generates warning levels on the basis of the conditional probability of predicting values exceeding predefined thresholds. After training and optimizing the GRU, the joint KDE of the predictions and the predicted errors of the wind data can be obtained. Hence, this framework can not only predict the future wind speed y_0 but also evaluate the conditional probability of $P(y_0 + e \geq 15 \mid y = y_0)$, where e is the prediction error between y_0 and the ground truth. Here, 15 (m/s) serves as the threshold for the speed limit of the HSR train. Two rule-of-thumb values of 0.40 and 0.80 for P, corresponding to two warning levels, are used. That is, the wind speed at the next step y_0 is first predicted by the GRU or general predictive models; then, the conditional probability P is calculated and compared with threshold values of 0.40 and 0.80. Because the joint distribution is built on the basis of previous predictions and prediction errors, it is reasonable to infer that this multilevel warning method can consider the randomness of predictions and has greater credibility.

3. Results and Analysis

3.1. Performance Criteria of the Prediction Models

To objectively evaluate the prediction performance of the GRU and other RNNs used in this work, various metrics, including the root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and coefficient of determination (R^2), are used. The expressions of these metrics are shown in Equations (23)–(26).

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2} \tag{23}$$

$$\text{MAE} = \frac{1}{N}\sum_{t=1}^{N}\left|y_t - \hat{y}_t\right| \tag{24}$$

$$\text{MAPE} = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{y_t - \hat{y}_t}{y_t}\right| \tag{25}$$

$$R^2 = 1 - \frac{\sum_{t=1}^{N}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{N}(y_t - \overline{y_t})^2} \tag{26}$$

where N is the length of the time series and where y_t and $\hat{y}_t$ are the actual and predicted values, respectively.

The multilevel warning system for speed limits actually serve as a binary classification model. The decision for the model is whether to implement a speed limit, whereas the real situation is whether the actual wind speed reaches 15 m/s. Hence, predictions can be categorized as true positive (TP), false positive (FP), true negative (TN), or false negative (FN). Furthermore, the true positive rate (TPR, also known as recall or sensitivity), false positive rate (FPR), and overall accuracy can be defined, as shown in Table 3.

Table 3. Definitions of the prediction results of the warning systems.

Total Population = P + N		**Predicted Conditions**		**Overall Accuracy:** $\frac{\mathbf{TP+TN}}{\mathbf{TP+TN+FP+FN}}$
		Positive	**Negative**	
Actual conditions	Positive	TP	FN	$\mathrm{TPR} = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$
	Negative	FP	TN	$\mathrm{FPR} = \mathrm{FP}/(\mathrm{FP}+\mathrm{TN})$

3.2. Results of Optimization for RNNs

An accurate prediction of the wind speed is essential to the final decision of the warning system. To maximize the performance of RNNs, three optimization algorithms, CPO, GWO, and the WOA, are used to optimize the parameters of a two-layer GRU model. The details are as follows: (i) A sequence of length 3200 in D1-1 (see Section 2.1) is selected for optimization. (ii) The training-test split ratio, batch size, and training epochs are 80/20, 512, and 100, respectively. (iii) Four features, including the wind speed and its first-order difference, temperature, and humidity, are used to predict the next step wind speed. (iv) For a fair comparison, the population size (agent number) and iteration epoch for all algorithms are set as 30 and 20, respectively.

Table 4 summarizes the range, initial value, and optimized selection of the parameters. Additionally, the obtained parameters are used to train and test the entire D1-1 sequence, and the model performance is shown in Table 4, where the best results are marked in bold and underline. The results of the optimization and prediction are shown in Figure 6.

Table 4. Optimization using various algorithms and their performance.

Model	**Parameters**	**Range**	**Initial Searching Range**			**Optimized Parameters**		
			CPO	**GWO**	**WOA**	**CPO**	**GWO**	**WOA**
GRU	Learning rate	0.001~0.03	0.001~0.029	0.001~0.029	0.002~0.029	0.007	0.03	0.020
	Dropout rate	0~0.3	0.002~0.285	0.010~0.291	0.003~0.289	0.002	0.082	0.211
	Hidden size	16~64	16~62	17~63	21~62	54	53	21
	Lookback	16~128	18~128	19~120	24~128	121	128	121
Model performance (on the test set of the entire D1-1)					MAPE:	**1.28%**	1.54%	1.52%
					MAE:	**0.127**	0.145	0.145
					RMSE:	**0.194**	0.220	0.222
					R^2:	**0.992**	0.990	0.990

Note: The best results are in bold and underlined.

As shown in Table 4 and Figure 6, all optimization algorithms have a board and satisfactory range for parameters since the initial search. According to the convergence curves in Figure 6a, the CPO obtains its optimized result at epoch 7 with an MAE loss of 0.00851, which is faster and better than those of the other two approaches. Figure 6b,c clearly show the search trends of lookback size, hidden size, and learning rate when the CPO is used. Along with Table 4, it is found that all optimization methods select a similar

and relatively larger lookback size but vary in the choices of the other three parameters. For example, the CPO tends to select a lower learning rate and dropout rate and a higher hidden size, whereas the WOA performs the opposite. When the obtained parameters are applied to the full D1-1 dataset (26,280 points), as shown in Figure 6d, the performance of the GRUs is consistent with their performance on the shorter sequence (3200 points). That is, all optimized GRUs can accurately predict the wind speed, and among them, the GRU optimized by CPO performs the best predictions in terms of various metrics, including the MAPE, MAE, RMSE, and R^2.

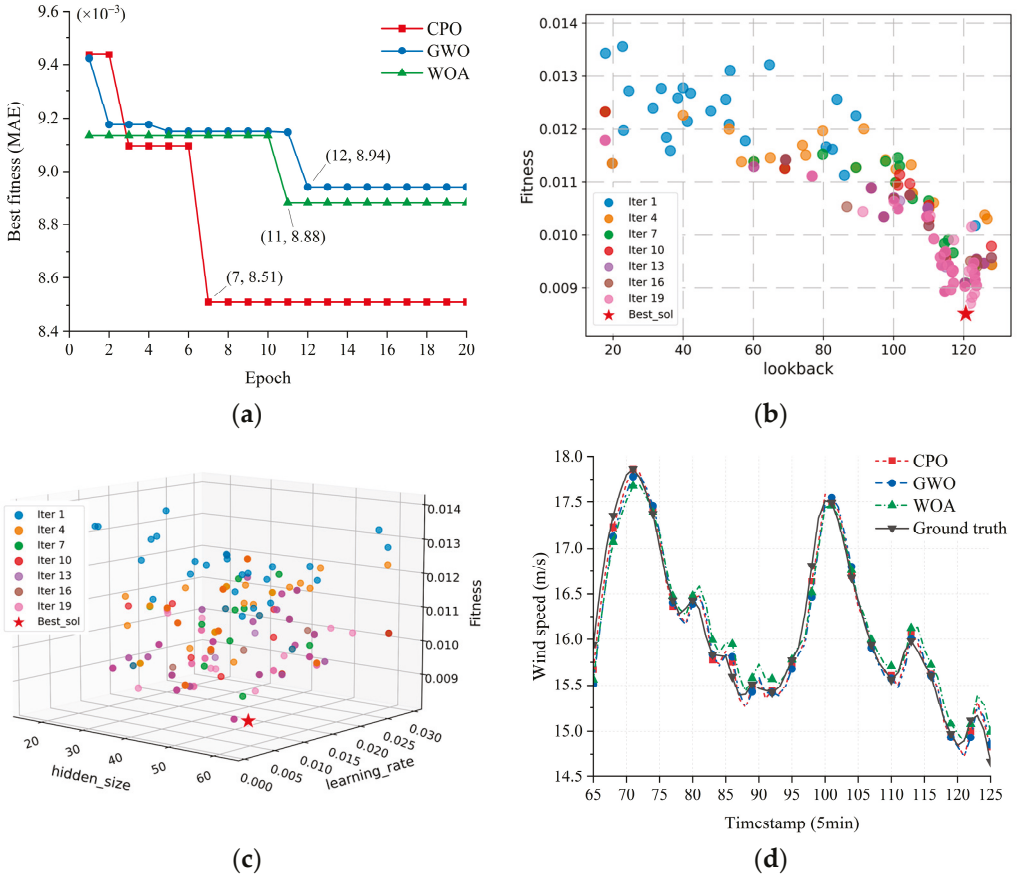


Figure 6. Optimization results of various methods and corresponding model performances. (**a**) Convergence curve of optimization methods; (**b**) searching trace of agents in the CPO for lookback size; (**c**) searching trace of agents in the CPO for hidden size and learning rate; (**d**) comparison of true wind speeds and predictions made by the GRU with different optimized parameters.

3.3. Results of Various Prediction Models

To demonstrate and further compare the prediction performance of diverse statistical and RNN models, another randomly selected continuous sequence of 3200 points from D1-2 is chosen for prediction. Methods such as ARIMA, extreme learning machine (ELM [48]), RNN (GRU, LSTM, and bidirectional LSTM), and transformer [49] are employed. For machine learning methods, parameters such as the lookback size, dropout rate, learning rate, and hidden size in RNNs, as well as the hidden size in the online recurrent ELM (ORELM [50]), are optimized via the CPO. To determine the order of ARIMA, an autocorrelation function and partial autocorrelation function measures are utilized. The selected parameters and prediction results are shown in Table 5 and Figure 7.

Table 5. Parameters and prediction performance of various models.

Model	Key Parameters				Metrics			
					MAPE	MAE	RMSE	R^2
	Learning rate	Dropout rate	Hidden size	Lookback				
GRU	0.01	0.02	63	24	**2.06%**	0.2108	**0.1280**	**0.9964**
LSTM	0.02	0.01	44	22	2.14%	**0.2105**	0.1305	0.9964
BiLSTM	0.01	0.001	57	21	2.25%	0.2187	0.1354	0.9961
	p	d	q	—				
ARIMA *	1	1	3	—	2.20%	0.2112	0.1350	0.9963
	3	1	1	—	2.19%	0.2111	0.1345	0.9964
	Hidden size	Reg_lambda	Forgetting factor					
ELM	30	0.001	—		2.50%	0.2195	0.1425	0.9961
ORELM	29	0.001	0.9485		3.81%	0.2142	0.3092	0.9922
	Layers	Dropout rate	No. of heads	FFN_dim				
Transformer *	2	0.1	18	256	2.83%	0.2849	0.2066	0.9928

* Note: Parameters p, d, and q in ARIMA are the order of autoregressive, degree of differencing, and order of moving average, respectively; for the transformer model, FFN_dim refers to the dimension of the feedforward layer. The best results of all methods are in bold and underlined.

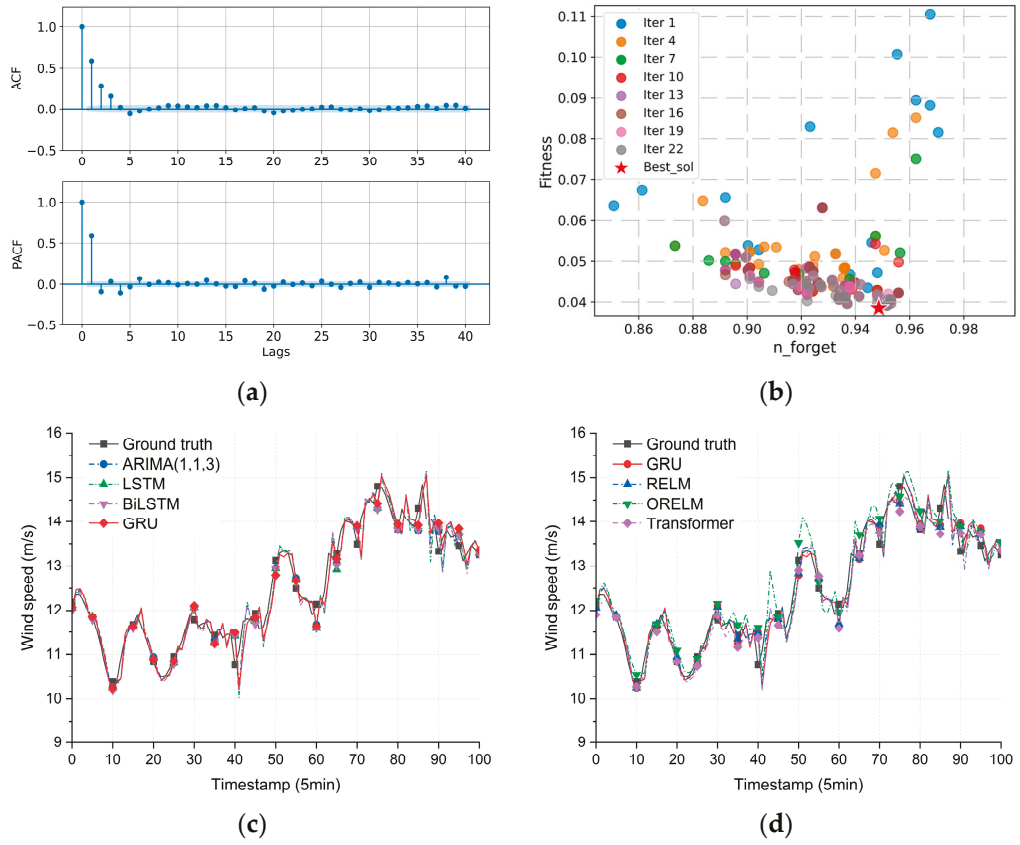


Figure 7. Predictions were made via various models. (**a**) The ACF and PACF of the sequence; (**b**) searching traces of the forgetting factor in ORELM via the CPO; (**c**) comparison between RNNs and an ARIMA model; (**d**) comparison between the GRU, ELMs, and a transformer model.

Frankly speaking, all applied methods show powerful applicability for the prediction of wind speed data. The prediction errors of ORELM are relatively high because the method uses an online recurrent training method, which benefits its application in real-time problems. Nevertheless, it is still more strongly affected by the choice of parameters, such as the forgetting factor; hence, the implementation of a proper optimization method is necessary. As shown in Figure 7b, the overall loss can be greatly reduced when optimized parameters are used. Moreover, it is clear that the optimized GRU model performs better

than the other models do in nearly all the statistical metrics, except for the MAE. In addition, there are no distinct lag patterns in the prediction values, which is essential for the subsequent development of early warning systems. With these results, the CPO-GRU method is then applied to all the collected samples of wind speed. The prediction performance of the CPO-GRU on the collected dataset is summarized in Table 6. Note that the agent number is 30, the optimization epoch is 20, and the training epoch is 100. As shown in Table 6, the optimized parameters generally yield good prediction performance, indicating the effectiveness of the proposed method.

Table 6. Predicting results on the collected dataset via the CPO-GRU method.

Dataset	Parameters				Metric			
	Learning Rate	Dropout Rate	Hidden Size	Lookback	R^2	MAPE	MAE	RMSE
D1-1	0.0155	0.2320	39	26	0.9987	1.78%	0.0885	0.1416
D1-2	0.0149	0.1261	50	17	0.9913	3.84%	0.1744	0.2882
D1-3	0.0192	0.0030	45	16	0.9965	2.09%	0.1043	0.1838
D1-4	0.0128	0.0063	52	55	0.9917	1.86%	0.1192	0.3060
D2-1	0.0211	0.1958	60	92	0.9990	1.72%	0.0603	0.1045
D2-2	0.0145	0.0097	20	125	0.9931	3.58%	0.1164	0.1893
D2-3	0.0018	0.0483	60	128	0.9961	2.42%	0.1192	0.2427
D2-4	0.0299	0.0390	50	90	0.9982	0.98%	0.0666	0.1384
D2-1	0.0170	0.1583	37	17	0.9978	1.13%	0.0742	0.1421
D2-2	0.0129	0.0027	34	114	0.9687	4.34%	0.1456	0.3681
D2-3	0.0156	0.1093	33	22	0.9970	2.10%	0.0809	0.1576
D2-4	0.0141	0.0565	50	102	0.9969	0.97%	0.0988	0.1925

3.4. Results of Multilevel Warning

Previous analysis has demonstrated the effectiveness of the proposed prediction method. With accurate prediction values, it is then applicable to form an early warning system of the wind speed for decision making in train operations. As shown in Figure 5, a multilevel early warning system is proposed on the basis of the conditional probability of predicting values exceeding predefined thresholds. To achieve this, for each dataset (D1, D2, and D3), a joint KDE of wind speed predictions and prediction errors is first calculated from 80% of the data points (i.e., the training set) and then used for evaluating the conditional probability $P(y_0 + e \geq 15 \mid y = y_0)$ of the remaining 20%, where y_0 and e are the prediction of the wind speed and the prediction error between y_0 and the ground truth, respectively. Notably, in practical uses, there is no need for data splitting in this phase as previous monitoring and prediction data can be used to form the joint KDE. As an example, the joint kernel density and related cumulative distribution function (CDF) of the prediction and prediction errors of D1-1 are shown in Figure 8a,b, respectively.

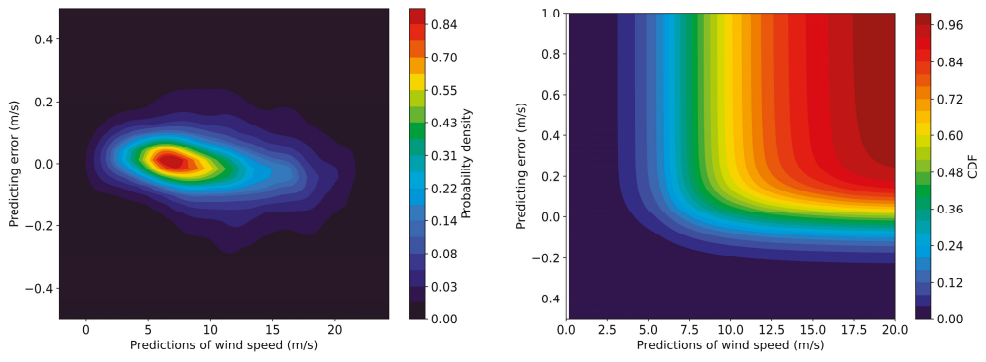


Figure 8. The KDE of predictions and prediction errors. (**a**) Joint kernel density; (**b**) CDF.

To demonstrate the effectiveness of the proposed multilevel warning system, Table 7 shows the decisions at certain positions made by the deterministic and probability-based methods when predicting data. For the deterministic method, the decision of whether to limit the train speed completely depends on the accuracy of the prediction model. When the predicted speed exceeds 15 m/s, a decision to limit the train speed will be made (e.g., points 3030 and 3031); otherwise, an opposite decision will be made (e.g., points 3025 and 3505). As shown in Table 7, when the actual wind speed is close to the threshold speed, the prediction performance is never sufficient at some points. Nevertheless, when a probabilistic method is utilized, a better result can be achieved, e.g., at points 3030, 3031, 3612, and 4896. Note that the decision of level 1 is made when the conditional probability is greater than 0.4, whereas that of level 2 is 0.8.

Table 7. Decisions made by various methods at points close to the critical wind speed.

Sequence ID	Values			Ideal Decision	Deterministic		Probabilistic			
	True (m/s)	Pred. (m/s)	p *		Decisions	Y/N	Decision_1	Y/N	Decision_2	Y/N
3025	15.01	14.91	0.1218	1	0	N	0	N	0	N
3030	14.99	15.02	0.5695	0	1	N	1	N	0	Y
3031	14.85	15.01	0.5136	0	1	N	1	N	0	Y
3505	15.09	14.97	0.2987	1	0	N	0	N	0	N
3508	14.94	15.08	0.8010	0	1	N	1	N	1	N
3514	15.02	14.94	0.2127	1	0	N	0	N	0	N
3547	14.87	15.08	0.8010	0	1	N	1	N	1	N
3612	14.97	15.01	0.5136	0	1	N	1	N	0	Y
4893	15.03	14.93	0.1773	1	0	N	0	N	0	N
4896	14.98	15.07	0.7637	0	1	N	1	N	0	Y

Note that p * is the conditional probability of a given predicting value (Pred.) made by CPO-GRU. For decisions, 1 means the train speed should be limited, while 0 indicates the opposite. For judgements, Y means a wrong decision while N indicates the opposite. Improved results are marked with background colors.

To further compare the performance of the proposed framework, a general LSTM model is used to provide a benchmark for decision making. The LSTM model is trained for 100 epochs at a learning rate of 0.01, a dropout rate of 0.1, a hidden size of 64, and a lookback size of 64. The results are summarized in Table 8. Here, the event is positive when the actual speed exceeds 15 m/s. The proposed CPO-GRU-KDE framework generally improves the overall accuracy, especially the TPR (see Table 3), of the prediction, which is highly significant for the presented case.

Table 8. Result of warning decisions for the collected dataset.

Dataset	Total No. of Positive (v $\geq$ 15 m/s)	Total No. of TP				TPR		Overall Accuracy	
		LSTM	CPO_GRU	CPO_GRU_KDE (CGK)		LSTM	CGK	LSTM	CGK
				Level 1	Level 2				
D1-1	185	135	164	164	168	0.7297	0.9081	0.9903	0.9967
D1-2	3	2	2	2	2	0.6667	0.6667	0.9996	0.9996
D1-3	15	0	8	8	6	0.0000	0.5333	0.9971	0.9988
D1-4	103	82	90	90	89	0.7961	0.8738	0.9959	0.9975
D2-1	53	40	52	52	50	0.7547	0.9811	0.9975	0.9998
D2-2	0	0	0	0	0	—	—	1.0000	1.0000
D2-3	293	289	292	292	291	0.9863	0.9966	0.9965	0.9977
D2-4	73	67	61	66	68	0.9178	0.9315	0.9977	0.9990
D3-1	83	79	81	81	79	0.9518	0.9518	0.9990	0.9992
D3-2	0	0	0	0	0	—	—	1.0000	1.0000
D3-3	0	0	0	0	0	—	—	1.0000	1.0000
D3-4	437	393	408	415	410	0.8993	0.9497	0.9910	0.9926

3.5. Discussions

Previous analysis has demonstrated the advances of the proposed framework. In this section, further discussions are presented to explore the reason behind these advances. In this work, to enhance the model performance, optimization algorithms are used for RNN, ELM, and transformer models. In general, using parameter optimization will not affect the properties of the model but will only find better results in the solution space. Hence, the advantage of using parametric optimization is distinct. The final choice of CPO is determined by its actual performance, as shown in Table 7. This is also the case for selecting the optimal model. As shown in Table 6, the GRU model performs the best over the other models. In addition, GRU has a simpler and more efficient structure when compared with other RNN models. Nevertheless, it is worth noting that using RNN models for time series may encounter the lagging problem—the predictive value lagging the truth value—especially when having only one training feature. Figure 9 plots the predicting results using only the previous wind speed data with MAE as the loss function. It is obvious that despite having fine metrics (R^2 of 0.98), the model actually provides ineffective predictions because of the occurrence of the lagging problem, which means the prediction is dominantly affected by the last one-step data. In other words, the model performance is similar to that of a naive forecast. Under this situation, the warning systems, especially those using deterministic methods, cannot perform well because the warning decisions would be lagging.

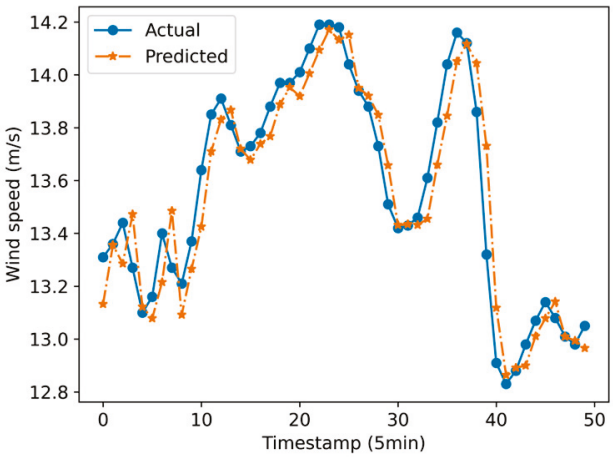


Figure 9. Lagging problem when using limited features in an RNN.

To further demonstrate the effectiveness of using probabilistic methods, two more threshold values, 12 m/s and 14 m/s, are selected to calculate the conditional probability, i.e., $P(y_0 + e \geq 12 \mid y = y_0$ and $P(y_0 + e \geq 14 \mid y = y_0$, respectively. The same joint kernel density and CDF of D1-1 dataset shown in Figure 8 are used. As plotted in Figure 10a, the conditional probability of predictions arises with the increase in threshold value. Therefore, a smaller threshold provides a broader envelope. Then, the prediction results are plotted together with their conditional probability in Figure 10b for comparison. It can be observed how the conditional probability follows the prediction values, which is achieved by the implementation of predicting error in historic data. On this basis, the final prediction values are statistically evaluated by considering both the current result of model prediction and previous database of prediction errors. This is achieved beyond the normal process of model prediction. Therefore, the subsequent decision of a warning system for the future data can be improved. As long as the model performance stays consistent, the predicting accuracy can be improved by compensating the predicting error.

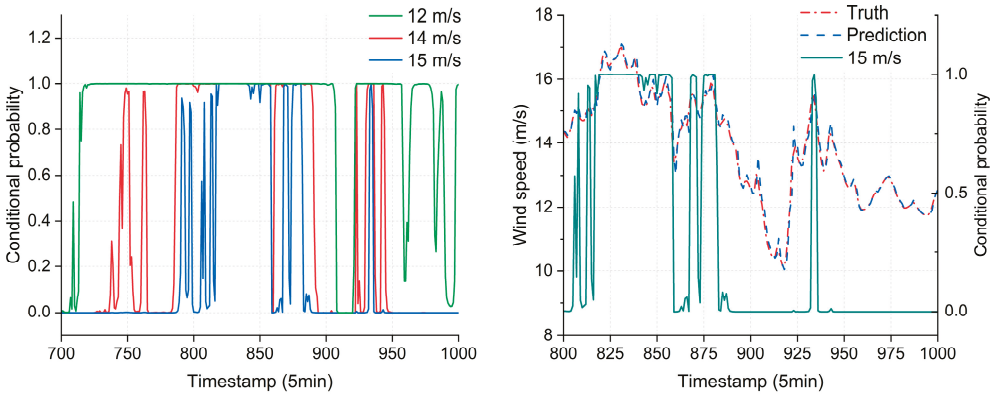


Figure 10. A demonstration of using probabilistic approaches for the warning system. (**a**) Conditional probability of predictions of D1-1 under various threshold values; (**b**) conditional probability and predictions.

4. Conclusions

In this work, we proposed a framework for wind speed prediction and related multi-level warning systems based on the combination of an optimized neural network and an adaptive kernel density estimator. The performance of various optimization algorithms, machine learning-based models, and probabilistic decision processes were evaluated. The results proved that the proposed framework performs well on the collected data. The main conclusions are as follows:

1. Recurrent networks perform well in predicting wind speeds. Nevertheless, their performance can be efficiently optimized by using a proper algorithm. A comparison of various algorithms, including ARIMA, ELM, and transformer, demonstrates that the GRU performs best on the collected database.
2. The results of various optimization algorithms indicate the significance of parameter optimization. The search trace of diverse parameters can provide valuable information for model selection. For example, in the present case, the lookback size greatly affects the model performance, while the model can achieve similar performance at diverse dropout rates, learning rates, or hidden sizes.
3. An analysis of the prediction results indicates that the CPO-GRU model outperforms the other combinations, with metrics of 1.28%, 0.127, 0.194, and 0.992 for the MAPE, MAE, RMSE, and R^2, respectively, in the present case. In addition, it achieves similar and excellent performance on all collected databases.
4. By using adaptive kernel estimators, the joint kernel density and cumulative distribution function of the predicted values and prediction errors can be obtained, thereby calculating the conditional probability at a given prediction of the wind speed. A comparison between the deterministic and probabilistic methods indicates that all methods yield high overall accuracy—due to the relatively large sample size—but also that the proposed framework can significantly address the TPR, which is valuable for practical decision making, especially when the predictions are near the critical value.

It is worth noting the proposed framework may encounter the lagging problem, which is a common issue in the use of RNNs. In this work, by using the L1loss function and various input features, such as temperature and humidity, the lagging problem is not obvious in the present results. Nevertheless, when the features are limited in practical use, the prediction performance should be further examined. In addition, by using joint kernel estimation and a related probabilistic approach, an advanced warning system was successfully established. The achievement of this system relies on a sufficient historic database of predictions and corresponding prediction errors. If this is not the case, the deterministic warning method,

i.e., comparing the newest prediction value with the threshold value, should be used instead. Further studies should be conducted to address these issues.

Author Contributions: Methodology and writing—original draft, P.W.; data curation and formal analysis, P.W., Q.L., H.Z., X.C. and R.Y.; supervision and writing—review and editing, F.G.; investigation and software, P.W. and F.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study may be available on reasonable request.

Conflicts of Interest: Authors Qiuliang Long, Hu Zhang, Xu Chen, and Ran Yu were employed by Hunan Harbor Engineering Corporation Limited. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

References

1. Liu, D.; Wang, Q.; Zhong, M.; Lu, Z.; Wang, J.; Wang, T.; Lv, S. Effect of wind speed variation on the dynamics of a high-speed train. *Veh. Syst. Dyn.* **2019**, *57*, 247–268.
2. Liu, D.; Wang, T.; Liang, X.; Meng, S.; Zhong, M.; Lu, Z. High-speed train overturning safety under varying wind speed conditions. *J. Wind Eng. Ind. Aerodyn.* **2020**, *198*, 104111.
3. Pan, D.; Liu, H.; Li, Y. A short-term forecast method for wind speed along Golmud-Lhasa section of Qinghai-Tibet railway. *China Railw. Sci.* **2008**, *29*, 129–133.
4. Kobayashi, N.; Shimamura, M. Study of a strong wind warning system. *Jr East Tech. Rev.* **2003**, *2*, 61–65.
5. Hoppmann, U.; Koenig, S.; Tielkes, T.; Matschke, G. A short-term strong wind prediction model for railway application: Design and verification. *J. Wind Eng. Ind. Aerodyn.* **2002**, *90*, 1127–1134.
6. Jing, H.; Zhong, R.; He, X.; Wang, H. Multi-Level Early Warning Method for Gale Based on LSTM-GMM Model. *China Railw. Sci.* **2023**, *44*, 221–228.
7. Soman, S.S.; Zareipour, H.; Malik, O.; Mandal, P. A review of wind power and wind speed forecasting methods with different time horizons. In Proceedings of the North American Power Symposium 2010, Arlington, TX, USA, 26–28 September 2010; pp. 1–8.
8. Hu, J.; Heng, J.; Wen, J.; Zhao, W. Deterministic and probabilistic wind speed forecasting with de-noising-reconstruction strategy and quantile regression based algorithm. *Renew. Energy* **2020**, *162*, 1208–1226.
9. Jung, J.; Broadwater, R.P. Current status and future advances for wind speed and power forecasting. *Renew. Sustain. Energy Rev.* **2014**, *31*, 762–777.
10. Kavasseri, R.G.; Seetharaman, K. Day-ahead wind speed forecasting using f-ARIMA models. *Renew. Energy* **2009**, *34*, 1388–1393.
11. Maatallah, O.A.; Achuthan, A.; Janoyan, K.; Marzocca, P. Recursive wind speed forecasting based on Hammerstein Auto-Regressive model. *Appl. Energy* **2015**, *145*, 191–197.
12. Liu, H.; Mi, X.; Li, Y. Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, LSTM network and ELM. *Energy Convers. Manag.* **2018**, *159*, 54–64.
13. Cassola, F.; Burlando, M. Wind speed and wind energy forecast through Kalman filtering of Numerical Weather Prediction model output. *Appl. Energy* **2012**, *99*, 154–166.
14. Song, J.; Wang, J.; Lu, H. A novel combined model based on advanced optimization algorithm for short-term wind speed forecasting. *Appl. Energy* **2018**, *215*, 643–658.
15. Neshat, M.; Nezhad, M.M.; Abbasnejad, E.; Mirjalili, S.; Tjernberg, L.B.; Garcia, D.A.; Alexander, B.; Wagner, M. A deep learning-based evolutionary model for short-term wind speed forecasting: A case study of the Lillgrund offshore wind farm. *Energy Convers. Manag.* **2021**, *236*, 114002.
16. Wan, J.; Liu, J.; Ren, G.; Guo, Y.; Yu, D.; Hu, Q. Day-ahead prediction of wind speed with deep feature learning. *Int. J. Pattern Recognit. Artif. Intell.* **2016**, *30*, 1650011.
17. Zhang, W.; Wang, J.; Wang, J.; Zhao, Z.; Tian, M. Short-term wind speed forecasting based on a hybrid model. *Appl. Soft Comput.* **2013**, *13*, 3225–3233.
18. Kiplangat, D.C.; Asokan, K.; Kumar, K.S. Improved week-ahead predictions of wind speed using simple linear models with wavelet decomposition. *Renew. Energy* **2016**, *93*, 38–44.
19. Liu, H.; Tian, H.-Q.; Li, Y.-F. An EMD-recursive ARIMA method to predict wind speed for railway strong wind warning system. *J. Wind Eng. Ind. Aerodyn.* **2015**, *141*, 27–38.
20. Zhang, Y.; Zhao, Y.; Kong, C.; Chen, B. A new prediction method based on VMD-PRBF-ARMA-E model considering wind speed characteristic. *Energy Convers. Manag.* **2020**, *203*, 112254.
21. Xiao, L.; Shao, W.; Jin, F.; Wu, Z. A self-adaptive kernel extreme learning machine for short-term wind speed forecasting. *Appl. Soft Comput.* **2021**, *99*, 106917.

22. Li, L.; Chang, Y.; Tseng, M.; Liu, J.; Lim, M.K. Wind power prediction using a novel model on wavelet decomposition-support vector machines-improved atomic search algorithm. *J. Clean. Prod.* **2020**, *270*, 121817.
23. Qian, Z.; Pei, Y.; Zareipour, H.; Chen, N. A review and discussion of decomposition-based hybrid models for wind energy forecasting applications. *Appl. Energy* **2019**, *235*, 939–953.
24. Marović, I.; Sušanj, I.; Ožanić, N. Development of ANN model for wind speed prediction as a support for early warning system. *Complexity* **2017**, *2017*, 3418145.
25. Gou, H.; Chen, X.; Bao, Y. A wind hazard warning system for safe and efficient operation of high-speed trains. *Autom. Constr.* **2021**, *132*, 103952.
26. Wang, L.; Li, X.; Bai, Y. Short-term wind speed prediction using an extreme learning machine model with error correction. *Energy Convers. Manag.* **2018**, *162*, 239–250.
27. Santhosh, M.; Venkaiah, C.; Kumar, D.V. Short-term wind speed forecasting approach using ensemble empirical mode decomposition and deep Boltzmann machine. *Sustain. Energy Grids Netw.* **2019**, *19*, 100242.
28. Liu, Y.; Zhang, Z.; Huang, Y.; Zhao, W.; Dai, L. Hybrid neural network-aided strong wind speed prediction along rail network. *J. Wind Eng. Ind. Aerodyn.* **2024**, *252*, 105813.
29. Wang, L.; Guo, Y.; Fan, M.; Li, X. Wind speed prediction using measurements from neighboring locations and combining the extreme learning machine and the AdaBoost algorithm. *Energy Rep.* **2022**, *8*, 1508–1518.
30. Chen, Y.; Dong, Z.; Wang, Y.; Su, J.; Han, Z.; Zhou, D.; Zhang, K.; Zhao, Y.; Bao, Y. Short-term wind speed predicting framework based on EEMD-GA-LSTM method under large scaled wind history. *Energy Convers. Manag.* **2021**, *227*, 113559.
31. Wang, H.; Xiong, M.; Chen, H.; Liu, S. Multi-step ahead wind speed prediction based on a two-step decomposition technique and prediction model parameter optimization. *Energy Rep.* **2022**, *8*, 6086–6100.
32. Zhang, C.; Ji, C.; Hua, L.; Ma, H.; Nazir, M.S.; Peng, T. Evolutionary quantile regression gated recurrent unit network based on variational mode decomposition, improved whale optimization algorithm for probabilistic short-term wind speed prediction. *Renew. Energy* **2022**, *197*, 668–682.
33. Najibi, F.; Apostolopoulou, D.; Alonso, E. Enhanced performance Gaussian process regression for probabilistic short-term solar output forecast. *Int. J. Electr. Power Energy Syst.* **2021**, *130*, 106916.
34. WRDB: Wind Resource Database. National Renewable Energy Laboratory. Available online: https://wrdb.nrel.gov/ (accessed on 16 August 2024).
35. Kaveh, A.; Dadras, A. A novel meta-heuristic optimization algorithm: Thermal exchange optimization. *Adv. Eng. Softw.* **2017**, *110*, 69–84.
36. Abdel-Basset, M.; Mohamed, R.; Abouhawwash, M. Crested Porcupine Optimizer: A new nature-inspired metaheuristic. *Knowl.-Based Syst.* **2024**, *284*, 111257.
37. Liang, J.J.; Qu, B.Y.; Suganthan, P.N. Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. *Comput. Intell. Lab. Zhengzhou Univ. Zhengzhou China Tech. Rep. Nanyang Technol. Univ. Singap.* **2013**, *635*, 2014.
38. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61.
39. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67.
40. Abualigah, L.; Shehab, M.; Alshinwan, M.; Alabool, H. Salp swarm algorithm: A comprehensive survey. *Neural Comput. Appl.* **2020**, *32*, 11195–11215.
41. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
42. Liu, X.; Lin, Z.; Feng, Z. Short-term offshore wind speed forecast by seasonal ARIMA-A comparison against GRU and LSTM. *Energy* **2021**, *227*, 120492.
43. ArunKumar, K.; Kalaga, D.V.; Kumar, C.M.S.; Kawaji, M.; Brenza, T.M. Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends. *Alex. Eng. J.* **2022**, *61*, 7585–7603.
44. Sheather, S.J. Density estimation. *Stat. Sci.* **2004**, *19*, 588–597.
45. Scott, D.W. *Multivariate Density Estimation: Theory, Practice, and Visualization*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
46. Shimazaki, H.; Shinomoto, S. Kernel bandwidth optimization in spike rate estimation. *J. Comput. Neurosci.* **2010**, *29*, 171–182.
47. Dhaker, H.; Ngom, P.; Mendy, P. New Approach for Bandwidth Selection in the Kernel Density Estimation Based on β-Divergence. 2016. Available online: https://hal.science/hal-01297034/file/articll.pdf (accessed on 16 August 2024).
48. Zhou, S.; Liu, X.; Liu, Q.; Wang, S.; Zhu, C.; Yin, J. Random Fourier extreme learning machine with $\ell 2$, 1-norm regularization. *Neurocomputing* **2016**, *174*, 143–153.
49. Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; Sun, L. Transformers in time series: A survey. *arXiv* **2022**, arXiv:2202.07125.
50. Park, J.-M.; Kim, J.-H. Online recurrent extreme learning machine and its application to time-series prediction. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 1983–1990.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Multi-Task Forecasting of the Realized Volatilities of Agricultural Commodity Prices

Rangan Gupta [1,*] and Christian Pierdzioch [2]

[1] Department of Economics, University of Pretoria, Private Bag X20, Hatfield 0028, South Africa
[2] Department of Economics, Helmut Schmidt University, Holstenhofweg 85, P.O. Box 700822, 22008 Hamburg, Germany; macroeconomics@hsu-hh.de
* Correspondence: rangan.gupta@up.ac.za

Abstract: Motivated by the comovement of realized volatilities (RVs) of agricultural commodity prices, we study whether multi-task forecasting algorithms improve the accuracy of out-of-sample forecasts of 15 agricultural commodities during the sample period from July 2015 to April 2023. We consider alternative multi-task stacking algorithms and variants of the multivariate Lasso estimator. We find evidence of in-sample predictability but scarce evidence that multi-task forecasting improves out-of-sample forecasts relative to a classic univariate heterogeneous autoregressive (HAR)-RV model. This lack of systematic evidence of out-of-sample forecasting gains is corroborated by extensive robustness checks, including an in-depth study of the quantiles of the distributions of the RVs and subsample periods that account for increases in the total spillovers among the RVs. We also study an extended model that features the RVs of energy commodities and precious metals, but our conclusions remain unaffected. Besides offering important lessons for future research, our results are interesting for financial market participants, who rely on accurate forecasts of RVs when solving portfolio optimization and derivatives pricing problems, and policymakers, who need accurate forecasts of RVs when designing policies to mitigate the potential adverse effects of a rise in the RVs of agricultural commodity prices and the concomitant economic and political uncertainty.

Keywords: agricultural commodities; realized volatility; multi-task forecasting

MSC: 62-08; 62-11; 62H12; 62P20

1. Introduction

Quite a number of empirical studies have been undertaken to shed light on the connectedness of volatility across agricultural commodities (see, for example, refs. [1–7], with [8] highlighting that accounting for co-volatility of Chinese futures of five agricultural commodities (corn, cotton, palm, wheat, and soybeans) improved the accuracy of volatility forecasts—in particular for corn, cotton, and wheat. It must be noted that the underlying spillovers of risk across agricultural commodities is not surprising, given that behavioral, macroeconomic, and financial shocks, which define the underlying state of these markets, tend to be common and that the commonality has grown stronger in recent years (see the detailed discussions in [9–11] in this regard). We contribute to this area of research by exploring whether stacking algorithms that have been developed in the recent bioinformatics literature can help to improve the accuracy of out-of-sample forecasts of the intraday data-based realized volatility (RV) of 15 important agricultural commodities during the daily sample period of July 2015 to April 2023.

An important advantage of using RV for our empirical analyses derives from the rich information contained in intraday data, besides being a consistent and asymptotically unbiased estimator of volatility [12–14]. In addition, RV is an observable and unconditional metric of "volatility". This, in turn, is unlike the latent processes underlying the class

of Generalized Autoregressive Conditional Heteroskedasticity (GARCH) and Stochastic Volatility (SV) models that have been widely used in predicting agricultural commodity price volatility (see [15,16] for reviews of this extensive literature). Moreover, the dynamics of RV can be easily modeled by means of the heterogeneous autoregressive (HAR)-RV model [17]. The HAR-RV model has been extensively studied in research on realized volatility, including that of agricultural commodities (as reviewed in [18–20]) because it is able to capture long-memory and multi-scaling properties of realized volatility, as reported by [21–23]. Because the HAR-RV model employs RVs at different time resolutions to model and predict RV, it can be interpreted as a simple empirical representation of the heterogeneous market hypothesis (HMH; Ref. [24], which stipulates that asset markets (in our case, markets for agricultural commodities) are populated by various types of market participants, such as investors, speculators and traders, who, in turn, in turn, vary in their sensitivity to information flows at high and low frequencies.

Another advantage of the HAR-RV model is that it can easily be adapted to a multi-task forecasting setting, i.e., a setting where a forecaster seeks to forecast not only the RV of a single agricultural commodity but the RVs of several agricultural commodities simultaneously. One possibility to address such a multi-task forecasting problem is to consider as a modeling framework one of the multivariate HAR-RV models with heteroskedastic error structures, as has been studied by, for instance, refs. [5,8,25–28]. The focus of many studies in this area, however, has been on modeling and forecasting co-volatilities (see, for example, refs. [5,27,29–32]). Moreover, applications of HAR-RV cum heteroskedastic error models are often restricted to settings where the number of RVs to be analyzed is relatively small, as was the case in [5,8], involving seven and five agricultural commodities, respectively. This is due to the fact that, in a multivariate setting, the number of parameters to be estimated rapidly increases in the dimension of the model unless a researcher is willing to impose restrictions on parameters and/or functional forms so as to obtain a parsimonious representation of the heteroskedastic error structures.

In our case, the dataset comprises the RVs of 15 agricultural commodities (and, in an extended model, the RVs of three additional important energy commodities and the RVs of five precious metals), so we use various computationally efficient multi-task stacking algorithms that have been proposed in the recent bioinformatics literature (along with a multivariate shrinkage estimator) to re-examine the out-of-sample predictability of the RVs of the commodities in our sample (for a recent application of stacking in a univariate forecasting exercise of stock returns, see [33]). We also focus on direct spillovers among the RVs as captured by a multi-task HAR-RV model and do not consider the issue of forecasting co-volatility, which requires the imposition of further structure on the residuals. The multi-task stacking algorithms are easy to implement, even when the dimension of the model is large. Moreover, they make it possible to employ and combine alternative popular machine learning algorithms that make it possible to estimate a multi-task HAR-RV model in a data-driven way, that is, without imposing any specific structure that restricts the spillover dynamics across the RVs a priori. Finally, the multi-task stacking algorithms can be set up in a way such that the resulting statistical model captures potential nonlinear structures in the data, an issue that certainly deserves special attention in the wake of the type of sudden outbreaks and clustering of volatility typical of financial markets and of markets for agricultural commodities as well. In the process, our paper adds to the growing literature on modeling and predicting the RVs of agricultural commodities by investigating the role of volatility spillovers; thus far, researchers in this literature have otherwise relied on realized moments (such as realized kurtosis and realized jumps) and various other predictors that relate, for example, to the state of financial and other (non-agricultural) commodity markets, investor sentiment, climate change-related risks, and infectious disease-related uncertainty (see, for example, [8,15,16,18–20,34–39]).

Agricultural commodities have become increasingly financialized [40–42]. This process has caused institutional investors to increase their holdings in agricultural commodities relative to traditional assets. Naturally, besides the academic value of our work, accurate

forecasts of the volatility of agricultural commodity prices are of key importance for investors because volatility is a core input in investment and portfolio allocation decisions, risk management, derivatives pricing, and assessments of hedging performance [43,44]. In addition, agricultural commodities comprise a large proportion of household consumption spending, implying that price volatility in agricultural commodities markets is likely to have substantial consequences for food security, especially as far as the economically vulnerable groups of the population are concerned [45–47]. Hence, from a policy perspective, it is important to produce accurate high-frequency forecasts of agricultural commodity price volatility so that policies can be discussed and implemented in a timely manner to protect vulnerable groups, in particular, from large and adverse food price fluctuations [48,49].

In order to present our empirical findings, we organize the rest of the paper as follows. In Section 2, we provide a description of the data we use in our study, while in Section 3, we outline our methods. In Section 4, we present our empirical results. In Section 5, we conclude the paper.

2. Data

In our empirical analysis, we use data on the RVs of 15 agricultural commodities. The data are available publicly for download from the Internet page of Professor Dacheng Xiu. (Internet address: https://dachxiu.chicagobooth.edu/#risklab. Data downloaded on 4 May 2024). The data are based on (Globex) data for the following 15 agricultural commodity futures: soybean oil futures (BO), cocoa futures (CC), corn futures (C), cotton no. 2 futures (CT), feeder cattle futures (FC), coffee C futures (KC), lumber futures (LB), live cattle futures (LC), lean hog futures (LH), orange juice futures (OJ), oat futures (O), sugar #11 futures (SB), soybean meal futures (SM), soybean futures (S), and CBOT wheat futures (W). After matching the data by date, the matched dataset starts on 27 July 2015 and ends on 28 April 2023. We plot the RV data of the agricultural commodities in Figure 1. The RVs display a discernible heterogeneity across the agricultural commodities, and they also exhibit the type of clusters and sudden outbursts characteristic of many financial market volatilities.

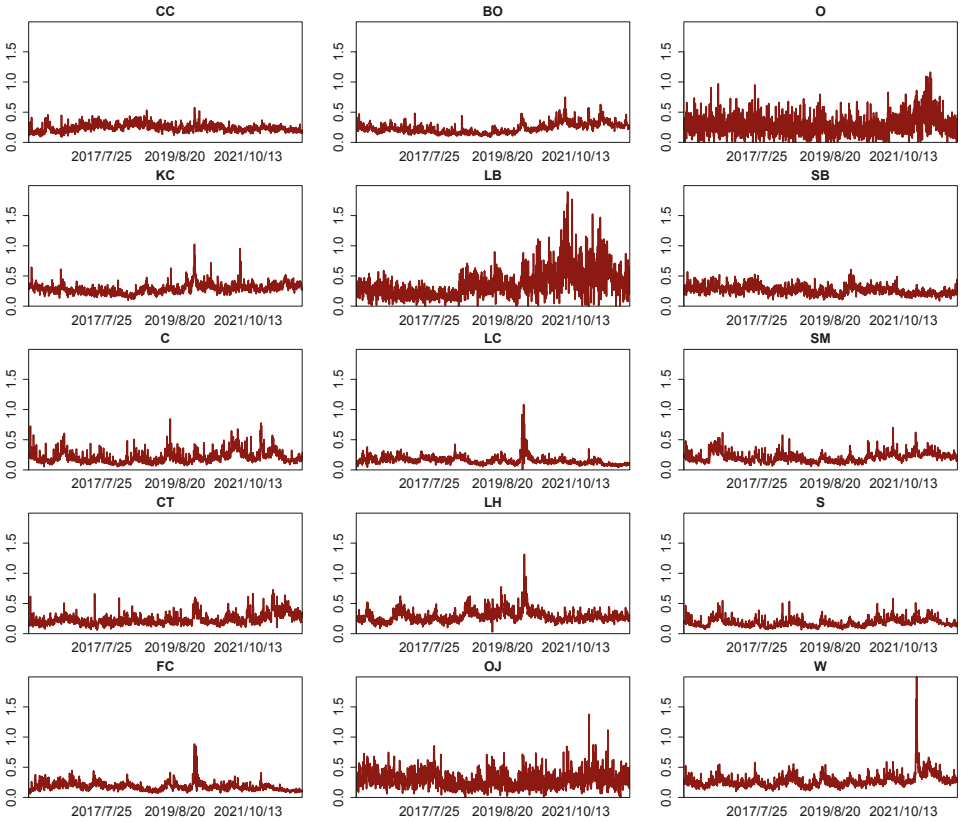


Figure 1. RVs of agricultural commodities.

In order to obtain a first glimpse of the comovement of the RVs, we plot their full-sample contemporaneous correlation matrix in Figure 2. The contemporaneous correlations vary from weakly negative to strongly positive, with the positive correlations mainly collected in the lower part of the matrix. For example, we observe strong positive contemporaneous correlations between C and S, S and SM, and LC and FC, among others. While the full-sample contemporaneous correlations shed light on an important feature of the data, one should bear in mind that the correlations do not inform about the question as to whether the comovement of the RVs can be exploited in a multi-task out-of-sample forecasting exercise to improve predictive accuracy at various forecast horizons.

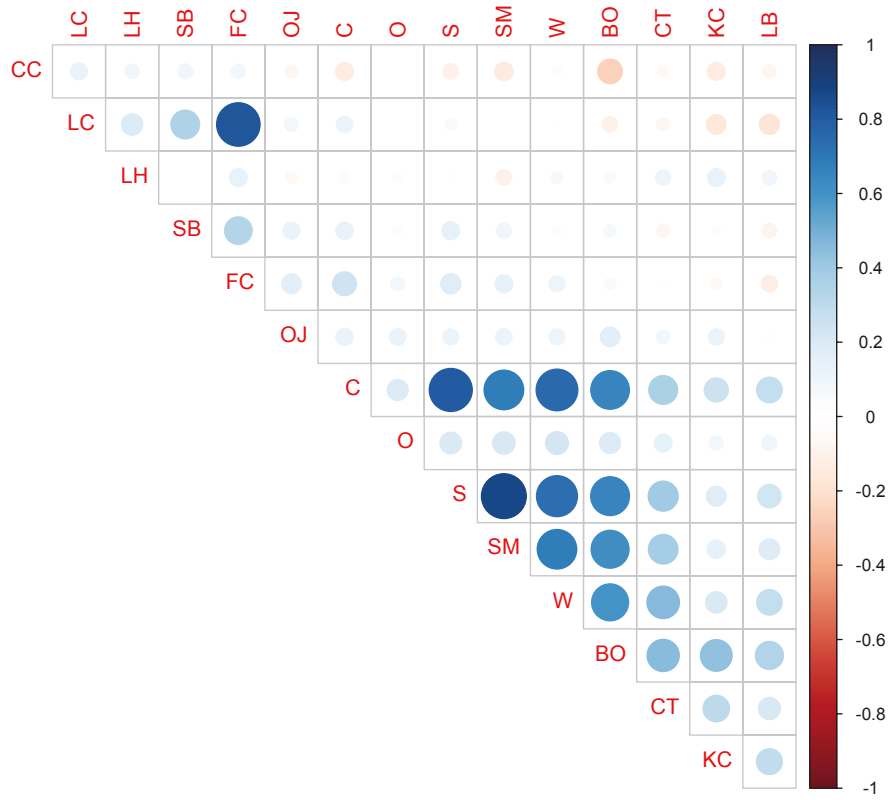


Figure 2. Full−sample correlation matrix.

3. Methods

3.1. Forecasting Models

We frame our empirical analysis in terms of the popular HAR-RV model developed by [17]. This model can be specified by the following equation:

$$RV_{t+h} = \beta_0 + \beta_1 RV_t + \beta_2 RV_{IF,t} + \beta_3 RV_{LF,t} + u_{t+h}, \quad (1)$$

which we estimate using the ordinary least squares (OLS) technique, where $\beta_j, j = 0, \ldots, 3$ are the coefficients to be estimated, u_{t+h} denotes a disturbance term, and RV_{t+h} denotes the average realized volatility over the forecast horizon (h). In our empirical research, we study one short, two intermediate, and one long forecast horizon. To this end, we specify $h = 1, 5, 10, 22$. The predictors are the daily realized volatility (RV_t), the intermediate-frequency (IF) realized volatility, ($RV_{t,IF}$), and the low-frequency (LF) realized volatility, ($RV_{t,LF}$). We define IF realized volatility as the average realized volatility from period $t-5$ to period $t-1$ and LF realized volatility as the average realized volatility from period $t-22$ to period $t-1$, as computed using the matched data.

We emphasize that in order to avoid non-negativity constraints and to bring the data closer to normality, we use the natural logarithm of the RV to estimate the HAR-RV model

(and its extension to the HAR-RV-S model), which accounts for potential spillover effects. However, we evaluate the resulting forecasts in terms of the anti-log of the RV.

The variant of the HAR-RV model that accounts for spillover effects, the HAR-RV-S model, is expressed by the following equation:

$$RV_{t+h,i} = \beta_0 + \beta_1 RV_{t,i} + \beta_2 RV_{IF,t,i} + \beta_3 RV_{LF,t,i} + \sum_{j \neq i} \left(\beta_{4,j} RV_{t,j} + \beta_{5,j} RV_{IF,t,j} + \beta_{6,j} RV_{LF,t,j} \right) + u_{t+h}, \quad (2)$$

where i is the agricultural commodity being studied and the index j denotes the other agricultural commodities. Hence, we obtain the HAR-RV-S model by adding the daily, intermediate-frequency, and low-frequency realized volatilities of the other agricultural commodities and, thereby, account for potential spillover effects at different time resolutions. We emphasize that the HAR-RV-S model captures direct spillover effects among the RVs, not the dynamics of co-volatilities.

We use the R language and environment for statistical computing ([50]; R version 4.3.1) to estimate our forecasting models and to compute all other results that we lay out in this research. We estimate the forecasting models either by means of a recursively expanding estimation window or by means of a rolling estimation window. We use 50% of the data to initialize the recursive estimations. Similarly, we use 50% of the data to define the length of a rolling estimation window. Finally, we use the root mean squared forecasting error (RMSFE) and the mean absolute forecasting error (MAFE) to evaluate the out-of-sample performance of the forecasting models, where we compute the ratio of the RMSFE (MAFE) of the HAR-RV-S model and the HAR-RV model to alleviate the interpretation of our empirical results. Hence, an RMSFE (MAFE) ratio smaller than unity implies that the HAR-RV-S model outperforms the HAR-RV model, while a ratio larger than unity signals that the HAR-RV model is the better forecasting model.

3.2. Stacking Algorithms

Given that our sample comprises 15 agricultural commodities and we have to consider (leaving the intercept term apart) a total of $15 \times 3 = 45$ predictors, we use computationally efficient stacking algorithms to estimate the forecasting model given in Equation (2).

The first stacking algorithm that we use (we call this algorithm the baseline stacking algorithm) was studied recently by [51]. This baseline stacking algorithm requires that we treat the forecasting model given in Equation (2) as a base learner. Accordingly, we estimate 15 base learners—one for every agricultural commodity. Given the large number of parameters to be estimated, we estimate the base learners either by means of the Lasso estimator, as an elastic net, or by means of the Ridge regression estimator (see [52,53]), where we choose the corresponding shrinkage parameter using 10-fold cross validation (CV). We use the CV-based out-of-fold predictions from the base learners to construct a matrix ($\hat{H}^{CV}$) with 15 columns—one for every agricultural commodity. Finally, we construct a meta learner by estimating a regression model—one for every agricultural commodity—of RV_{t+h} on all predictors in $\hat{H}^{CV}$. Hence, the baseline stacking algorithm implies that the second-stage meta learners extract the information embedded in the predictors of the base learners in a way such that the forecast of RV_{t+h} combines the first-stage estimated effects on all RVs in a linear way. We use the shrinkage estimator that we apply to the base learners to estimate the meta learners. We use R add-on package "joinet" [51] to implement the baseline stacking algorithm.

In addition, we use a modified stacking algorithm that was proposed recently by [54]. Specifically, we use their residual stacking algorithm and corresponding R add-on package"MTPS". The modified stacking algorithm requires that we fit base learners in the first stage and compute the resulting fitted values of the RVs. One then models the residuals one obtains for agricultural commodity k using the fitted first-stage RVs (excluding the one for k) and obtains a meta learner using the first-stage base learner plus the fitted residual function. As a result, one can combine, for example, a first-stage Lasso estimator with a

another Lasso estimator or, in case one suspects that the data feature nonlinear patterns that are worthwhile studying, regression trees [55] to obtain a meta learner. We call the latter a Lasso-RF model because a regression tree represents a general (rather than a linear) residual function.

4. Empirical Results

4.1. Full-Sample Results

We start the discussion of our empirical results by eyeballing the heat maps we plot in Figure 3, which show the full-sample coefficients of the HAR-RV-S model for the four different forecasting horizons. The results are based on the Lasso version of the baseline stacking algorithm. The upper-left heat map shows that at the short forecasting horizon ($h = 1$), the coefficients of the classic HAR-RV model (that is, the diagonal cells of the map) dominate the scenery. The colors of most of the off-diagonal cells indicate that the spillover coefficients are close to zero and, in some cases, negative. The coefficients of the HAR-RV-S model somewhat gain in prominence as we move on to one of the intermediate forecast horizons ($h = 5, 10$) plotted in the upper-right and lower-left heat maps. While there are several positive off-diagonal coefficients, we also observe various negative estimated coefficients, especially when we consider the off-diagonal (RV_{LF}) coefficients in the upper part of the heat maps. Finally, at the long forecast horizon ($h = 20$), it appears that, while there are still some noticeable spillover effects, the own RV_{LF} coefficients gain somewhat in relative importance again (lower-right panel). Taken together, the estimated coefficients indicate that it may be possible to improve in-sample model fit by accounting for spillover effects.

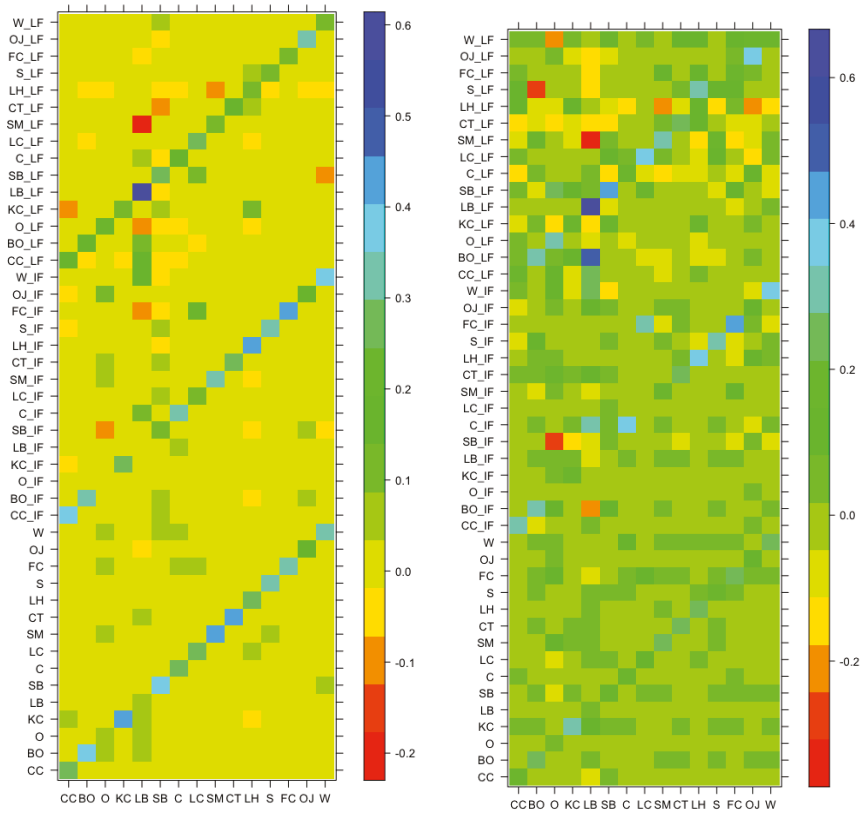


Figure 3. *Cont.*

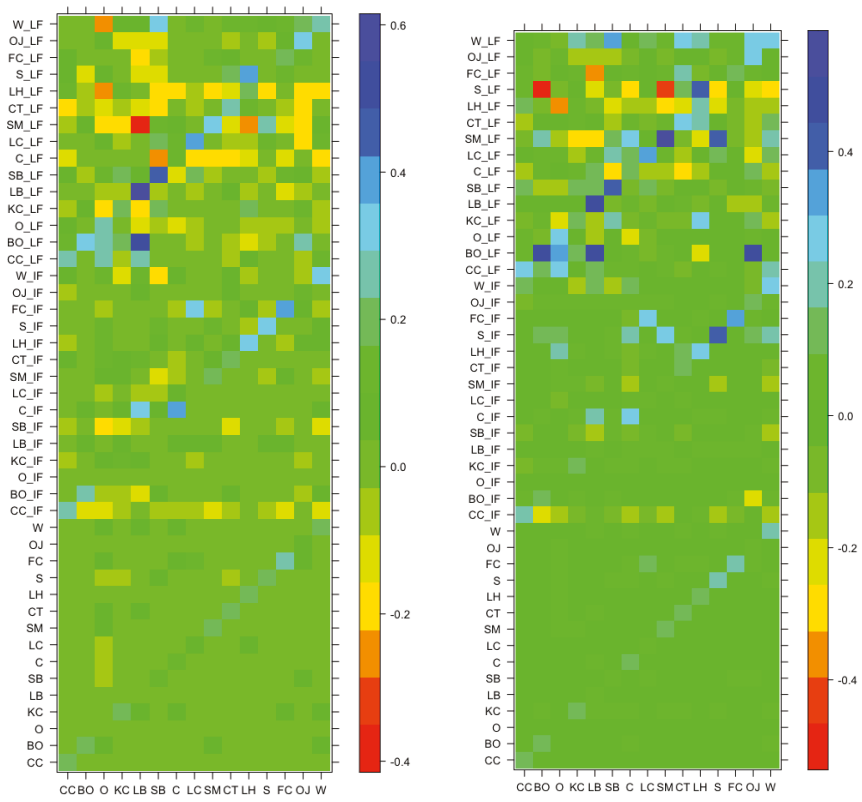


Figure 3. Full−sample estimated coefficients (baseline stacking algorithm). The forecast horizons are $h = 1, 5, 10, 20$ (starting in the upper-left panel).

The results we summarize in Figure 4 illustrate that this is, indeed, the case. Figure 4 plots in-sample ratios of the RMSFE for a comparison of the HAR-RV-S model with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the HAR-RV-S model produces a smaller in-sample RMSFE than the HAR-RV model. The results, irrespective of whether we study a Lasso estimator, an elastic net, or a Ridge regression estimator, indicate that the in-sample fit of the HAR-RV-S model relative to the classic HAR-RV model tends to improve as we increase the length of the forecast horizon.

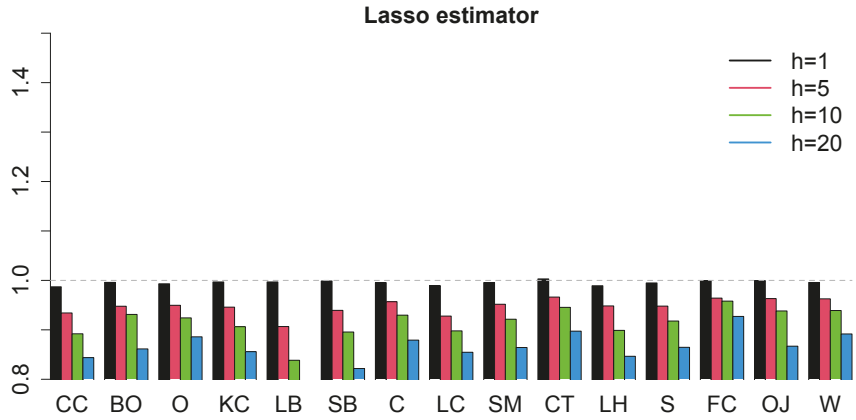


Figure 4. *Cont.*

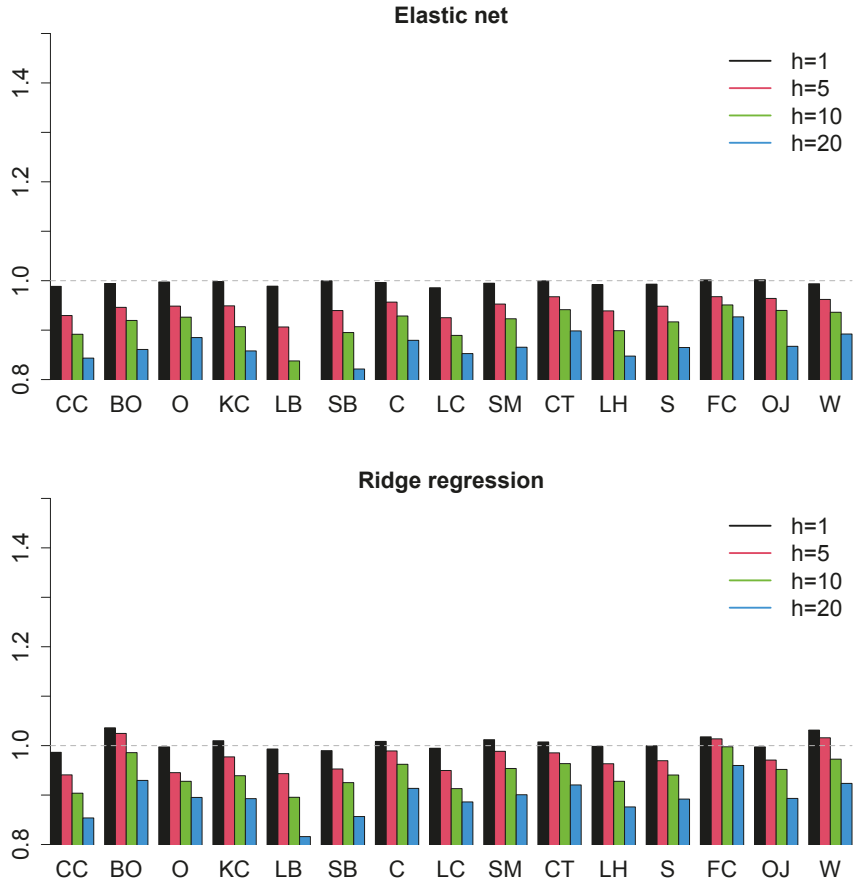


Figure 4. RMSFE ratios for the full sample (baseline stacking algorithm). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

4.2. Forecasting Results

In-sample fit does not necessarily carry over to an out-of-sample analysis. We start our comparison of the out-of-sample performance of the HAR-RV-S model, as estimated by the baseline stacking estimator, with that of the HAR-RV model, as estimated by the OLS technique. Figure 5 depicts the resulting RMSFE ratios that we obtain when we consider a recursive estimation window, while Figure 6 depicts the corresponding RMSFE ratios for a rolling estimation window. The key result for both types of estimation windows is that the classic HAR-RV model outperforms the HAR-RV-S model for the vast majority of commodities, especially when we increase the length of the forecasting horizon. This key result is not sensitive to the specific choice of the shrinkage estimator (Lasso estimator, elastic net, or ridge regression).

Figure 7 shows, for the example of a recursive estimation window, that we observe the superior performance of the HAR-RV model relative to the HAR-RV-S model when we also consider the MAFE as our metric of forecasting accuracy. The MAFE ratios should be less sensitive to large forecasting error in the wake of a sudden outburst of RV (see Figure 1) than the RMSFE ratio, but the results clearly demonstrate that our key result is robust to changes in the metric of forecast accuracy.

Figure 8 (for a recursive estimation window) and Figure 9 (for a rolling estimation window) summarize the results we obtain when we study the modified stacking algorithm. For the modified stacking algorithm, we consider the following four alternative combinations of estimators: a Lasso–Lasso estimator, a Lasso-RF (that is, a general regression tree-based residual function) estimator, a ridge–ridge estimator, and a ridge-RF estimator. Four all

four combinations of estimators, we use the RMSFE ratio to quantify relative forecasting performance. Across the four combinations of estimators, we observe that the HAR-RV-S model does not outperform the classic HAR-RV model. Quite to the contrary, the HAR-RV model exhibits a robustly superior performance, especially as the length of the forecasting horizon increases.

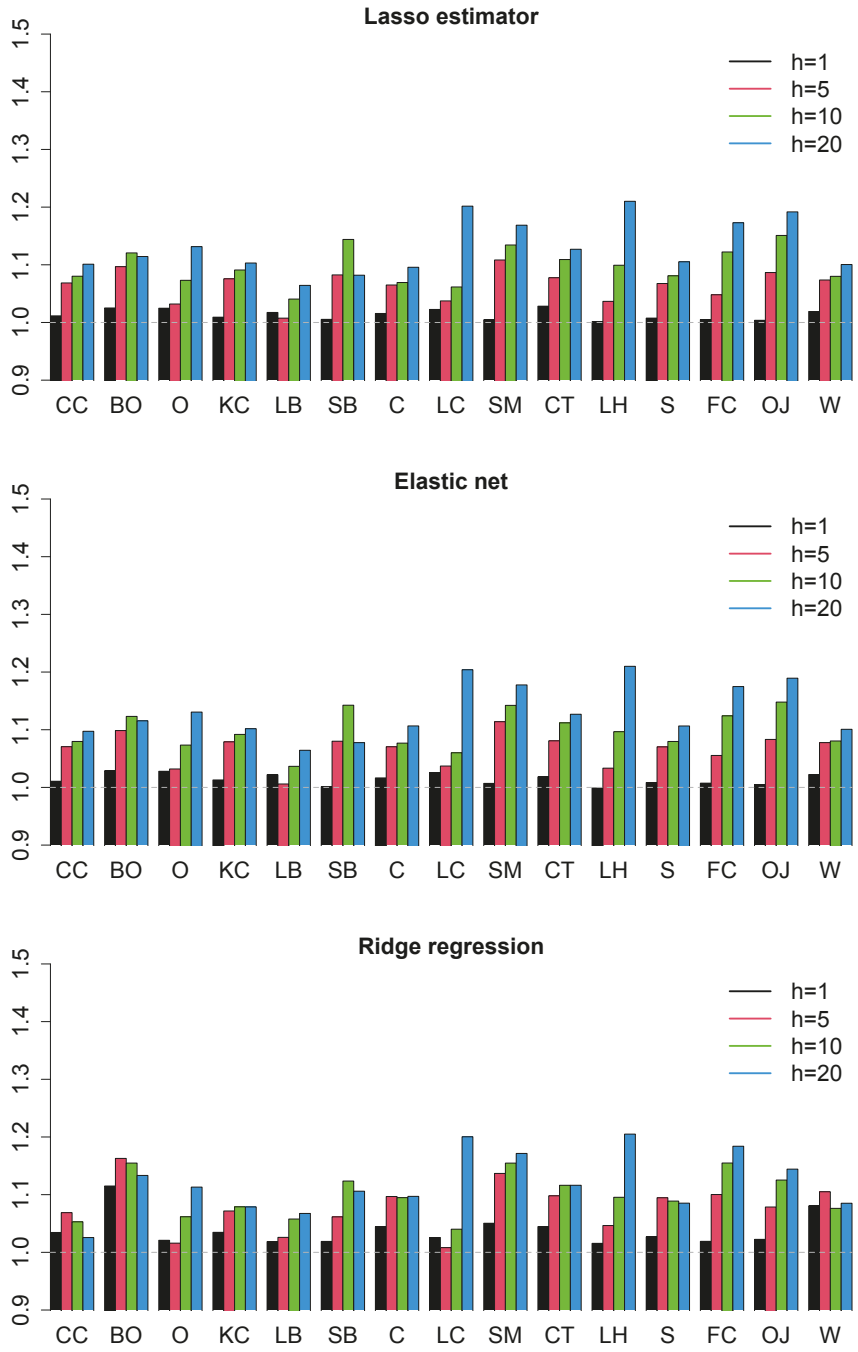


Figure 5. RMSFE ratios for a recursive window (baseline stacking algorithm). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

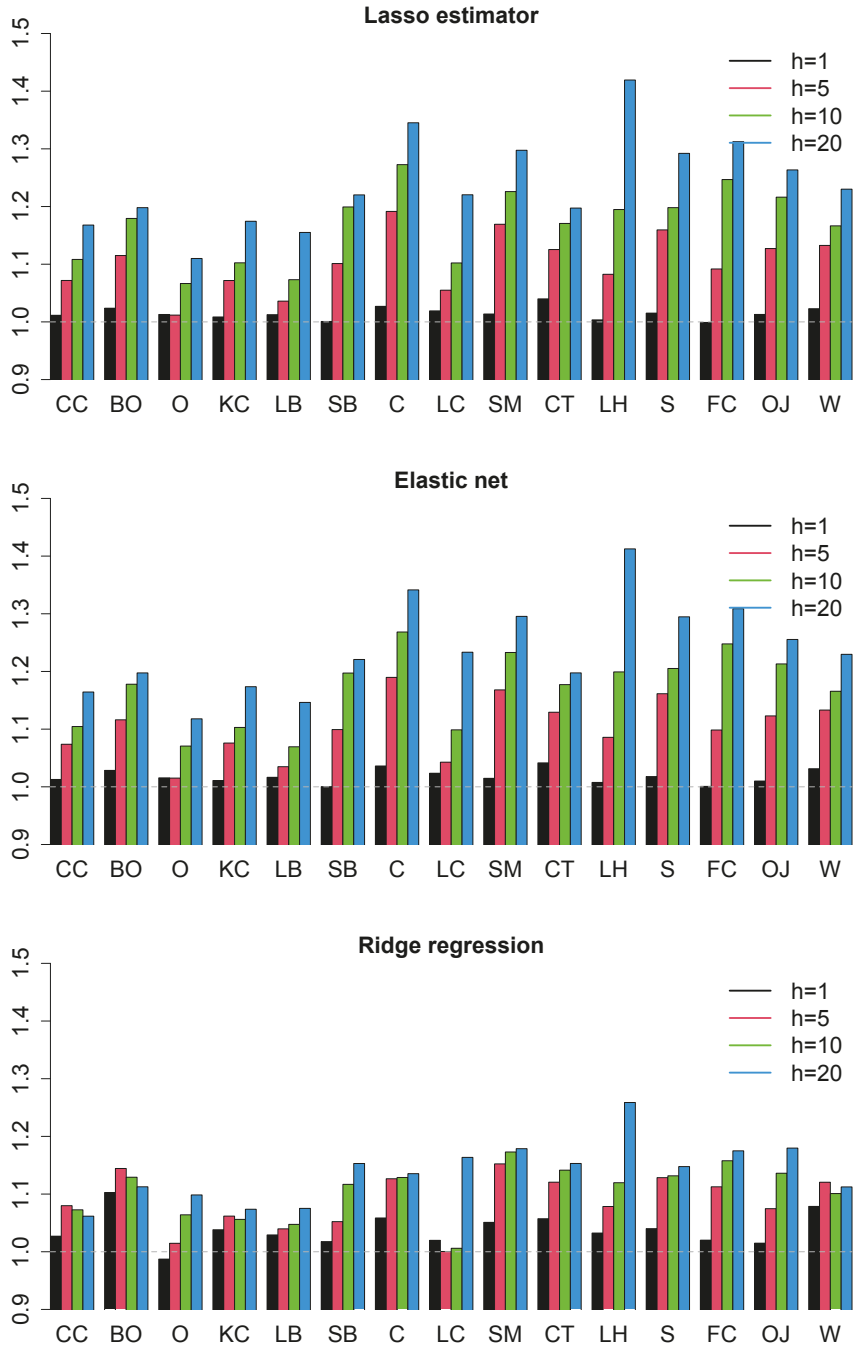


Figure 6. RMSFE ratios for a rolling window (baseline stacking algorithm). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

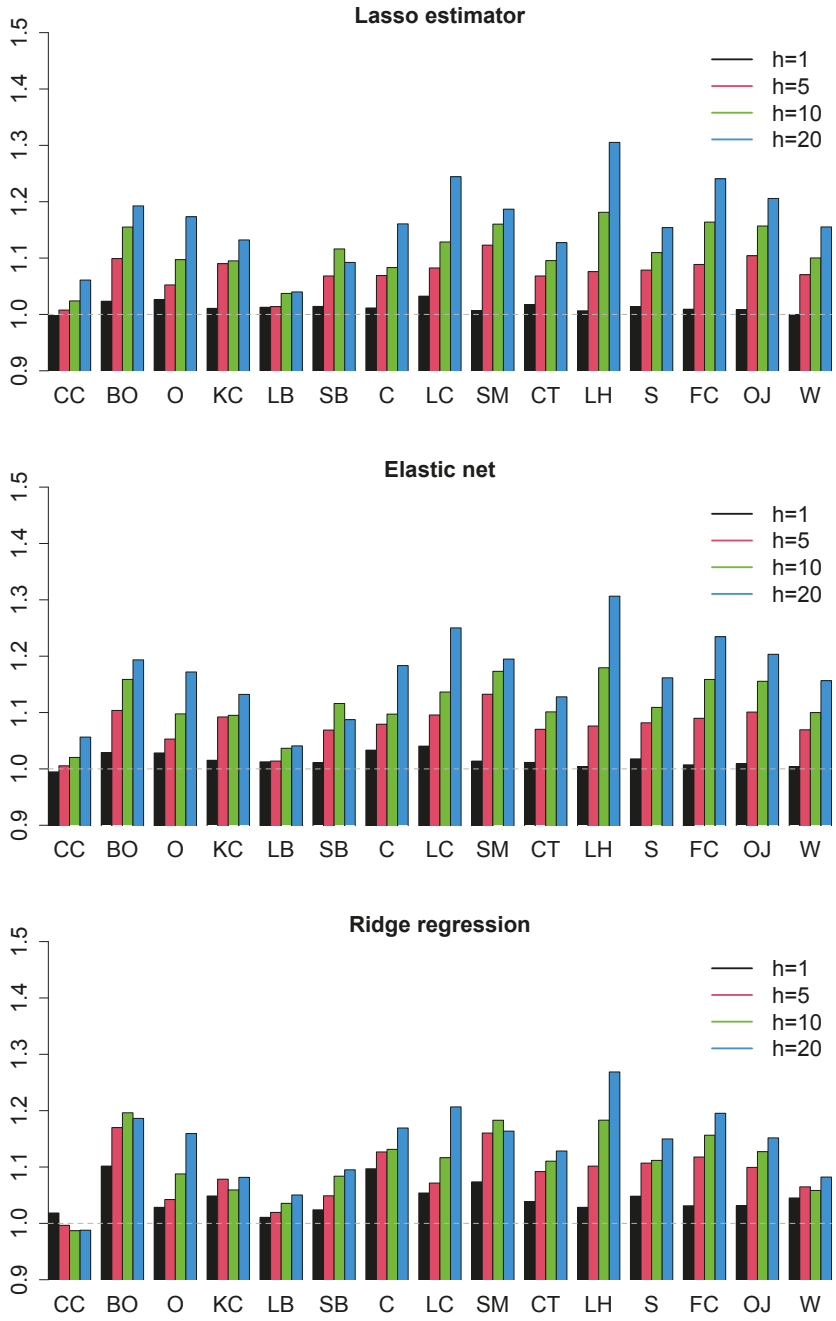


Figure 7. MAFE ratios for a recursive window (baseline stacking algorithm). MAFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. A MAFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample MAFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

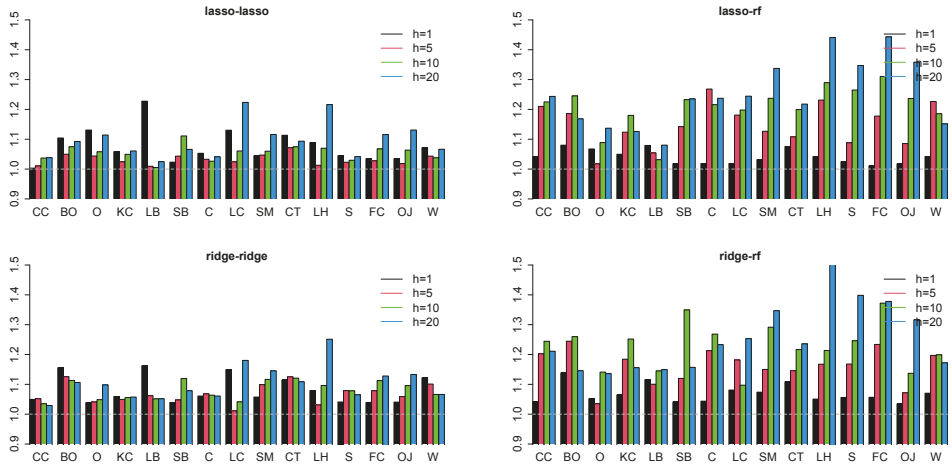


Figure 8. RMSFE ratios for a recursive window (modified stacking algorithm). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

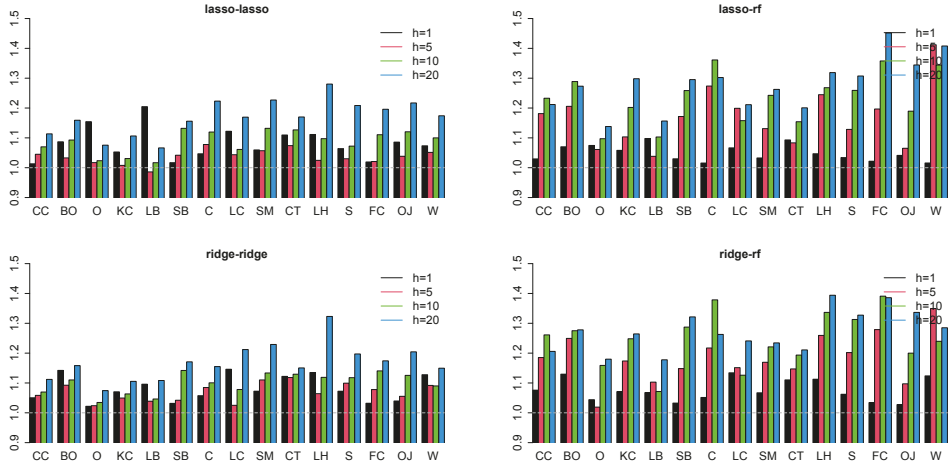


Figure 9. RMSFE ratios for a rolling window (modified stacking algorithm). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

4.3. Robustness Checks

As our first robustness check, in Figure 10 (recursive estimation window) and Figure 11 (rolling estimation window) we summarize the results for multivariate Lasso, multivariate elastic net, and multivariate ridge regression estimators. For estimation of the multivariate shrinkage estimators, we use R add-on package "glmnet" [56]. For the short forecast horizon, we observe a few cases for which the HAR-RV-S model performs better than the classic HAR-RV model, but the general message conveyed by the results is in line with the results for the stacking algorithms. The classic HAR-RV model performs well for the majority of agricultural commodities at the short forecast horizon, and it performs better than the HAR-RV-S model at the intermediate and long forecast horizons.

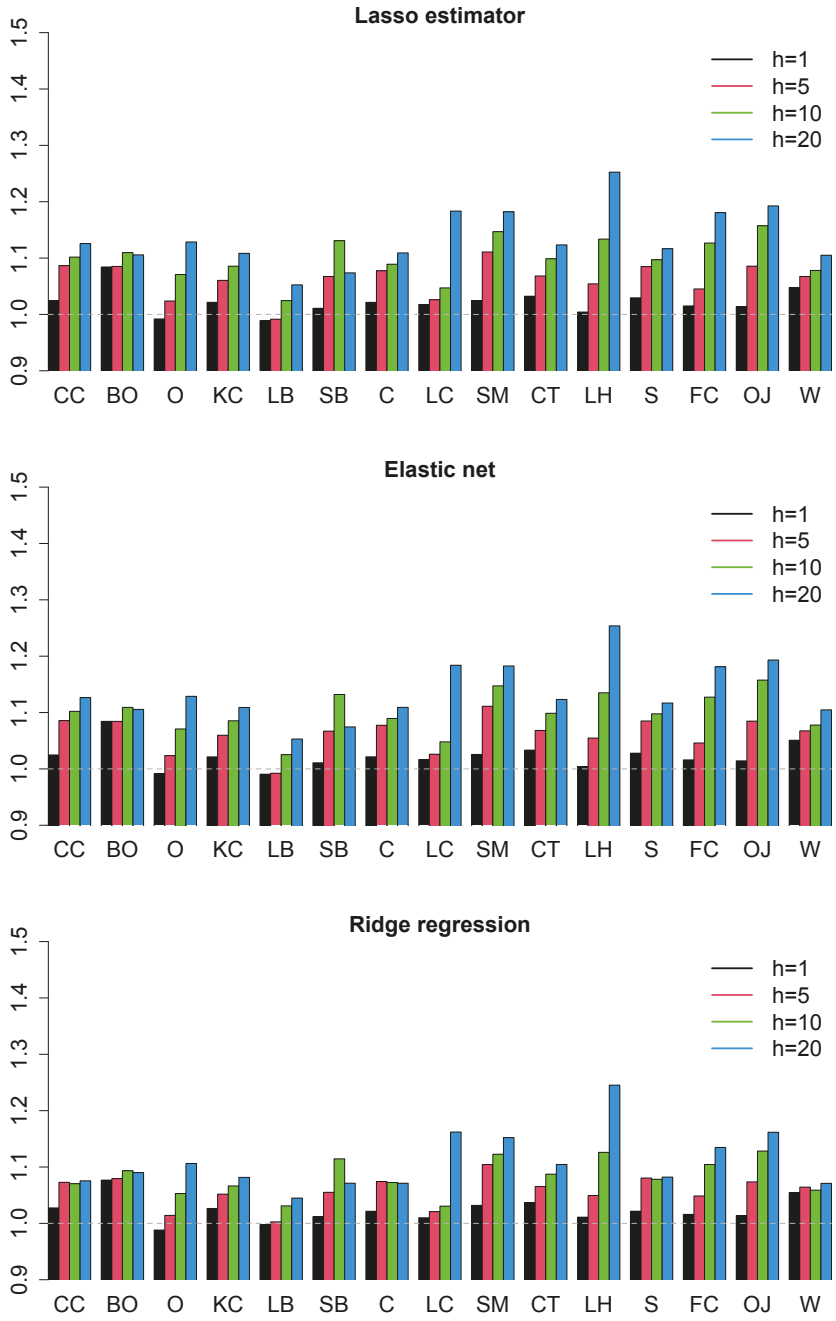


Figure 10. RMSFE ratios for a recursive window (multivariate shrinkage estimator). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

As another robustness check, we study the relative performance of the HAR-RV-S model along the quantiles of the distribution of the actual realizations of RV during the out-of-sample period. We plot the results in Figure 12 (recursivce estimation window) and Figure 13 (rolling estimation window), where we focus on the baseline stacking algorithm for the sake of space. We observe that the HAR-RV-S model slightly outperforms the HAR-RV model at the short forecast horizon for intermediate quantiles close to the median, mainly when we consider a Lasso estimator. For the intermediate and long forecast horizons, in contrast, the HAR-RV model clearly outperforms the HAR-RV-S model. The relative performance of the HAR-RV-S model only starts improving at some of

the very upper quantiles, but this improvement is not strong enough to beat the forecasting performance of the HAR-RV model in a robust way.

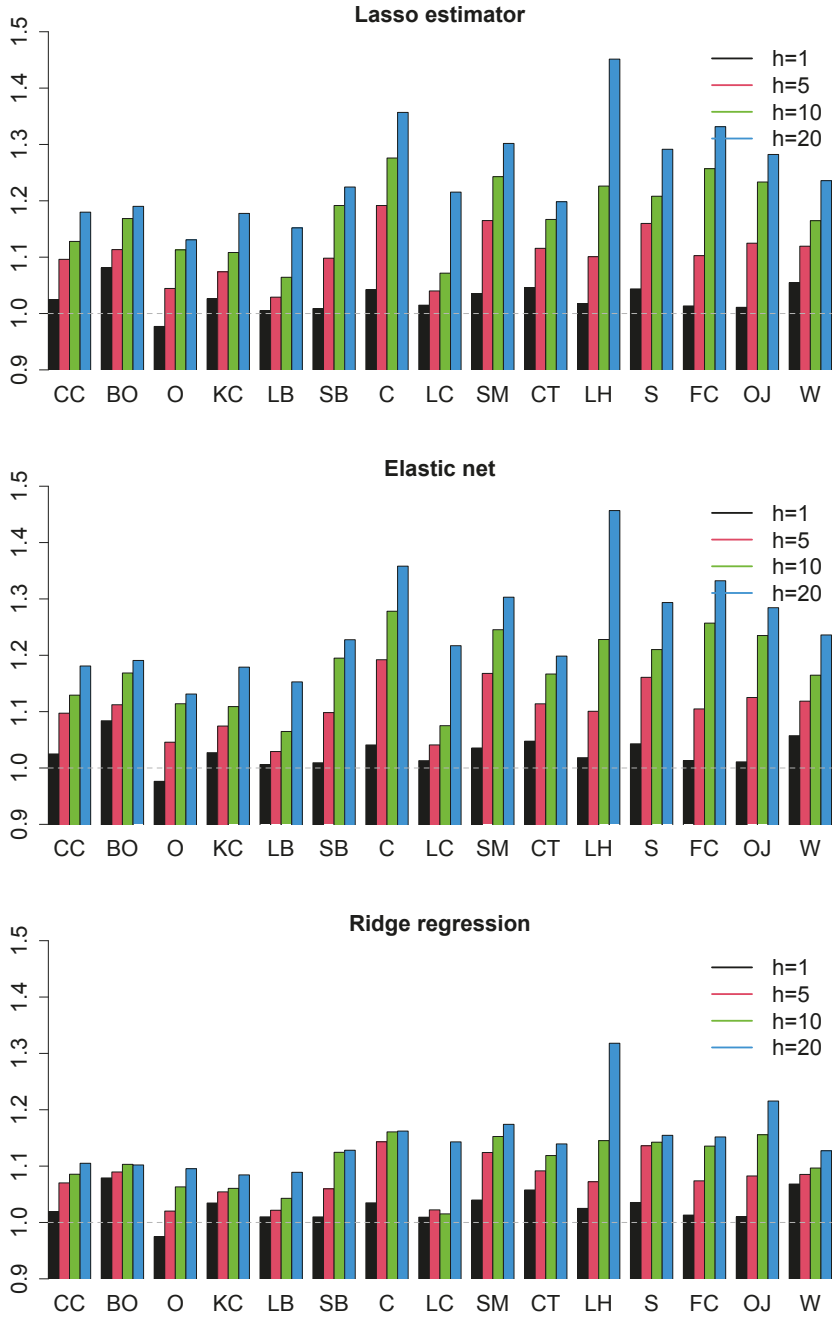


Figure 11. RMSFE ratios for a rolling window (multivariate shrinkage estimator). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

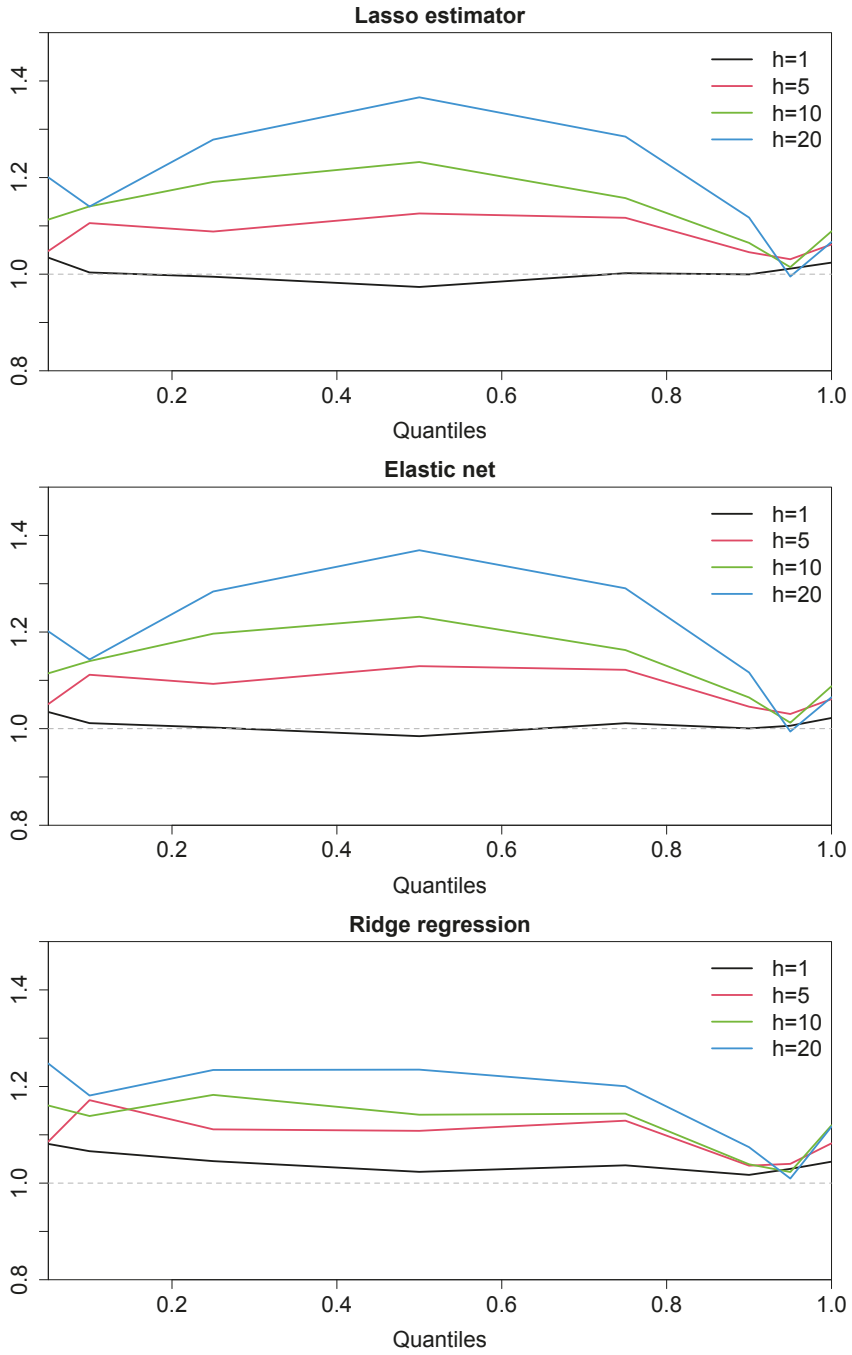


Figure 12. Quantile-based RMSFE ratios for a recursive window (baseline stacking algorithm). RMSFE ratios for different quantiles of the realizations of RV. RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

As yet another robustness check, we consider the possibility that the strength of spillover effects between the RVs varies over time. If so, the performance of the HAR-RV-S model relative to that of the classic HAR-RV model may have undergone corresponding changes over time. In order to study this question in some more detail, in Figure 14, we plot rolling-window estimates of the Diebold–Yilmaz [57,58] total dynamic spillover index (implemented using the add-on package "Spillover"; see [25,59] for a discussion of the link between the multivariate HAR-RV model and the Diebold–Yilmaz index). The estimation results clearly reveal that the strength of the total spillover effects among the RVs of the

agricultural commodities in our sample has increased over time. This increase in the strength of the total spillover effects warrants a closer inspection of the relative forecasting performance of the HAR-RV-S and HAR-RV models during subsample periods.

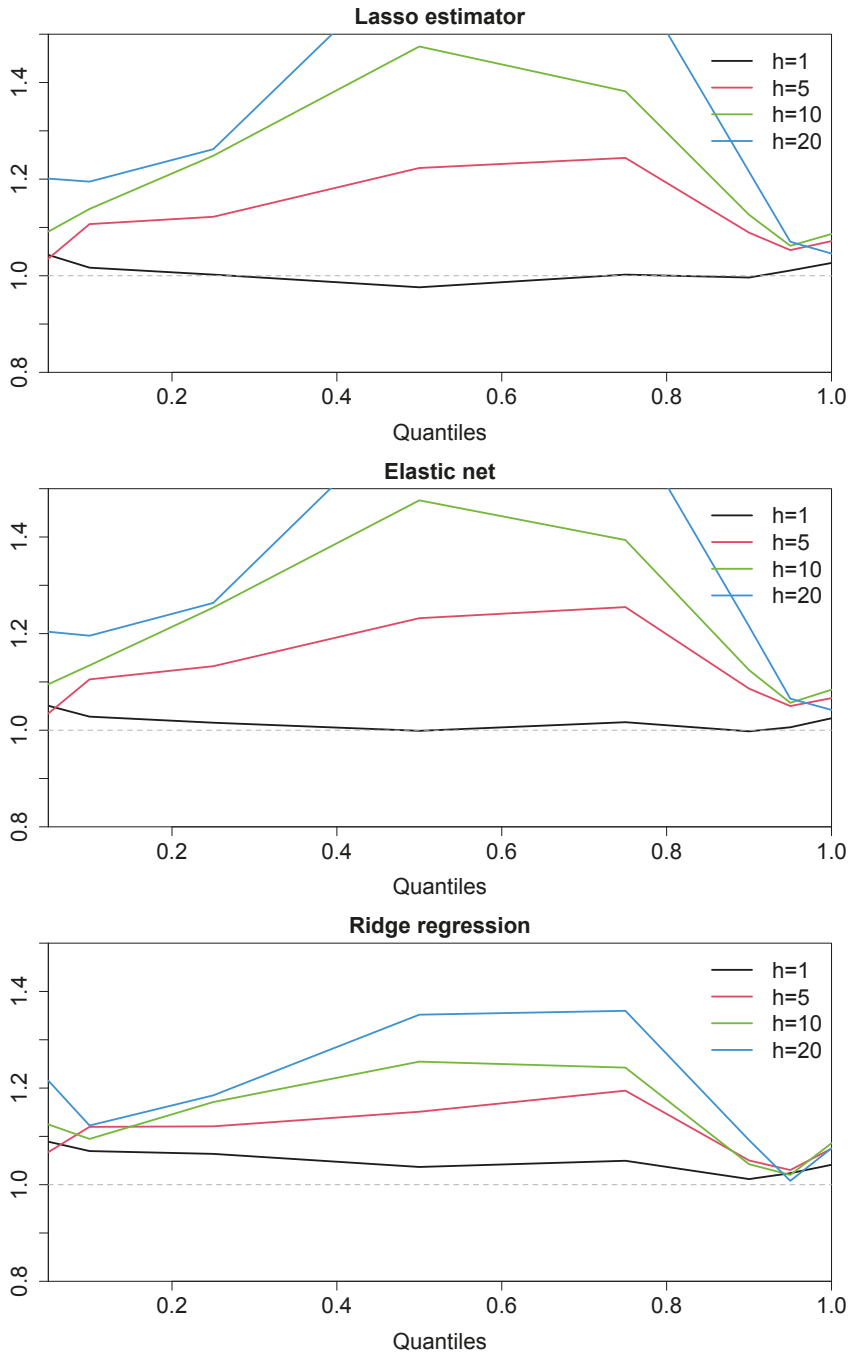


Figure 13. Quantile-based RMSFE ratios for a rolling window (baseline stacking algorithm). RMSFE ratios for different quantiles of the realizations of RV. RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

While providing a detailed examination of the sources of the increase in the strength of the total spillover effects is beyond the scope of this research, a plausible economic conjecture is that factors like increased geopolitical tensions and war attention, as well as climate policy uncertainty, are factors that have contributed to the increase in the strength

of the total spillover effects. These global factors are likely to have had an impact on the agricultural commodities in our sample, although perhaps different in magnitude and timing. Furthermore, the trends towards a deeper financialization of commodity markets may have strengthened the importance of behavioral aspects like investor sentiment and investor panic for the total spillover effects. With respecto the roles played by war attention, climate risks, and investor panic for financial markets, as well as financial and economic stability, see the recent research by [59–61]. Also see the remarks on the relevant literature in the introductory section.

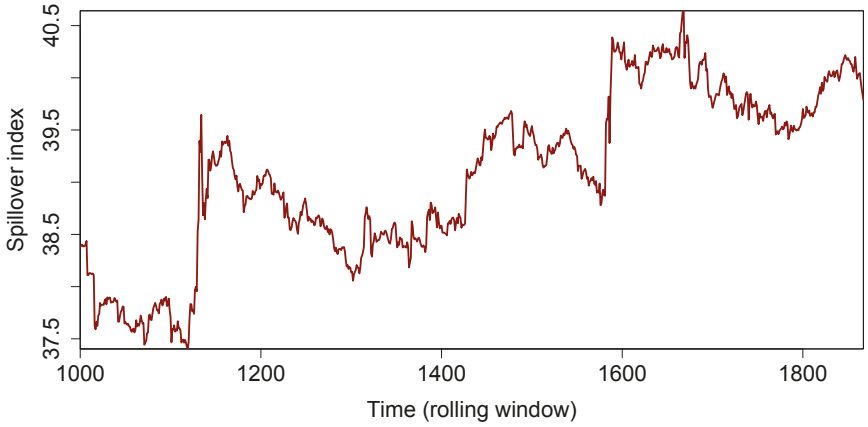


Figure 14. Rolling-window estimates of a spillover index. The total dynamic spillover index is derived from a VAR(5) model estimated using a rolling estimation window with a length of 1000 observations and a 10-step-ahead generalized forecast error variance decomposition.

We summarize the results of such a subsample analysis in Figure 15 (recursive estimation window) and Figure 16 (rolling estimation window), where we use the first 450 out-of-sample forecasts to define the first subsample period and the remaining forecasts to define the second subsample period (the exact number of out-of-sample forecasts depends on the forecast horizon). While we find the superior performance of the HAR-RV-S model in terms of the RMSFE ratio for some combinations of agricultural commodities and forecast horizons, the general picture that emerges from the analysis of the subsamples is that the relative forecasting performance of the HAR-RV-S model is not systematically better in the second than in the first subsample. There are a few exceptions to this general picture. For example, in some model configurations, the HAR-RV-S model outperform the HAR-RV model in the second but not in the first subsample when we consider the RVs of CC and KC, but not at all forecast horizons and not for all four combinations of estimators. Moreover, as in the case of the full-sample analysis, the relative performance of the HAR-RV model, in general, strengthens with the length of the forecast horizon. It is also interesting to observe that in the first subsample, the HAR-RV-S-model outperforms the HAR-RV model for C and S (in the case of the latter, only for the rolling estimation window) when we use the Lasso–Lasso and the ridge–ridge estimators, where the performance of the HAR-RV-S model tends to strengthen with the length of the forecast horizon. Thus, our results for the first subsample partially overlap (that is, for C and S) with the results reported by [8], who reported results for the sample period of 2013–2018 (using Chinese data, so it is clear that the results are not directly comparable).

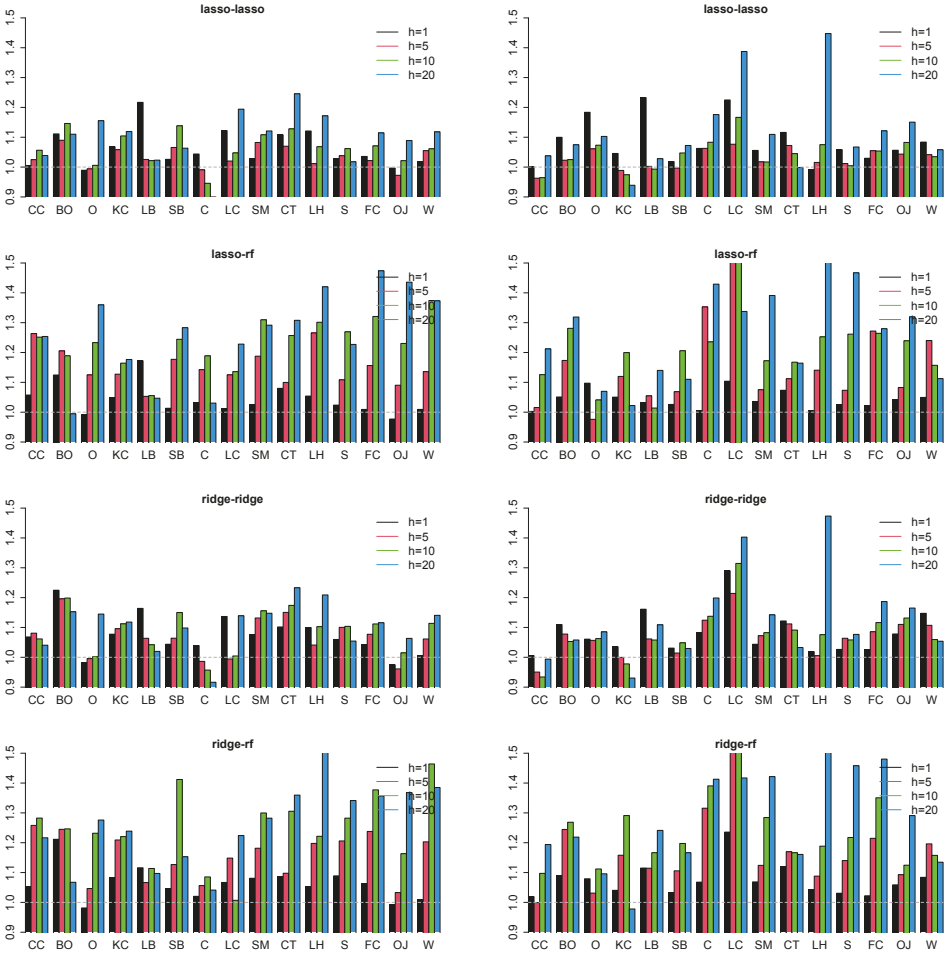


Figure 15. Subsample analysis for a recursive window (modified stacking algorithm). The panels on the left-hand side summarize the results for the first subsample. The panels on the right-hand side summarize the results for the first subsample. The first subsample comprises the first 450 out-of-sample forecasts. The second subsample is obtained upon deleting the first 450 out-of-sample forecasts. RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

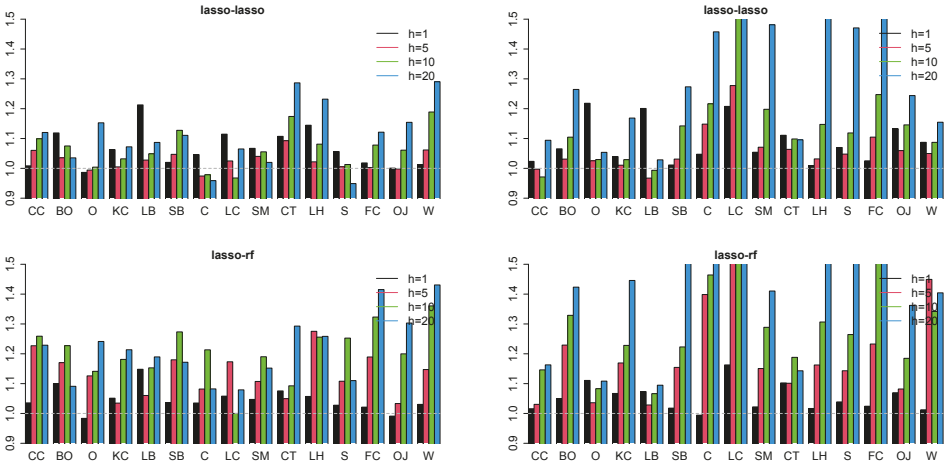


Figure 16. *Cont.*

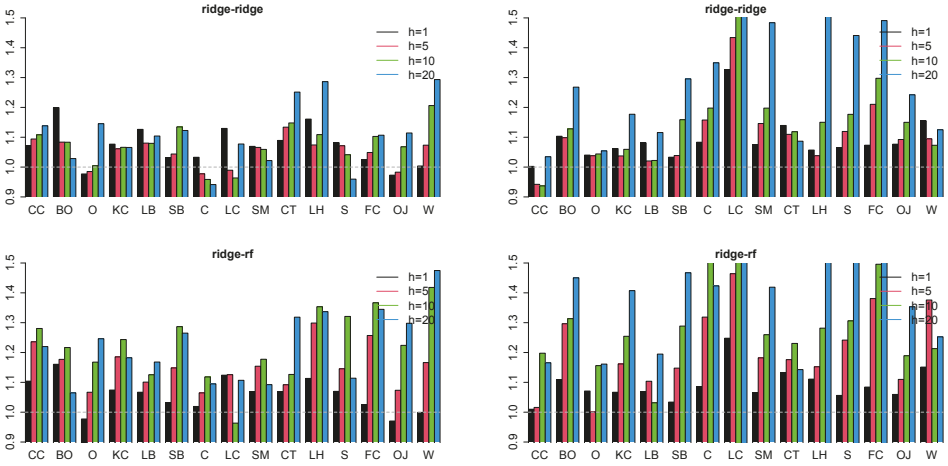


Figure 16. Subsample analysis for a rolling window (modified stacking algorithm). The panels on the left-hand side summarize the results for the first subsample. The panels on the right-hand side summarize the results for the first subsample. The first subsample comprises the first 450 out-of-sample forecasts. The second subsample is obtained upon deleting the first 450 out-of-sample forecasts. RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

4.4. Extension to Energy Commodities and Precious Metals

It is interesting to study whether the out-of-sample results obtained when we extend our HAR-RV-S model for agricultural commodities to include other important commodities. In extending the model in this way, we can account for the potential impact of spillover effects across different classes of commodities, as noted in a number of studies involving the agriculture, energy, and precious metals markets, (see, for example, refs. [62–71]) on out-of-sample forecasting performance. In order to extend the HAR-RV-S model in this way, we consider the RVs of several energy commodities (natural gas (NG), heating oil (HO), and coal (CL)) and precious metals (gold (GC), copper (HG), palladium (PA), platinum (PL), and silver (SI)). The data source is the same as that described in detail in Section 2, so we can directly match the RVs of the agricultural commodities with those of the energy commodities and the precious metals by date. We plot the RVs of the energy commodities and the precious metals at the end of the paper (Appendix A; Figures A1 and A2), where we also report the full-sample correlation matrix for the members of the three commodity groups (Figure A3). We also plot the corresponding total dynamic spillover index (Figure A4), which shows that the spillover effects among the three members of the three commodity groups increased towards the end of the sample period. As in the agricultural-commodities-only model, the spillover effects are also visible in the full-sample RMSFE ratios (based on the modified stacking algorithm; Figure A5). The RMSFE ratios clearly decrease with the length of the forecast horizon, especially when we combine the shrinkage estimators with regression trees. Finally, the results that we report in Figure A6 (for a recursive estimation window) and in Figure A7 (for a rolling estimation window) corroborate the main finding of our empirical research that the HAR-RV-S model—with only a few exceptions—does not outperform out-of-sample the forecasting performance of the classic HAR-RV model, especially as the length of the forecasting horizon increases.

5. Concluding Remarks

Modeling and forecasting realized volatilities of financial asset prices, in general, and of commodity price fluctuations, in particular, is of key importance for financial market participants and policymakers. Financial market participants rely on accurate forecasts of realized volatilities when solving portfolio optimization problems and pricing derivative securities. Policymakers, in turn, need accurate forecasts of realized volatilities when

designing policies to mitigate the potential adverse effects of a rise in economic and—in case of agricultural commodities—perhaps even political uncertainty associated with sudden increases in the volatility of price fluctuations. A natural and important research question, therefore, is whether forecasts of the realized volatilities of commodity price fluctuations can benefit when a forecaster takes into account spillover effects across the realized volatilities of agricultural commodities. The results we report in this research clearly demonstrate that such spillover effects exist, that they can be strong, that they may vary over time, and that accounting for spillover effects by means of a simple HAR-RV-S model has beneficial effects in an in-sample analysis. However, we do not observe systematic out-of-sample forecasting gains relative to a classic HAR-RV model.

In order to obtain out-of-sample forecasts of the RVs of 15 agricultural commodities (and, in an extended model, three energy commodities and five precious metals), we use various multi-task stacking algorithms, as well as a multivariate shrinkage estimator. The multi-task stacking algorithms, in particular, have the advantage of being straightforward to implement in high-dimensional multi-task forecasting problems. Modeling and forecasting the realized volatilities of the various agricultural commodities that we studied in this research can be interpreted as belonging to this class of problems. While the multivariate shrinkage estimator retains a simple linear structure of the forecasting model, the multi-task stacking algorithm opens up the possibility of combining different base and meta learners, where for the latter, we also have use regression trees so as to explore potential nonlinear structures in the data. Irrespective of the algorithm or combination of base and meta learners that we studied, we obtained the same main finding that spillover effects do not leverage out-of-sample forecast accuracy relative to the classic HAR-RV model. Our main finding implies that the research strategy used by some researchers in recent papers (see, for example, refs. [18–20]) to forecast the RVs of agricultural commodities in an univariate modeling approach is likely to be a good starting point for further analysis and can be also considered beneficial from the perspective of investors looking for optimal portfolio allocations and policymakers aiming to stabilize food prices.

This does not mean that a multivariate modeling approach cannot yield important additional and novel insights. In fact, in future research, it would be interesting to study whether other algorithms developed in the large and rapidly growing machine learning literature corroborate our main finding or whether the application of other algorithms brings to the forefront features of the data that the algorithms and estimators we applied in our research fail to detect. In technical terms, it is also interesting to explore how the stacking algorithms we studied in this research can be combined with the type of multivariate HAR-RV cum GARCH models discussed in related earlier literature. Such an extension would also render it possible to more directly compare the results we report in this paper with the results that [8] reported in their recent empirical study of a small set of agricultural commodities (with a shorter sample period and using Chinese data). Furthermore, against the background of the much discussed financialization of commodity markets, it is worthwhile to investigate whether accounting for spillover effects across the realized volatilities of different asset classes (for example, agricultural commodities and stock markets) yields insights that can help to improve the accuracy of out-of-sample forecasts of realized volatilities (as in, for example, ref. [26]).

Author Contributions: Conceptualization, R.G.; Methodology, R.G. and C.P.; Software, C.P.; Formal analysis, C.P.; Data curation, R.G.; Writing—original draft, R.G. and C.P.; Writing—review and editing, R.G. and C.P. All authors have read and agreed to the published version of the manuscript.

Funding: The authors declare that they did not receive any funding for this research.

Data Availability Statement: The data used to derive the results documented in this research are available from the authors upon reasonable request.

Acknowledgments: We would like to thank two anonymous referees for many helpful comments. The usual disclaimer applies.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

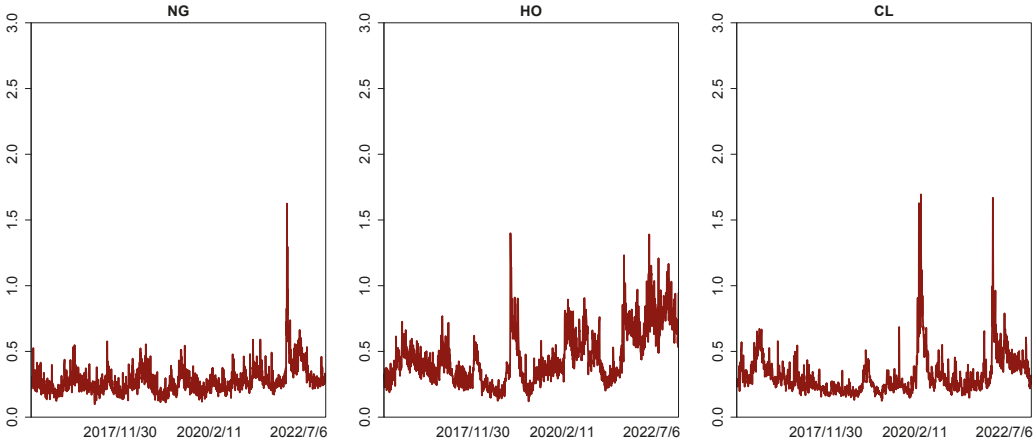


Figure A1. RVs of energy commodities.

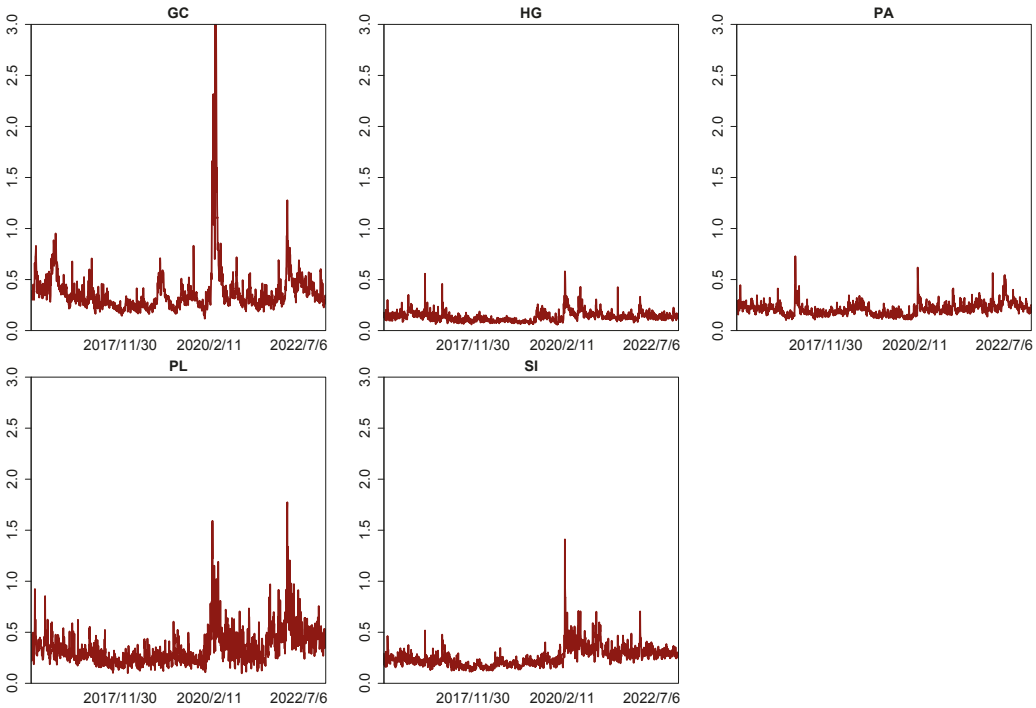


Figure A2. RVs of precious metals.

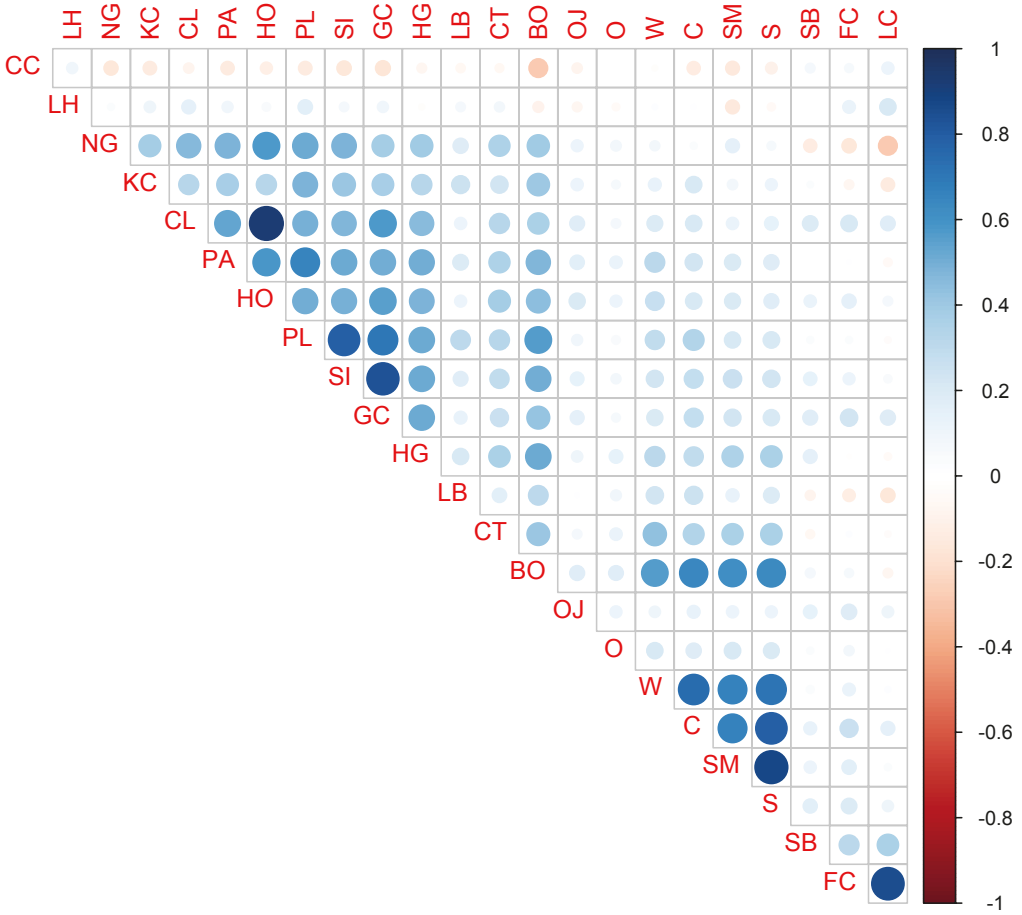


Figure A3. Full−sample correlation matrix (extended sample of commodities).

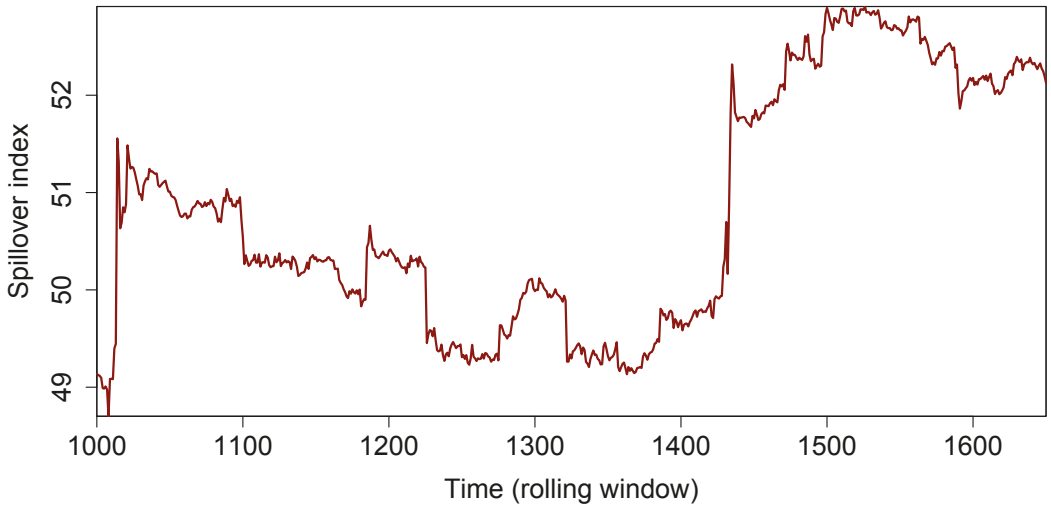


Figure A4. Rolling-window estimates of a spillover index (extended sample of commodities). The total dynamic spillover index is derived from a VAR(5) model estimated using a rolling estimation window with a length of 1000 observations and a 10-step-ahead generalized forecast error variance decomposition.

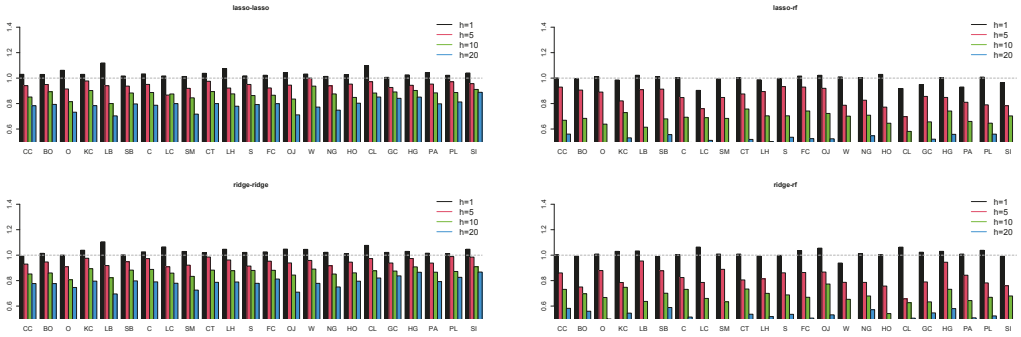


Figure A5. Full-sample RMSFE ratios (modified stacking estimator). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

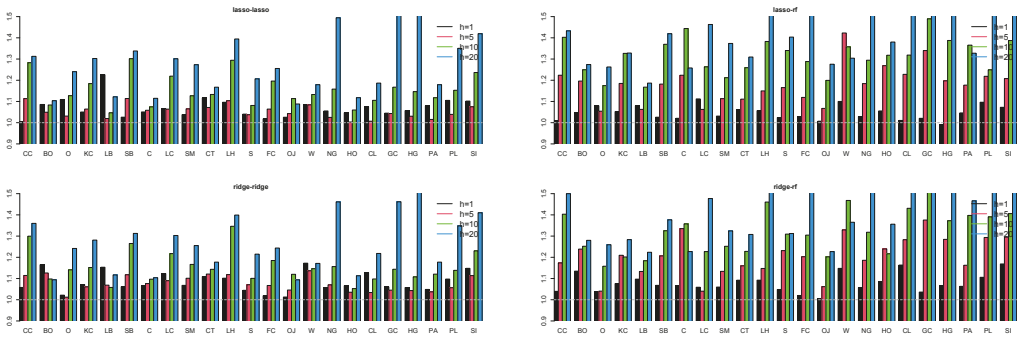


Figure A6. RMSFE ratios for a recursive window (modified stacking estimator). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

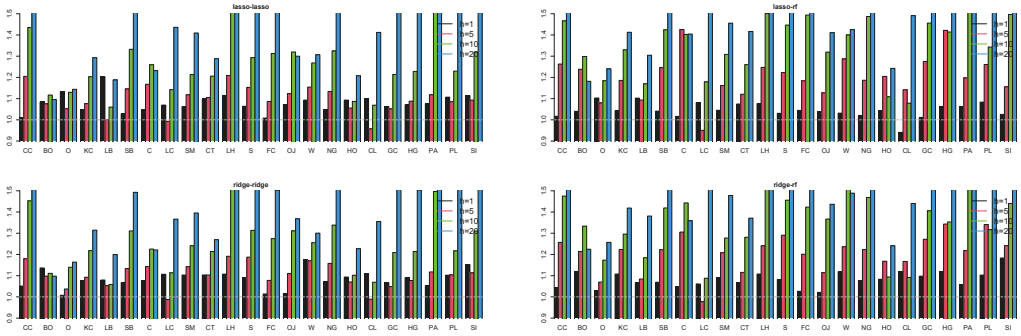


Figure A7. RMSFE ratios for a rolling window (modified stacking estimator). RMSFE ratios for a comparison of the HAR-RV-S model (meta learner) with the HAR-RV model. An RMSFE ratio smaller than unity indicates that the meta learner produces a smaller in-sample RMSFE than the HAR-RV model. The forecast horizons are $h = 1, 5, 10, 20$.

References

1. Trujillo-Barrera, A.; Mallory, M.; Garcia, P. Volatility spillovers in U.S. crude oil, ethanol, and corn futures markets. *J. Agric. Resour. Econ.* **2012**, *37*, 247–262.
2. Beckmann, J.; Czudaj, R. Volatility transmission in agricultural futures markets. *Econ. Model.* **2014**, *36*, 541–546. [CrossRef]
3. Gardebroek, C.; Hernandez, M.; Robles, M. Market interdependence and volatility transmission among major crops. *Agric. Econ.* **2016**, *47*, 141–155. [CrossRef]
4. Bonato, M. Realized correlations, betas and volatility spillover in the agricultural commodity market: What has changed? *J. Int. Financ. Mark. Inst. Money* **2019**, *62*, 184–202. [CrossRef]
5. Luo, J.; Chen, L. Multivariate realized volatility forecasts of agricultural commodity futures. *J. Futures Mark.* **2019**, *39*, 1565–1586. [CrossRef]

6. Cagli, E.C.; Mandaci, P.E.; Taskin, D. The volatility connectedness between agricultural commodity and agri businesses: Evidence from time-varying extended joint approach. *Financ. Res. Lett.* **2023**, *52*, 103555. [CrossRef]
7. Luo, J.; Marfatia, H.A.; Ji, Q.; Klein, T. Co-volatility and asymmetric transmission of risks between the global oil and China's futures markets. *Energy Econ.* **2023**, *117*, 106466. [CrossRef]
8. Marfatia, H.A.; Luo, J.; Ji, Q. Forecasting the volatility of agricultural commodity futures: The role of co-volatility and oil volatility. *J. Forecast.* **2022**, *41*, 383–404. [CrossRef]
9. Kyriazi, F.; Thomakos, D.D.; Guerard, J.B. Adaptive learning forecasting, with applications in forecasting agricultural prices. *Int. J. Forecast.* **2019**, *35*, 1356–1369. [CrossRef]
10. Karali, B.; Irwin, S.H.; Isengildina-Massa, O. Supply fundamentals and grain futures price movements. *Am. J. Agric. Econ.* **2020**, *102*, 548–568. [CrossRef]
11. Kwas, M.; Paccagnini, A.; Rubaszek, M. Common factors and the dynamics of cereal prices. A forecasting perspective. *J. Commod. Mark.* **2022**, *28*, 100240. [CrossRef]
12. Andersen, T.G.; Bollerslev, T. Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *Int. Econ. Rev.* **1998**, *39*, 885–905. [CrossRef]
13. McAleer, M.; Medeiros, M.C. Realized volatility: A review. *Econ. Rev.* **2008**, *27*, 10–45. [CrossRef]
14. Chatziantoniou, I.; Degiannakis, S.; Filis, G.; Lloyd, T. Oil price volatility is effective in predicting food price volatility. Or is it? *Energy J.* **2021**, *42*, 25–48. [CrossRef]
15. Degiannakis, S.; Filis, G.; Klein, T.; Walther, T. Forecasting realized volatility of agricultural commodities. *Int. J. Forecast.* **2022**, *38*, 74–96. [CrossRef]
16. Luo, J.; Klein, T.; Ji, Q.; Hou, C. Forecasting realized volatility of agricultural commodity futures with infinite hidden Markov HAR models. *Int. J. Forecast.* **2022**, *38*, 51–73. [CrossRef]
17. Corsi, F. A simple approximate long-memory model of realized volatility. *J. Financ. Econ.* **2009**, *7*, 174–196. [CrossRef]
18. Bonato, M.; Cepni, O.; Gupta, R.; Pierdzioch, C. El Niño, La Niñna, and forecastability of the realized variance of agricultural commodity prices: Evidence from a machine learning approach. *J. Forecast.* **2022**, *42*, 785–801. [CrossRef]
19. Bonato, M.; Cepni, O.; Gupta, R.; Pierdzioch, C. Forecasting the realized volatility of agricultural commodity prices: Does sentiment matter? *J. Forecast.* **2024**, *43*, 2088–2125. [CrossRef]
20. Bonato, M.; Cepni, O.; Gupta, R.; Pierdzioch, C. Financial stress and realized volatility: The case of agricultural commodities. *Res. Int. Bus. Financ.* **2024**, *71*, 102442. [CrossRef]
21. Gil-Alana, L.A.; Cunado, J.; de Gracia, F.P. Persistence, long memory, and unit roots in commodity prices. *Can. J. Agric. Econ.* **2012**, *60*, 451–468. [CrossRef]
22. Živkov, D.; Njegić, J.; Pexcxanac, M. Multiscale interdependence between the major agricultural commodities. *Agric. Econ.* **2019**, *65*, 82–92. [CrossRef]
23. Živkov, D.; Duraškovič, J.; Gajixcx-Glamoxcxlija, M. Multiscale downside risk interdependence between the major agricultural commodities. *Agribusiness* **2022**, *38*, 990–1011. [CrossRef]
24. Müller, U.A.; Dacorogna, M.M.; Davé, R.D.; Olsen, R.B.; Pictet, O.V. Volatilities of different time resolutions—Analyzing the dynamics of market components. *J. Empir. Financ.* **1997**, *4*, 213–239. [CrossRef]
25. Bubák, V.; Kočenda, E.; Žikeš, F. Volatility transmission in emerging European foreign exchange markets. *J. Bank. Financ.* **2011**, *35*, 2829–2841. [CrossRef]
26. De Nicola, F.; De Pace, H. Co-movement of major energy, agricultural, and food commodity price returns: A time-series assessment. *Energy Econ.* **2016**, *57*, 28–41. [CrossRef]
27. Luo, J.; Chen, L. Modeling and forecasting the multivariate realized volatility of financial markets with time-varying sparsity. *Emerg. Mark. Financ. Trade* **2020**, *56*, 392–408. [CrossRef]
28. Luo, J.; Chen, L. Realized volatility forecast with the Bayesian random compressed multivariate HAR model. *Int. J. Forecast.* **2020**, *36*, 781–799. [CrossRef]
29. Asai, M.; McAleer, M. The impact of jumps and leverage in forecasting co-volatility. *Econ. Rev.* **2017**, *36*, 638–650. [CrossRef]
30. Čech, F.; Baruník, J. On the modelling and forecasting of multivariate realized volatility: Generalized Heterogeneous Autoregressive (GHAR) Model. *J. Forecast.* **2017**, *36*, 181–206. [CrossRef]
31. Asai, M.; Gupta, R.; McAleer, M. The Impact of jumps and leverage in forecasting the co-Volatility of oil and gold futures. *Energies* **2019**, *12*, 3379. [CrossRef]
32. Asai, M.; Gupta, R.; McAleer, M. Forecasting volatility and co-volatility of crude oil and gold futures: Effects of leverage, jumps, spillovers, and geopolitical risks. *Int. J. Forecast.* **2020**, *36*, 933–948. [CrossRef]
33. Zhao, A.B.; Cheng, T. Stock return prediction: Stacking a variety of models. *J. Empir. Financ.* **2022**, *67*, 288–317. [CrossRef]
34. Tian, F.; Yang, K.; Chen, L. Realized volatility forecasting of agricultural commodity futures using long memory and regime switching. *J. Forecast.* **2017**, *36*, 421–430. [CrossRef]
35. Tian, F.; Yang, K.; Chen, L. Realized volatility forecasting of agricultural commodity futures using the HAR model with time-varying sparsity. *Int. J. Forecast.* **2017**, *33*, 132–152. [CrossRef]
36. Yang, K.; Tian, F.; Chen, L.; Li, S. Realized volatility forecast of agricultural futures using the HAR models with bagging and combination approaches. *Int. Rev. Econ. Financ.* **2017**, *49*, 276–291. [CrossRef]

37. Shiba, S.; Aye, G.C.; Gupta, R.; Goswami, S. Forecastability of agricultural commodity futures realised volatility with daily infectious disease-related uncertainty. *J. Risk Financ. Manag.* **2022**, *15*, 525. [CrossRef]
38. Gupta, R.; Pierdzioch, C. Climate risk and the volatility of agricultural commodity price fluctuations: A prediction experiment. In *Behavioral Finance and Asset Prices: The Influence of Investor's Emotions*; Bourghelle, D., Grandin, P., Jawadi, F., Rozin, P., Eds.; Contributions to Finance and Accounting; Springer: Cham, Switzerland; New York, NY, USA, 2023; pp. 23–44.
39. Luo, J.; Zhang, Q. Air pollution, weather factors, and realized volatility forecasts of agricultural commodity futures. *J. Futures Mark.* **2024**, *44*, 151–217. [CrossRef]
40. Bahloul, W.; Balcilar, M.; Cunado, J.; Gupta, R. The role of economic and financial uncertainties in predicting commodity futures returns and volatility: Evidence from a nonparametric causality-in-quantiles test. *J. Multinatl. Financ. Manag.* **2018**, *45*, 52–71. [CrossRef]
41. Aït-Youcef, C. How index investment impacts commodities: A story about the financialization of agricultural commodities. *Econ. Model.* **2019**, *80*, 23–33. [CrossRef]
42. Ji, Q.; Bahloul, W.; Geng, J.-B.; Gupta, R. Trading behaviour connectedness across commodity markets: Evidence from the hedgers' sentiment perspective. *Res. Int. Bus. Financ.* **2020**, *52*, 101114. [CrossRef]
43. Poon, S.-H.; Granger, C.W.J. Forecasting volatility in financial markets: A review. *J. Econ. Lit.* **2003**, *41*, 478–539. [CrossRef]
44. Rapach, D.E.; Wohar, M.E.; Strauss, J. Forecasting stock return volatility in the presence of structural breaks. In *Forecasting in the Presence of Structural Breaks and Model Uncertainty*; Frontiers of Economics and Globalization; Rapach, D.E., Wohar, M.E., Eds.; Emerald: Bingley, UK, 2018; pp. 381–416.
45. FAO. Price Volatility in Agricultural Markets: Evidence, Impact on Food Security and Policy Responses. 2010. Available online: http://www.fao.org/docrep/013/am053e/am053e00.pdf (accessed on 1 March 2024).
46. FAO. Price Volatility in Food and Agricultural Markets: Policy Responses. 2011. Available online: http://www.fao.org/fileadmin/templates/est/Volatility/Interagency_Report_to_the_G20_on_Food_Price_Volatility.pdf (accessed on 1 March 2024).
47. Ordu, B.M.; Oran, A.; Soytas, U. Is food financialized? Yes, but only when liquidity is abundant. *J. Bank. Financ.* **2018**, *95*, 82–96. [CrossRef]
48. Greb, F.; Prakash, A. Has price volatility changed? In *Food Outlook*; Food and Agriculture Organization of the United Nations: Rome, Italy, 2015.
49. Greb, F.; Prakash, A. Assessing volatility patterns in food crops. In *FAO Commodity and Trade Policy Research Working Paper Series*; FAO: Rome, Italy, 2017. Available online: http://www.fao.org/3/a-i7066e.pdf (accessed on 1 March 2024).
50. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2023. Available online: https://www.R-project.org/ (accessed on 4 May 2024).
51. Rauschenberger, A.; Glaab, E. Predicting correlated outcomes from molecular data. *Bioinformatics* **2021**, *37*, 3889–3895. [CrossRef]
52. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **1996**, *58*, 267–288. [CrossRef]
53. Zou, H.; Hastie, T. Regularization and variable selection via the elastic net. Journal of the Royal Statistical. *Soc. Ser. B Stat. Methodol.* **2005**, *67*, 301–320. [CrossRef]
54. Xing, L.; Lesperance, M.L.; Zhang, X. Simultaneous prediction of multiple outcomes using revised stacking algorithms. *Bioinformatics* **2020**, *36*, 65–72. [CrossRef]
55. Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. *Classification and Regression Trees*; Wadsworth: Belmont, CA, USA, 1984.
56. Friedman, J.; Tibshirani, R.; Hastie, T. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **2010**, *33*, 1–22. [CrossRef] [PubMed]
57. Diebold, F.X.; Yilmaz, K. Measuring financial asset return and volatility spillovers, with application to global equity markets. *Econ. J.* **2009**, *119*, 158–171. [CrossRef]
58. Diebold, F.X.; Yilmaz, K. Better to give than to receive: Predictive directional measurement of volatility spillovers. *Int. J. Forecast.* **2012**, *28*, 57–66. [CrossRef]
59. Urbina, J. Spillover: Spillover/Connectedness Index Based on VAR Modelling. R Package Version 0.1.0.3. 2023. Available online: https://CRAN.R-project.org/package=Spillover (accessed on 6 June 2024).
60. Dai, Z.; Zhang, X.; Liang, C. Efficient predictability of oil price: The role of VIX-based panic index shadow line difference. *Energy Econ.* **2024**, *129*, 107234. [CrossRef]
61. Liang, C.; Wang, L.; Duon, D. More attention and better volatility forecast accuracy: How does war attention affect stock volatility predictability? *J. Econ. Behav. Org.* **2024**, *218*, 1–9. [CrossRef]
62. Nazlioglu, S.; Erdem, C.; Soytas, U. Volatility spillover between oil and agricultural commodity markets. *Energy Econ.* **2013**, *36*, 658–665. [CrossRef]
63. Kang, S.H.; McIver, R.; Yoon, S.M. Dynamic spillover effects among crude oil, precious metal, and agricultural commodity futures markets. *Energy Econ.* **2017**, *62*, 19–32. [CrossRef]
64. Mensi, W.; Tiwari, A.K.; Bouri, E.; Roubaud, D.; Al-Yahyaee, K.H. The dependence structure across oil, wheat, and corn: A wavelet-based copula approach using implied volatility indexes. *Energy Econ.* **2017**, *66*, 122–139. [CrossRef]
65. Ji, Q.; Bouri, E.; Roubaud, D.; Shahzad, S.J.H. Risk spillover between energy and agricultural commodity markets: A dependence-switching CoVaR-copula model. *Energy Econ.* **2018**, *75*, 14–27. [CrossRef]
66. Luo, J.; Ji, Q. High-frequency volatility connectedness between the US crude oil market and China's agricultural commodity markets. *Energy Econ.* **2018**, *76*, 424–438. [CrossRef]

67. Chang, C.L.; Liu, C.P.; McAleer, M. Volatility spillovers for spot, futures, and ETF prices in agriculture and energy. *Energy Econ.* **2019**, *81*, 779–792. [CrossRef]
68. Lu, Y.; Yang, L.; Liu, L. Volatility spillovers between crude oil and agricultural commodity markets since the financial crisis. *Sustainability* **2019**, *11*, 396. [CrossRef]
69. Dahl, R.E.; Oglend, A.; Yahya, M. Dynamics of volatility spillover in commodity markets: Linking crude oil to agriculture. *J. Commod. Mark.* **2020**, *20*, 100–111. [CrossRef]
70. Luo, J.; Zhang, Q. Risk contagions between global oil markets and China's agricultural commodity markets under structural breaks. *Appl. Econ.* **2020**, *53*, 1–22. [CrossRef]
71. Yip, P.S.; Brooks, R.; Do, H.X.; Nguyen, D.K. Dynamic volatility spillover effects between oil and agricultural products. *Int. Rev. Financ. Anal.* **2020**, *69*, 101465. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Analyzing the Influence of Telematics-Based Pricing Strategies on Traditional Rating Factors in Auto Insurance Rate Regulation

Shengkun Xie

Global Management Studies, Ted Rogers School of Management, Toronto Metropolitan University, Toronto, ON M5B 2K3, Canada; shengkun.xie@torontomu.ca; Tel.: +1-416-979-5000 (ext. 543474)

Abstract: This study examines how telematics variables such as annual percentage driven, total miles driven, and driving patterns influence the distributional behaviour of conventional rating factors when incorporated into predictive models for capturing auto insurance risk in rate regulation. To effectively manage the complexity inherent in telematics data, we advocate for the adoption of non-negative sparse principal component analysis (NSPCA) as a structured approach for data dimensionality reduction. By emphasizing sparsity and non-negativity constraints, NSPCA enhances the interpretability and predictive power of models concerning both loss severity and claim counts. This methodological innovation aims to advance statistical analyses within insurance pricing frameworks, ensuring the robustness of predictive models and providing insights crucial for rate regulation strategies specific to the auto insurance sector. Results show that, to enhance auto insurance risk pricing models, it is essential to address data dimension reduction challenges when integrating telematics data variables. Our findings underscore that integrating telematics variables into predictive models maintains the integrity of risk relativity estimates associated with traditional policy variables.

Keywords: non-negative sparse principal component analysis; auto insurance rate regulation; insurance pricing; telematics data; usage-based insurance; dimension reduction

MSC: 62H20; 62J12; 62P05

1. Introduction

Insurance pricing, particularly in auto insurance, has been a pivotal element of the insurance industry, serving not only as a crucial business component but also as an essential tool for ensuring fairness and regulatory compliance [1]. Auto insurance pricing, encompassing both additive and multiplicative algorithms [2,3], has traditionally focused on determining risk relativities for rating factors across various levels. Predictive models, such as generalized linear models (GLMs), have been employed to streamline this process. In auto insurance, a revolutionary approach has emerged, known as usage-based insurance (UBI) [4–7]. UBI takes advantage of telematics data to predict future claim loss and claim frequency, employing this predictive insight to establish risk factor-level relativities for insurance pricing [8]. This cutting-edge methodology involves harnessing real-time data from telematics devices installed in vehicles, with the primary objective of capturing and analyzing the driving patterns of individuals [9–11]. Unlike conventional approaches, UBI introduces a dynamic and personalized dimension to insurance pricing. However, the influence of incorporating telematics data into predictive models for insurance losses remains largely unexplored, particularly from the rate regulation and management of aggregate risk perspective. In rate regulation, the primary focus is on determining the relativity of major risk factors used to assess the underlying risks associated with drivers and vehicles. It is important to assess the impact of incorporating telematics variables on the estimation of risk relativity for traditional major risk factors.

One of the primary advantages of adopting UBI lies in its ability to incentivize responsible driving behaviour [12–14]. By closely monitoring individual driving habits through

telematics data, UBI allows for a more accurate assessment of risk. As a result, drivers who exhibit safe and responsible driving patterns stand to be rewarded with more favourable premium rates. This not only creates a fairer and more personalized insurance pricing structure but also aligns with broader societal goals of promoting road safety and reducing accidents [15,16]. Furthermore, UBI contributes to a paradigm shift in the insurance industry by fostering a proactive approach to risk management. Traditional pricing models rely heavily on historical claim data and generalized assumptions of loss distribution, whereas UBI embraces real-time information, providing insurers with a more granular understanding of risk factors [17]. This not only enhances the accuracy of pricing but also enables insurers to promptly respond to changing driving behaviours and emerging trends [18]. Nevertheless, the implementation of UBI contracts remains relatively slow in North America, with insights from [19] employing economic theory aiding in elucidating this phenomenon. Studying how usage-based insurance (UBI) data can enhance the accuracy of insurance pricing and ensure actuarial fairness, as required in regulatory practices, may aid in promoting the adoption of UBI contracts.

As the insurance pricing landscape continues to evolve, embracing innovative solutions like UBI becomes imperative for insurers to stay competitive, adaptive, and aligned with the evolving expectations of both regulators and policyholders. The pressing question at hand revolves around the impact that the inclusion of telematics risk factors in predictive models has on the estimation of loss severity and claim frequency [20,21]. In rate regulation for auto insurance, the traditional methodology involves determining the relativity associated with major factors, such as age group, insured gender, car age, car use, and marital status [22,23]. Companies then employ these estimated risk relativities as benchmarks for individualized insurance pricing, where more risk factors may be involved. However, as telematics data come to play in the pricing process [14,24,25], introducing variables like annual percentage driven, total miles driven, percentage driven during a day, and percentage driven in a week into predictive models, the dynamics of the claim probability distribution or loss cost distribution for each traditional rating variable may undergo changes. This alteration, essentially an impact of telematics variables, has the potential to exert varying effects on the established regulating variables. It becomes crucial to explore whether these changes lead to substantial shifts in the traditional regulating variables or if their influence remains relatively unaffected. This prompts a compelling need to delve into a comprehensive study on the effect of telematics data on the estimation of both claim frequency and claim severity. By understanding how telematics variables influence the outcome of rate regulation in auto insurance, we can gain valuable insights into the relationship between traditional and evolving factors.

In predictive modelling, managing model complexity is essential for understanding the contribution of each variable. To enhance the efficiency of models estimating claim probability and claim amounts using telematics data, we applied non-negative sparse principal component analysis (NSPCA) [26,27] for dimension reduction. This approach simplifies the model and improves interpretability by systematically reducing the dimensionality of the telematics data, providing a more focused input for the predictive models. NSPCA is particularly suitable for telematics data, as variables like driving speed and acceleration are inherently non-negative, aligning naturally with NSPCA's non-negativity constraint. Additionally, NSPCA's sparsity constraint identifies a smaller set of influential factors, focusing on key driving behaviours. This enhances interpretability while reducing complexity, which is crucial for developing accurate models for predicting claim counts and amounts. Our method involves applying NSPCA to each telematics risk factor at the observation level, reducing the dimensionality by retaining only the most significant principal components. The integration of sparsity ensures that only the most influential levels, contributing meaningfully to these components, are preserved. This deliberate reduction streamlines the data and enhances the interpretability of the major components. Furthermore, the non-negativity constraint ensures that the resultant combinations of

variable levels are both meaningful and practically relevant, reinforcing the effectiveness of the predictive models.

This study aims to explore the impact of various pricing strategies on estimating the risk relativity of traditional policy variables, providing insights into how telematics-based auto insurance pricing influences rate regulation. Following the reduction of telematics data dimensions using NSPCA, three predictive models were constructed for claim counts and claim amounts. Initially, a model was built solely on traditional policy variables, excluding telematics data. Subsequently, a model incorporating telematics data was considered, followed by a combined model using both traditional policy variables and telematics-derived variables. The distributional behaviours of the resulting relativities from these models were compared to empirically calculated ones. This study not only contributes to the refinement of predictive models but also aids in the formulation of more accurate and responsive rate regulation strategies that align with the dynamic landscape of the auto insurance industry. In essence, it allows us to adapt and optimize the regulatory framework to better cater to the evolving needs and complexities of the auto insurance market.

The novelty of this research is the introduction of NSPCA to the analysis of telematics, which helps identify essential components of each telematics variable and reduces dimensionality, enabling more effective management of telematics data. The findings of this study carry substantial implications for auto insurance rate regulation. They affirm that the emphasis on traditional policy variables in rate regulation practices is justified. The integration of telematics data alongside traditional policy variables does not result in a significant deviation in the distribution of key regulatory risk factors. Hence, it represents a suitable pricing strategy for insurance markets that incorporate UBI.

2. Literature Review

The existing literature has predominantly concentrated on the development of statistical models using telematics data, with a primary focus on achieving more precise pricing for auto insurance policies and fostering responsible driving habits. For instance, ref. [24] employed data from a Belgian telematics product specifically targeting young drivers. The research studied how to determine car insurance premiums based on telematics data. By using statistical modelling techniques, specifically, generalized additive models, the research convincingly demonstrated that incorporating telematics variables significantly improves predictive accuracy. The study revealed that gender, once considered a crucial rating variable, becomes unnecessary when assessing expected claim frequency in the presence of telematics variables. The work conducted in [24] shares some similarities with our work, but we address the impact of telematics variables on traditional policy variables including gender from a rate regulation perspective.

Modelling the annual usage of a driver is a key aspect in UBI. In [28], the research contributes to this aspect by using telematics data to enhance auto insurance rating mechanisms. The analysis incorporates the distance travelled per year into a zero-inflated Poisson model, strategically predicting the excess of zero claims. The findings unveil a learning effect, suggesting that extended driving durations may lead to higher premiums, but drivers accumulating longer distances over time may benefit from potential discounts. This underscores the potential of telematics information in not only refining insurance models but also bolstering efforts to enhance overall traffic safety. Another interesting study using teletmatics data by [29] employed anomaly detection algorithms on vehicles' trip summaries. The study developed routine and anomaly profiles for each vehicle, finding that features extracted from the vehicles' profile enhance claim classification in the framework of anomaly-based insurance. Other recent research that uses telematics data for insurance pricing or discovering the driving patterns so that the associated risk level can be identified, has appeared in [30–33].

In addition to telematics data, the utilization of GPS data for insurance ratemaking has emerged as a significant avenue of exploration. A notable study by [34] integrates GPS data into motor insurance ratemaking, highlighting the insurance sector's potential

to determine premium rates based on driver behaviour. Employing count data regression models, the approach incorporates parameters that capture various characteristics of automobile usage known to influence claiming behaviour. By augmenting classical frequency models with telematics information derived from usage-based insurance policies, the study demonstrates the substantial impact of both the distance travelled and driver habits on the expected number of accidents. This, in turn, directly influences the premium of insurance coverage.

Similarly, the investigation carried out by [35] studied the integration of second-by-second GPS data into auto insurance pricing structures. The study showcases the effective use of real-time GPS trajectory data obtained from a traffic APP and survey data. Notably, the research underscores the potential of incorporating unique contextual-based risk measurements, extending beyond traditional UBI factors. By comparing driving speed to others on the same road segment, the study introduces an innovative approach to assessing risk. The findings reveal compelling correlations between driving behaviours, such as hard brakes, hard starts, peak-time travel, speeding, deviations from traffic flow, and accident rates. These insights provide valuable guidance for insurance companies venturing into the UBI sector, aiding in the establishment of accurate auto insurance premium rates and the mitigation of underwriting losses. In our research, we delved into the correlation between various telematics variables and their influence on the relative risk associated with conventional rating factors.

In the work conducted by [36], the research scrutinizes the predictive capabilities of telematics data features extracted through speed acceleration heatmaps. The telematics covariates under investigation, which involved K-means classification, principal components, and bottleneck activations from a neural network, exhibited a superiority in out-of-sample prediction for claim frequencies when compared to conventional factors such as driver's age, as demonstrated in the conducted case study. The numerical examples presented in the study strongly advocate for the adoption of these telematics covariates, especially emphasizing the efficacy of the first principal component and bottleneck activations, as a recommended approach to enhance the precision of car insurance pricing models. Concurrently, the study presented in [37] introduces innovative pricing strategies for German car insurance, specifically designed to seamlessly integrate telematics data into actuarial pricing decisions. The outcomes of the study reveal a substantial impact on pricing decisions, enabling the implementation of usage-based insurance premiums. This is achieved by applying discounts or surcharges based on driving behaviour, marking a significant shift in the traditional approach to car insurance pricing. The study not only showcases the feasibility of incorporating telematics data but also emphasizes its potential to revolutionize the landscape of actuarial decision-making in the context of car insurance.

In a complementary vein, the study conducted by [38] analyzed UBI adopters and discovered the impact of UBI on driving behaviour. The study revealed a 21% reduction in daily average hard-brake frequency after six months of UBI adoption, indicating a substantial improvement in the safety of driving habits. Furthermore, the research identified heterogeneous effects across demographic groups, highlighting that younger drivers and women tend to exhibit more significant improvements. Moreover, the correlation between negative feedback and economic incentives with enhanced driving behaviour emphasizes the societal benefits of UBI, particularly in the realm of improved road safety. These insights collectively contribute to the multifaceted impacts of UBI on both insurers and drivers. As the study demonstrates that UBI pricing strategies may rely on traditional rating variables, particularly those used for regulatory purposes, it is crucial to explore the extent of these influences.

While there exists limited research focusing on addressing the regulatory impact of UBI on auto insurance pricing, a notable study by [6] fills this gap by investigating the effects of UBI on private passenger auto liability insurers' underwriting performance. The findings of this study reveal that UBI significantly enhances underwriting performance by reducing the loss ratio, especially among early adopters. Notably, as UBI matures, its

benefits become more pronounced, leading to a notable increase in market share for early adopters and an overall positive impact on return on assets and return on equity. This underscores the growing importance of UBI in reshaping the landscape of auto insurance pricing, with tangible benefits emerging over time.

3. Materials and Methods

3.1. Synthetic Data

Telematics or GPS data, while valuable for quantifying relationship between driving behaviour and insurance claims, are often limited in scope for research purposes. This is due to several factors: access restrictions, privacy concerns, and the proprietary nature of telematics data, which are typically owned by insurance companies. As a result, researchers may only have access to small or incomplete datasets that are insufficient for robust statistical analysis or for developing highly predictive models. To address this limitations, researchers often resort to data augmentation techniques such as oversampling or using synthetic data generation methods like the Synthetic Minority Oversampling Technique (SMOTE). These techniques are particularly useful when the available data are imbalanced or too small for effective model training. Oversampling involves replicating data observations from underrepresented classes to balance the dataset, while SMOTE generates new synthetic data points by interpolating between existing observations. In the case of telematics data, SMOTE can be applied to create synthetic driving behaviour patterns that mimic real-world variations, helping to enlarge the dataset and reduce the risk of model overfitting. By using synthetic data generation techniques, researchers can create larger, more balanced datasets, which not only improve model training but also enhance the reliability of predictive models. This approach allows for better generalization, making it possible to test models on a more comprehensive range of driving behaviours, and, ultimately, improving the robustness of research findings.

The dataset used in this research originates from a synthetic dataset comprising 100,000 auto usage-based insurance (UBI) policies, as provided by [39]. This synthetic dataset encompasses three distinct types of variables: traditional policy variables, driving-pattern-related variables, and response variables, specifically claim amounts and claim occurrence. Across the dataset, there are 52 variables, encompassing essential factors such as insured age group, car age range, annual mileage driven, and credit scores. The dataset also includes telematics variables that detail driving patterns, including metrics such as acceleration and braking speeds, as well as the daily percentage of driving within a week. Additionally, certain variables capture the intensity of left or right turns. In the dataset, each telematics data variable is linked to a specific level of a corresponding telematics risk factor. These variables are expected to demonstrate correlations, as they authentically align with the same underlying risk factor. For example, the percentage of driving is presented in terms of the weekly distribution of driving relative frequencies for each day within a week. Consequently, we end up with seven correlated variables, each capturing the relative frequency of driving for a specific day. Analogous variables include those related to acceleration, braking, the percentage of driving over an extended period during a day, the percentage of driving during morning and afternoon rush hours, as well as the intensities of left and right turns. In essence, these variables share commonality in their measurement due to their connection to broader driving behaviours and patterns. These data variables mentioned above require dimension reduction due to their correlation among levels within the factor. The process of dimension reduction and how we deal with the reduced levels of the telematics variable is illustrated in Figure 1. There is also a set of single-level factor variables such as annual percentage driven, total mileage driven, and average of day driven per week.

It is essential to highlight that this dataset follows a transactional structure, where each record denotes the presence or absence of a claim. In cases where a claim is reported, the associated record includes the corresponding loss amount. For a more detailed exploration

of telematics variables and a comprehensive descriptive analysis of the dataset's variables, interested readers can refer to [39].

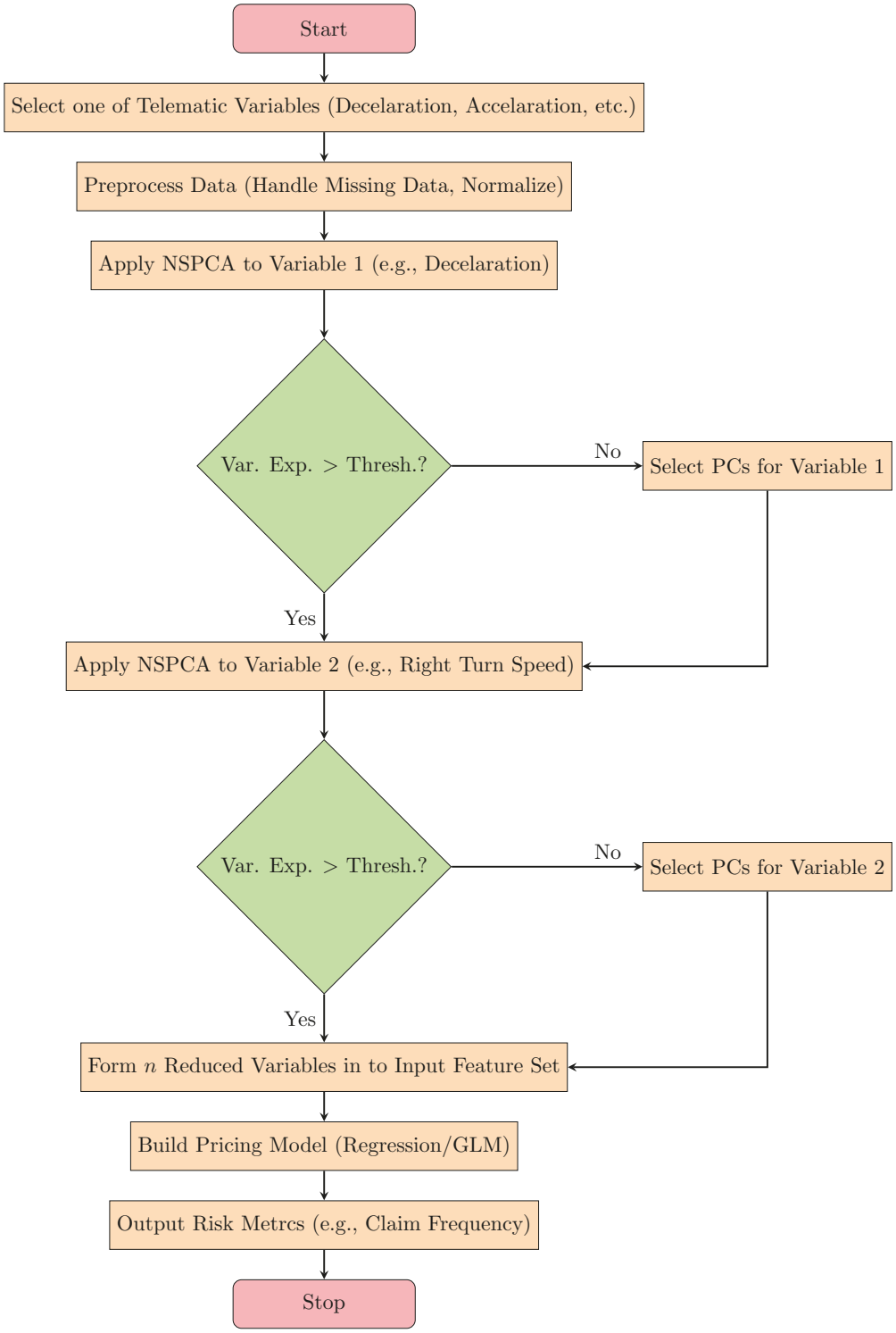


Figure 1. Flowchart illustrating the modelling process with dimensionality reduction applied to telematics variables.

3.2. Non-Negative Sparse PCA

In statistical multivariate analysis, NSPCA extends classical PCA by imposing constraints of non-negativity and sparsity on the principal components. Another related method is Non-negative Sparse Matrix Factorization (NSMF), designed to decompose

matrices while adhering to sparsity and non-negativity constraints. However, the PCA-based approach is deemed more advantageous than matrix decomposition in this research. This is attributed to its focus on concentrating variation into major principal components, facilitating an additional layer of sparsity control. This paper utilized such an approach to achieve dimension reduction, as detailed in the subsequent sections. Let us assume $X^{(i)}$ to be the data matrix correspond to the ith group of telematics data variables, where $i = 1, 2, \ldots 7$ in this work. They include daily percentage driven per week, acceleration intensity, braking intensity, rush hours percentage driven, left turn intensity, and right turn intensity. The objective function for NSPCA is given by

$$\text{minimize} \quad \|X^{(i)} - Z^{(i)}D^{(i)}\|_F^2 + \lambda\|D^{(i)}\|_1, \text{ subject to } {Z^{(i)}}^\top Z^{(i)} = I, \text{ where } Z^{(i)}, D^{(i)} \geq 0. \tag{1}$$

Here, $\|\cdot\|_F$ denotes the Frobenius norm, and $\|\cdot\|_1$ is the L_1-norm. Each $X^{(i)}$ has n samples and $p^{(i)}$ variables, where n = 100,000 and the value of $p^{(i)}$ depends on i. For instance, if $p^{(i)}$ refers to daily percentage driven per week, then it is equal to 7. $Z^{(i)}$ is the matrix of principal components scores of $X^{(i)}$. $D^{(i)}$ is the matrix of principal component loadings. λ is the sparsity parameter. When $Z^{(i)}$ and $D^{(i)}$ have no non-negativity constraint, the given minimization problem is reduced to sparse PCA, and if, further, there is sparsity control, which means that $\lambda = 0$, then the problem just becomes a traditional PCA via matrix decomposition. The introduction of sparsity to regulate the cardinality of principal component loading represents the initial layer in our approach. Additionally, we incorporate a secondary layer of sparsity control, where we selectively retain the most significant principal components. These retained components serve as the foundation for our telematics-based predictors, which, in turn, form the basis for the predictive models we have developed. To obtain these non-negative sparse principal components, the *nsprcomp* [40] R 4.3.0 software package (see https://www.r-project.org, created originally by by John Chambers and colleagues) was used for implementation.

As we mentioned, NSPCA is an extension of classical PCA that has been widely adopted for high dimensional data analysis, especially when interpretability is a priority. The non-negativity constraint allows for more meaningful interpretations of the principal components, especially when dealing with data that should not be negative values, such as telematics variables. Applications of NSPCA have been explored in fields like genomics, image processing, and finance, where it has proven useful in extracting interpretable components from complex datasets while reducing dimensionality [41,42]. The incorporation of sparsity further enhances the model's utility by identifying a subset of factor levels that are most influential, thereby reducing noise and simplifying interpretation. This is especially advantageous in telematics data, where only a few driving behaviours may meaningfully impact claim outcomes.

Despite its advantages, NSPCA does introduce certain computational and practical challenges. One key limitation is its computational complexity, especially when applied to large datasets. The iterative optimization process required to enforce both non-negativity and sparsity can be computationally intensive, particularly when dealing with high dimensional data. This is why we chose to reduce the dimensionality within each individual telematics variable, rather than combining all telematics variables together. Moreover, the choice of the sparsity parameter, λ, is crucial to balance between sparsity and model fit, and determining an optimal λ often involves cross-validation, adding further computational demands. The Expectation Maximization (EM) algorithm [40] was proposed to address some of these challenges, but the convergence of this method can be slow in practice, especially for large datasets.

Another practical consideration is the model interpretability–performance trade-off [43]. While the non-negativity constraint enhances the interpretability of the principal components, it may limit the model's ability to capture more subtle relationships between variables. This trade-off has been discussed in various applications, with studies showing that PCA can sometimes result in a loss of predictive power [44] compared to more flexible

models that do not impose such constraints. In our application, we mitigate this risk by incorporating an additional layer of sparsity control, retaining only the most significant components, which ensures that the model focuses on the most critical driving behaviours influencing claim risk.

3.3. Modelling with Traditional Policy Variables

Auto insurance pricing based on traditional rating variables involves assessing risk factors such as age group, insured gender, car age, car use, and marital status. Insurance companies estimate the risk relativities associated with these variables to establish benchmarks for pricing. The traditional approach relies on historical data and generalized assumptions to determine the impact of these variables on the likelihood and severity of claims. This method provides a standardized framework for pricing, with premiums adjusted based on the perceived risk associated with individual drivers and their vehicles. In predicting claim frequency using traditional policy variables, the following generalized linear model is constructed:

$$\begin{aligned} Occurrence \sim\ & Insured.AgeGroup + Insured.sex + Car.age.range + Marital + Car.use + \\ & Credit.score.range + Location.Cluster + Annual.miles.drive.range + Years.noclaims.range. \end{aligned} \tag{2}$$

Similarly, the model used to predict the claim amount is give as follows:

$$\begin{aligned} Amount \sim\ & Insured.AgeGroup + Insured.sex + Car.age.range + Marital + Car.use + \\ & Credit.score.range + Location.Cluster + Annual.miles.drive.range + Years.noclaims.range. \end{aligned} \tag{3}$$

It should be noted that all traditional policy variables are grouped and their estimated coefficients are for each level with the factor. The purpose of grouping is to improve the creditability of variables used in pricing and maintain a relative stability for each risk level.

3.4. Modelling with Telematics Data Only

When auto insurance pricing is exclusively based on telematics data, the insurance industry undergoes a significant shift towards a more dynamic and personalized model. Telematics data, encompassing real-time information about driving behaviour collected through devices installed in vehicles, take precedence as the primary factor influencing insurance premiums. This approach allows for a more detailed assessment of risk by focusing on individual driving patterns and habits. One of the notable implications of relying solely on telematics data is the heightened level of personalization in insurance premiums. Pricing becomes intricately tailored to the specific driving behaviour of each individual, potentially resulting in lower premiums for safe drivers and higher costs for those engaging in riskier behaviours. This departure from traditional models emphasizes a more nuanced understanding of risk based on real-time data.

The models used to predict claim probability and claim amounts become the following based on the data we used:

$$\begin{aligned} Occurrence \sim\ & Annual.pct.driven + Total.miles.driven + Pct.drive1 + Pct.drive2 + Pct.drive3 \\ & + Pct.drive4 + Pct.drive.hrs + Rush.ampm + Avgdays.week + Accel1 + Accel2 + Brake1 \\ & + Brake2 + Left.turn1 + Left.turn2 + Right.turn1 + Right.turn2, \end{aligned} \tag{4}$$

where variables $Pct.drive1$, $Pct.drive2$, $Pct.drive3$, and $Pct.drive4$ represent the first four PCs retained of daily percentage of driving over a week; $Accel1$ and $Accel2$ are the first two PCs retained of the acceleration levels; the variables $Brake1$ and $Brake2$ correspond to the first two PCs of braking level. Similarly, the variables $Left.turn1$ and $Left.turn2$ represent the first two PCs of left turn intensity, while $Right.turn1$ and $Right.turn2$ represent the first two PCs of right turn intensity. The dimension reduction via non-negative sparse PCA makes the model more interpretable, as the number of variables included in the model has

been significantly reduced. Similarly, the model used to predict the claim amount is give as follows:

$$\begin{aligned} Amount \sim\ & Annual.pct.driven + Total.miles.driven + Pct.drive1 + Pct.drive2 + Pct.drive3 \\ & + Pct.drive4 + Pct.drive.hrs + Rush.ampm + Avgdays.week + Accel1 + Accel2 + Brake1 \\ & + Brake2 + Left.turn1 + Left.turn2 + Right.turn1 + Right.turn2, \end{aligned} \tag{5}$$

Moreover, the use of telematics data enables real-time adjustments in pricing. With continuous updates from telematics devices, insurers can promptly adapt their pricing structures to reflect any changes in driving habits. This responsiveness contributes to a more adaptive and immediate approach to assessing and managing risk factors compared to traditional models that rely on historical data and assumptions. While the move towards telematics-based pricing brings several benefits, challenges and considerations exist. Issues related to privacy, data security, and potential bias in interpreting telematics data must be carefully addressed. Additionally, the transition from traditional rating variables to a telematics-centric approach requires thoughtful consideration of regulatory frameworks and industry standards. In essence, the industry's evolution towards telematics-based pricing represents a broader shift towards a more data-driven, personalized, and responsive model in the auto insurance landscape.

3.5. Modelling with Traditional Policy Variables and Telematics Variables

The last modelling strategy was to include both traditional policy variables and telematics variables. This hybrid approach offers a more comprehensive and accurate assessment of individual risk profiles. By integrating demographic factors like age and gender with real-time driving behaviour insights, insurers can create better models that reflect both historical characteristics and current habits. This precision in risk assessment contributes to fairer and more personalized insurance premiums. Moreover, the inclusion of traditional rating variables alongside telematics data enhances the overall fairness and equity of the pricing model. This is particularly significant in ensuring that insurance premiums are not solely determined by a specific set of factors, mitigating potential biases. The combination of traditional and telematics data creates a more balanced and objective approach, aligning with industry efforts to promote transparency and fairness in insurance pricing. The integration of traditional and telematics data also enhances the robustness of predictive models against data gaps and uncertainties. In situations where telematics data may be incomplete or unavailable, traditional variables act as stabilizing factors, ensuring the reliability and adaptability of the pricing model. This resilience is crucial for insurers to navigate changing circumstances without compromising the accuracy of risk assessments.

From a regulatory perspective, the impact of combining traditional and telematics data on auto insurance rate regulation is significant. Many regulators require the consideration of specific traditional rating variables. By incorporating these alongside telematics data, insurers can ensure compliance with regulatory standards. The comprehensive approach aligns with regulatory expectations while allowing insurers to benefit from the valuable insights derived from driving behaviour data. In the analysis section, we will explore the relativity patterns of traditional policy variables based on the prediction results of claim occurrence and claim amount for each policyholder.

In this study, we used a generalized linear model to analyze the data, opting for a Poisson distribution to represent the number of claims and a Gamma distribution to account for the variability in claim amounts. We utilized the identity link function for the Poisson distribution and applied a log-link function for the Gamma distribution, aligning with common practices in auto insurance rate regulation for modelling claim frequency and severity. For telematics data, this is a prudent strategy for several reasons. Firstly, telematics data often exhibit characteristics such as skewness and heteroscedasticity in loss amounts, which are well-captured by the flexibility of the gamma distribution. This distribution allows for modelling non-negative continuous data with a right-skewed distribution,

making it suitable for capturing the variability in loss amounts observed in insurance claims data. Secondly, the Poisson distribution is commonly employed for modelling count data, such as the frequency of insurance claims in telematics datasets. It naturally accommodates the discrete nature of claim counts and is particularly effective when dealing with rare events, which is often the case in insurance claims data. By incorporating these distributions into a GLM framework, we can effectively model both the frequency and severity of claims simultaneously, providing a comprehensive understanding of the underlying risk factors in telematics data. This approach enables insurers to better assess and manage risks, leading to more accurate pricing and improved decision-making processes. Although alternative error functions were not explored in this work, the rationale behind our model selection is rooted in its practical applicability within the insurance industry.

3.6. Some Discussions on Methodologies

In predictive modelling, achieving a manageable model size is pivotal for gaining deeper insights into the contribution of each variable to the model-building process. This underscores the primary rationale behind our decision to implement dimension reduction on each telematics factor, characterized by different levels. This strategic approach is aimed at effectively reducing the model's complexity and facilitating a better understanding of its contributions of each factor level.

NSPCA and NSMF are two methods often employed in machine learning and high dimensional data analysis that share certain fundamental principles. Both methods incorporate a non-negativity constraint, ensuring that the derived components or factors are non-negative, aligning with practical interpretations in various real-world problems. Additionally, both NSPCA and NSMF leverage a sparsity constraint, aiming to identify a set of components or factors with a substantial number of zero elements. This sparsity control is advantageous for promoting interpretability and efficiency in representing complex datasets like telematics data. Moreover, both techniques often find application in dimensionality reduction, seeking to condense the information in the data into a reduced set of non-negative and sparse components that capture essential patterns. Nonetheless, despite these commonalities, NSPCA and NSMF diverge in their primary objectives and structures. NSPCA focuses on extracting a set of sparse and non-negative principal components that maximize the variance of the data. These components are orthogonal to each other, emphasizing the capture of uncorrelated directions of maximum variance. In contrast, NSMF involves decomposing a given matrix into two non-negative matrices. The objective is to find a sparse and non-negative representation of the input data, where the components do not need to be orthogonal. This reflects a different approach, emphasizing additive combinations of basis vectors to approximate the input matrix.

In this research, NSPCA stood out as a more suitable technique for the dimension reduction of telematics data used in the predictive modelling of claim counts and claim amounts for several reasons. Our telematics data involve variables such as driving speed, acceleration, and other behavioural factors, which inherently should not have negative values. The non-negativity constraint enforced by NSPCA aligns seamlessly with the nature of telematics data, preserving their inherent characteristics in the process of dimensionality reduction. Moreover, the sparsity constraint imposed by NSPCA is particularly advantageous in handling the high-dimensional telematics data. This constraint promotes a sparse set of principal components, enabling the identification of a subset of influential features. In the context of telematics data, where specific driving behaviours may significantly impact claims, having sparse principal components facilitates a more focused and interpretable representation of the telematics data. The interpretability of the extracted components is a crucial aspect, especially when dealing with telematics data for insurance purposes. NSPCA's ability to induce sparsity enhances interpretability, allowing actuaries and insurance professionals to attribute significance to specific driving behaviours. This feature contributes to a better understanding of the impact of various factors on claim counts and amounts. In addition to interpretability, the dimension reduction capabilities

of NSPCA are particularly relevant for telematics data. By reducing dimensionality while preserving the most relevant features, NSPCA aids in building predictive models that avoid overfitting and enhance generalization performance. By focusing on capturing the most significant variations in the data, this technique identifies key driving behaviours that contribute to a higher likelihood of claims or affect claim amounts. This emphasis on influential factors can contribute to the development of more accurate predictive models based on telematics data.

However, the application of NSPCA introduces a potential trade-off between enhancing interpretability and maintaining predictive power. While NSPCA enforces non-negativity, which aligns well with the nature of telematics variables, this restriction may limit the flexibility of the model. In many machine learning methods, data transformations like standardization, where variables are centred around the mean and scaled to unit variance, are common pre-processing steps that allow variables to assume both positive and negative values. Such transformations can improve interpretability by representing deviations from the mean and enabling a more balanced understanding of variable contributions. The non-negativity constraint in NSPCA may, however, hinder the ability to capture such variations, potentially leading to a reduction in predictive power. Standardized variables may reveal patterns or relationships that could be obscured when constrained to non-negative values. To address this issue, one possible remedy is to reverse the data transformation (i.e., de-standardize) prior to applying NSPCA. By transforming the data back to their original non-negative scale after standardization, the method's non-negativity constraint can be upheld while still benefiting from pre-processing techniques that aid in balancing and scaling the data.

Alternatively, NSPCA could be applied selectively, avoiding pre-processing steps that introduce negative values where this constraint would conflict with model assumptions. This selective approach would ensure that NSPCA's strengths in promoting interpretability through sparsity are maintained without undermining the potential for predictive accuracy. Ultimately, careful consideration must be given to whether the enhanced interpretability of NSPCA compensates for any loss in predictive power, with additional strategies employed to mitigate the effect of non-negativity restrictions on standardized data. This balance between interpretability and predictive accuracy is crucial and should be tailored to the specific goals and characteristics of the data being analyzed.

4. Results

In this section, we provide a comprehensive analysis of the results obtained through our proposed methodology. Tables 1 and 2 showcase the principal component loading vectors for each telematics variable across different principal components. As there are seven distinct telematics variables, each with varying levels, the maximum number of components varies accordingly. An insightful observation from the results presented in the tables is the discernible impact of imposed sparsity, resulting in a reduction in the cardinality of each vector. This reduction facilitates the exclusion of less significant levels within each variable, enabling a focused capture of the primary effects from more substantial levels. Furthermore, the obtained principal components exhibit clear differentiation, aiding us in the selection of the final retained principal component for predictive modelling of both claim counts and claim amounts. Overall, these findings underscore the efficacy of our methodology in distilling meaningful insights from complex telematics data, thereby enhancing the predictive accuracy and interpretability crucial for informed decision-making in insurance analytics.

The telematics variable representing the percentage of daily driving per week is segmented into seven levels, each corresponding to a specific day of the week. Through the application of non-negative sparse PCA, our analysis reveals that the primary variation (approximately 75%) is captured within the first four components. By leveraging the sparsity constraint on the principal component loading vector, we discern distinct driving patterns: the first component characterizes weekend driving, the second elucidates patterns

observed on Wednesday and Thursday, the third predominantly reflects Friday's patterns, and the fourth corresponds to Monday and Tuesday driving behaviours. This clustering by non-negative sparse PCA effectively categorizes driving habits into four distinct groups, each aligned with specific days or combinations of days within the week. Such insights offer a nuanced understanding of driving behaviours over the course of a week. Two additional driving metrics pertain to extended durations and rush-hour travel. Extended driving is categorized into three levels: more than 2 h, more than 3 h, and more than 4 h. Meanwhile, rush-hour driving encompasses both morning and afternoon peak periods. Our analysis reveals that maintaining multiple levels for these factors is unnecessary. Through a weighted average of the percentage of driving across the original levels, we can consolidate these two factors into single-level constructs. These newly derived factors serve as refined predictors for predictive modelling of claim counts and claim amounts. This streamlined approach enhances the efficiency of our predictive models by capturing the essence of extended and rush-hour driving in a more concise and representative manner.

Table 1. The principal component load vectors obtained from the NSPCA of each telematics risk factor. The dimensions for the factors of weekly percentage driven, acceleration, braking, percentage driven for extended periods, and percentage driven during morning and afternoon are, respectively, 7, 6, 6, 3 and 2.

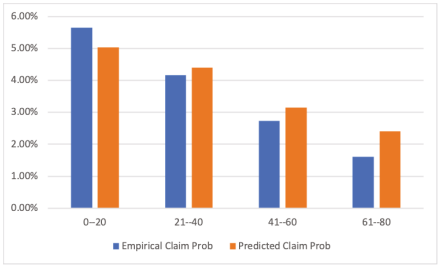
	PC1	**PC2**	**PC3**	**PC4**	**PC5**	**PC6**	**PC7**
Pct.drive.Mon	0	0	0	0.564	0.959	0.227	0
Pct.drive.Tue	0	0	0	0.826	0	0	0.104
Pct.drive.Wed	0	0.690	0	0.016	0	0.924	0
Pct.drive.Thr	0	0.723	0.316	0	0.249	0	0
Pct.drive.Fri	0	0	0.920	0	0	0.307	0.215
Pct.drive.Sat	0.722	0.020	0.233	0	0	0	0
Pct.drive.Sun	0.692	0	0	0	0.134	0	0.971
Standard Deviation	1.140	1.035	0.979	1.018	0.808	0.702	0.600
Proportion of Variance	0.221	0.183	0.163	0.176	0.111	0.084	0.061
Cumulative Proportion	0.221	0.404	0.567	0.744	0.855	0.939	1
Accel.06miles	0	0.497	0.999	0.046	0.010	0.054	
Accel.08miles	0	0.637	0	0.226	0	0	
Accel.09miles	0	0.590	0	0	0.977	0	
Accel.11miles	0.570	0	0	0.973	0.211	0.007	
Accel.12miles	0.589	0	0	0	0	0.998	
Accel.14miles	0.573	0	0.053	0	0	0	
Standard Deviation	1.692	1.532	0.649	0.257	0.225	0.136	
Proportion of Variance	0.496	0.407	0.073	0.011	0.009	0.003	
Cumulative Proportion	0.496	0.904	0.977	0.988	0.997	1	
Brake.06miles	0	0.500	1.000	0	0	0.00004	
Brake.08miles	0	0.646	0	0	0.099	0.016	
Brake.09miles	0	0.577	0	0.996	0	0	
Brake.11miles	0.574	0	0	0	0	0	
Brake.12miles	0.585	0	0	0.018	0	1.000	
Brake.14miles	0.573	0	0.025	0.090	0.995	0	
Standard Deviation	1.707	1.517	0.656	0.266	0.202	0.107	
Proportion of Variance	0.505	0.399	0.075	0.012	0.007	0.002	
Cumulative Proportion	0.505	0.904	0.979	0.991	0.998	1	
Pct.drive.2hrs	0.544	1	0				
Pct.drive.3hrs	0.615	0	0				
Pct.drive.4hrs	0.570	0	1				
Standard Deviation	1.556	0.533	0.308				
Proportion of Variance	0.865	0.101	0.034				
Cumulative Proportion	0.865	0.966	1				
Pct.drive.rush am	0.697	1					
Pct.drive.rush pm	0.717	0					
Standard Deviation	1.162	0.578					
Proportion of Variance	0.801	0.199					
Cumulative Proportion	0.801	1					

Table 2. The principal component load vectors obtained from the NSPCA of each telematics risk factor. The dimensions for the factors of left and right turns are 5.

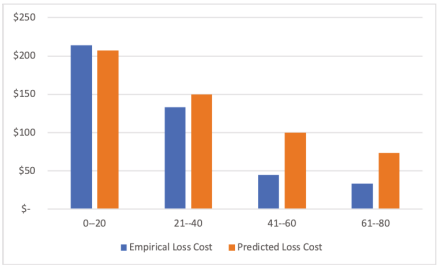
	PC1	**PC2**	**PC3**	**PC4**	**PC5**
Left.turn.intensity08	0	0.707	0.263	0	0
Left.turn.intensity09	0	0.707	0	0.964	0
Left.turn.intensity10	0.576	0.021	0.965	0.265	0
Left.turn.intensity11	0.579	0	0	0	1
Left.turn.intensity12	0.577	0	0	0	0
Standard Deviation	1.726	1.412	0.107	0.052	0.042
Proportion of Variance	0.597	0.400	0.002	0.001	0.0004
Cumulative Proportion	0.597	0.997	0.999	1.000	1
Right.turn.intensity08	0	0.707	0.271	0.881	0.426
Right.turn.intensity09	0	0.707	0	0	0
Right.turn.intensity10	0.576	0.034	0.963	0	0
Right.turn.intensity11	0.581	0	0	0	0.905
Right.turn.intensity12	0.576	0	0	0.474	0
Standard Deviation	1.721	1.408	0.132	0.096	0.071
Proportion of Variance	0.595	0.398	0.004	0.002	0.001
Cumulative Proportion	0.595	0.994	0.997	0.999	1

Acceleration and braking intensity serve as pivotal telematics variables aimed at capturing driving behaviour. Within the dataset, each of these variables is delineated into six levels of intensity, reflecting the degree to which a vehicle accelerates or decelerates. Higher intensities in acceleration or deceleration correlate with increased driver risk. Remarkably, both acceleration and deceleration exhibit a distinct threshold of intensity, typically falling between 9 and 11 miles. Consequently, the driving patterns associated with these actions can be effectively categorized into two distinct levels: intensities below 10 miles and intensities above 10 miles. This delineation suggests that merely two principal components, collectively explaining approximately 90% of the data variation, suffice for subsequent predictive modelling pertaining to these variables. Comparable to acceleration and deceleration, the intensity of left and right turns serves as a crucial metric for gauging driving safety. Analysis indicates a discernible cut-off range between 9 and 10 miles per hour, indicating the need for two principal components to effectively encapsulate these data. Consequently, a new factor is constructed, with intensity below 10 miles per hour constituting one level, and intensity above 10 miles per hour forming the other. Together, these two principal components elucidate nearly 100% of the variation in turn intensity, providing comprehensive insights into driving behaviours.

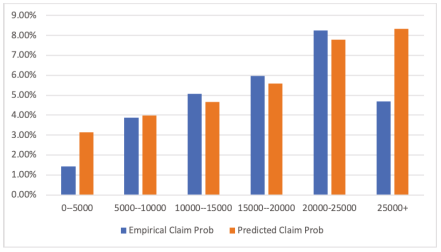
We further examined how various pricing strategies impact traditional policy variables, encompassing years of no claims, annual mileage driven, credit scores, insured age, and car age. Our focus on these particular variables stemmed from their availability within the dataset. Figures 2 and 3 present a comparison of claim probability and loss costs concerning traditional policy variables between empirical estimates and predicted values derived from a model incorporating only telematics variables. The telematics variables utilized for forecasting claim counts and loss costs comprise a reduced set obtained through non-negative sparse PCA. Notably, we discern significant disparities for specific levels within the policy variables. For instance, concerning years of no claims surpassing 40, the model overestimates loss costs. Similarly, for loss costs linked to credit scores, those associated with lower scores, such as 700 and below, are underestimated. A similar situation arises with insured age and car age, wherein certain levels witness substantial underestimation while others experience overestimation within the policy variable.



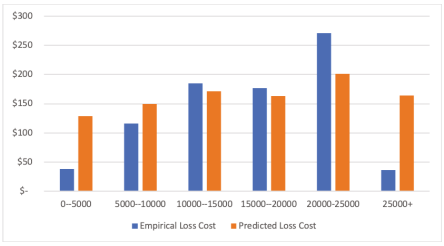
(**a**) Claim probability, years of no claims



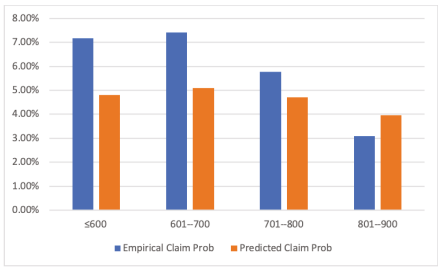
(**b**) Loss cost, years of no claims



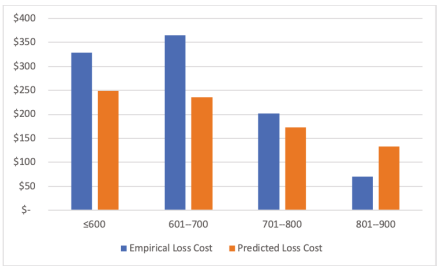
(**c**) Claim probability, annual mileage driven



(**d**) Loss Cost, annual mileage driven

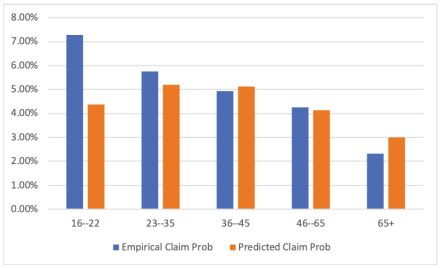


(**e**) Claim probability, credit scores

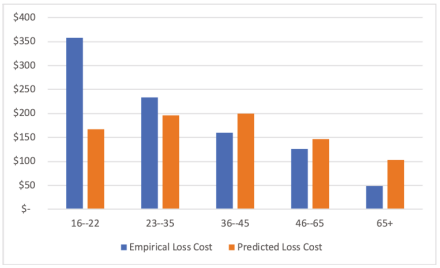


(**f**) Loss cost, credit scores

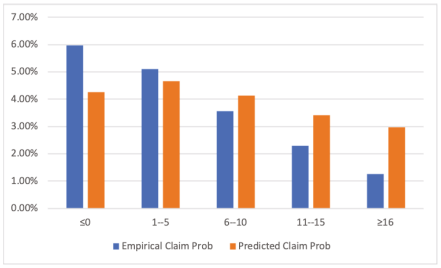
Figure 2. The predicted probability and loss cost by different levels of years of no claims, annual mileage driven, and credit scores, obtained from the model using only telematics data variables, compared to the empirical results.



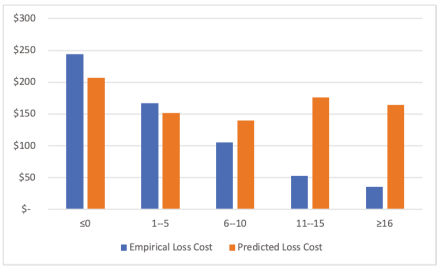
(**a**) Claim probability, insured age



(**b**) Loss cost, insured age



(**c**) Claim probability, car age



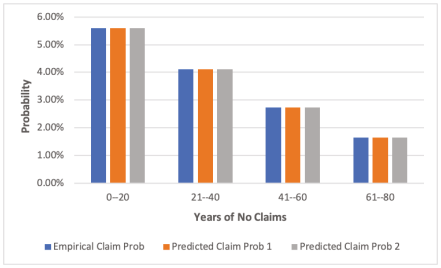
(**d**) Loss cost, car age

Figure 3. The predicted probability and loss cost by different levels of insured age and car age, obtained from the model using only telematics data variables, compared to the empirical results.

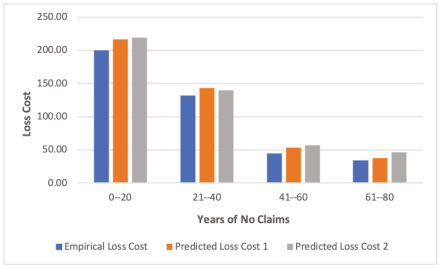
A finding emerges from the analysis: the predictive model relying solely on telematics variables demonstrates superior performance in recovering empirical patterns for predicting claim counts compared to loss prediction. This observation suggests the potential capacity of telematics variables to reflect the influence of driving patterns on traditional policy variables, thus underscoring the critical necessity of factoring in such impacts when regulating insurance rates. However, it becomes evident that solely considering telematics variables in predicting claim counts or claim amounts may not suffice, given the substantial deviation from empirical results. This highlights the complexity of insurance risk assessment and the importance of incorporating multiple factors beyond just telematics data to achieve more accurate predictions and better inform rate-making practices.

Telematics variables are directly related to driving behaviour, such as acceleration, braking, and cornering. By analyzing these driving behaviours, insurers can more accurately assess the risk profile of individual drivers and predict the likelihood of future accidents, thereby influencing claim frequency predictions. However, when it comes to modelling claim amounts, telematics data may have a lesser impact, as factors influencing severity, such as the cost of repairs or medical expenses, are often influenced by external variables beyond driving behaviour. While telematics data can still provide valuable insights into driving habits that may indirectly affect claim amounts, such as the frequency of highway driving, its primary strength lies in its ability to improve the accuracy of claim frequency predictions by identifying risky driving behaviours.

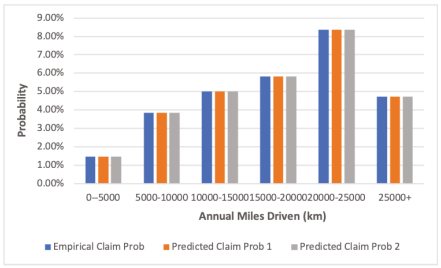
Figures 4 and 5 illustrate the comparison among empirical estimates, predictions made by the model using only traditional policy variables, and those obtained from the model incorporating both traditional policy variables and a reduced set of telematics data. Notably, we observe no significant deviation in either predicted claim frequency or predicted loss cost. This observation underscores the importance of including traditional policy variables when forecasting both claim counts and claim amounts. Given that telematics variables play a role in determining individual premiums and their predicted outcomes significantly impact the distribution of claim probabilities or claim amounts for traditional policy variables, we infer that predictive modelling using both types of variables surpasses alternative pricing strategies.



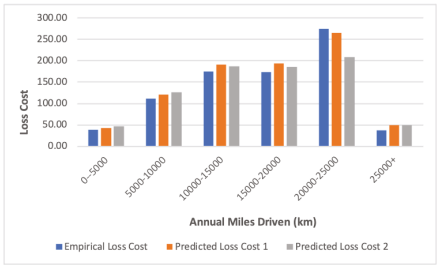
(**a**) Claim probability, years of no claims



(**b**) Loss cost, years of no claims

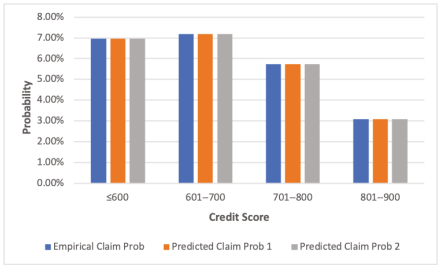


(**c**) Claim probability, annual mileage driven



(**d**) Loss cost, annual mileage driven

Figure 4. *Cont.*

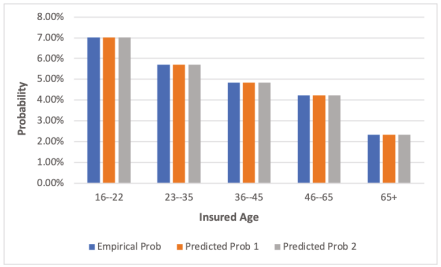


(**e**) Claim probability, credit scores

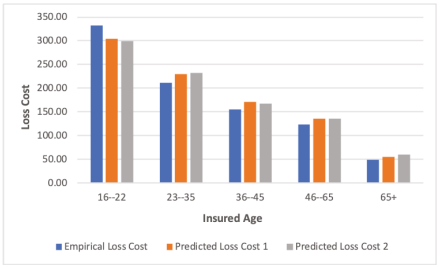


(**f**) Loss cost, credit scores

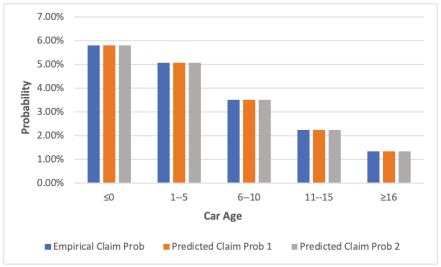
Figure 4. The predicted probability and loss cost by different levels of insured age, car age, and credit scores, obtained from the model using traditional variables only (labelled as 1 in the figure legend), traditional plus telematics variables (labelled as 2 in the figure legend), and compared to the empirical results.



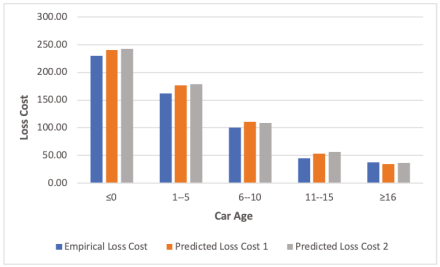
(**a**) Claim probability, insured age



(**b**) Loss cost, insured age



(**c**) Claim probability, car age



(**d**) Loss cost, car age

Figure 5. The predicted probability and loss cost by different levels of insured age and car age, obtained from the model using traditional variables only (labelled as 1 in the figure legend), traditional plus telematics variables (labelled as 2 in the figure legend), and compared to the empirical results.

Tables 3 and 4 present the estimated model coefficients and their associated standard deviations for both predictive models used in predicting claim probability and claim amounts. These models incorporate both traditional policy variables and a reduced set of telematics variables. Our analysis reveals that the inclusion of insured gender and marital status does not yield a significant impact on either claim probability or claim amount predictions. However, variables such as insured age, car age, credit score, location, annual miles driven, and years of no claims exhibit statistically significant effects when predicting claim frequency. Consequently, this set of variables is streamlined to car age, credit score, and location. This finding suggests that, while more traditional policy variables play a significant role in predicting claim frequency, claim amounts are predominantly influenced by car age, credit score, and location, aligning with common practices in auto insurance pricing, where the most influential variables are identified. Moreover, all telematics variables, except for right turn, demonstrate statistical significance. Additionally, there are instances where certain levels within a variable may not appear statistically significant compared to the base level. This observation hints at the possibility of further reducing the telematics variable set. However, given their manageable size, a further reduction

could potentially diminish the predictive power of the model. Thus, despite the potential for further refinement, maintaining the current set of telematics variables is advisable to preserve the model's effectiveness in predicting claim frequency and claim amounts.

After computing and comparing the risk relativities using loss cost based on the model coefficients obtained with and without the inclusion of telematics data, we present the findings in Figure 6 and Table 5. Our analysis reveals noteworthy implications regarding the impact of telematics variables on risk relativities. Surprisingly, the inclusion of telematics variables in the predictive models for claim probability and claim amount does not result in significant changes in loss cost relativities for most factor levels, with the exception of a few levels of location and the majority of levels of the annual miles driven variable.

Specifically, we observe a significant increase in risk relativity for most levels following the inclusion of telematics variables, particularly in relation to the usage of vehicles. This finding sheds light on the importance of vehicle usage patterns in determining risk profiles and underscores the value of telematics data in refining risk assessment methodologies. By revealing variations in risk relativities associated with different levels of location and annual miles driven, the study highlights the nuanced impact of telematics variables on insurance risk, offering insights that can inform more tailored pricing strategies and underwriting practices.

Table 3. Traditional variable coefficients and their standard errors obtained from the GLM model with both traditional policy variables and telematics variables.

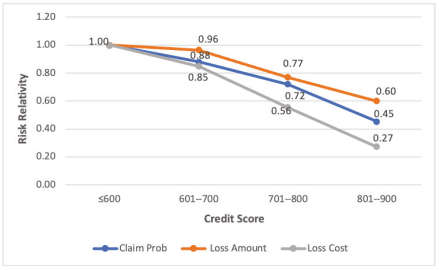
	Dependent Variable			
	Num_Claim	**Num_Claim**	**AMT_Claim**	**AMT_Claim**
	Poisson	*Poisson*	*glm: Gamma link = log*	*glm: Gamma link = log*
	(1) Coefficients	**(2) Standard Errors**	**(3) Coefficients**	**(4) Standard Errors**
Insured.AgeGroup23–35	−0.473 ***	(0.112)	−0.030	(0.140)
Insured.AgeGroup36–45	−0.527 ***	(0.116)	−0.222	(0.146)
Insured.AgeGroup46–65	−0.336 ***	(0.118)	−0.200	(0.147)
Insured.AgeGroup65+	−0.561 ***	(0.131)	−0.134	(0.164)
Insured.sexMale	0.045	(0.031)	0.010	(0.041)
Car.age.range1–5	−0.210 ***	(0.049)	−0.077	(0.064)
Car.age.range6–10	−0.489 ***	(0.053)	−0.263 ***	(0.070)
Car.age.range11–15	−0.781 ***	(0.076)	−0.546 ***	(0.098)
Car.age.range ≥16	−1.290 ***	(0.263)	−0.514	(0.323)
MaritalSingle	0.046	(0.035)	−0.006	(0.046)
Car.useCommute	−0.095	(0.081)	0.063	(0.107)
Car.useFarmer	−0.602 *	(0.234)	−0.805 ***	(0.293)
Car.usePrivate	−0.047	(0.086)	−0.073	(0.113)
Credit.score.range601–700	−0.127 *	(0.074)	−0.037	(0.095)
Credit.score.range701–800	−0.328 ***	(0.069)	−0.261 ***	(0.089)
Credit.score.range801–900	−0.791 ***	(0.068)	−0.511 ***	(0.089)
factor(Location.Cluster)2	0.032	(0.093)	−0.028	(0.122)
factor(Location.Cluster)3	−1.080 ***	(0.412)	−0.679	(0.554)
factor(Location.Cluster)4	0.057	(0.069)	0.238 ***	(0.091)
factor(Location.Cluster)5	0.298	(0.307)	−0.120	(0.395)
factor(Location.Cluster)6	−0.284 ***	(0.082)	−0.040	(0.107)
factor(Location.Cluster)7	−0.070	(0.111)	0.522 ***	(0.141)
factor(Location.Cluster)8	−0.801 *	(0.451)	1.397 *	(0.553)
factor(Location.Cluster)9	−0.099	(0.067)	0.009	(0.087)
factor(Location.Cluster)10	0.150 *	(0.073)	0.233 *	(0.095)
factor(Location.Cluster)11	0.096	(0.503)	1.678 ***	(0.617)
factor(Location.Cluster)12	−0.572 ***	(0.118)	0.118	(0.150)
factor(Location.Cluster)13	−0.163 *	(0.076)	0.068	(0.100)
factor(Location.Cluster)14	0.012	(0.066)	0.121	(0.086)
Annual.miles.drive.range5000–10,000	0.733 ***	(0.131)	0.041	(0.168)
Annual.miles.drive.range10,000–15,000	0.829 ***	(0.133)	0.159	(0.170)
Annual.miles.drive.range15,000–20,000	0.839 ***	(0.144)	0.126	(0.186)
Annual.miles.drive.range20,000–25,000	0.903 ***	(0.193)	0.018	(0.242)
Annual.miles.drive.range25,000+	0.299	(0.351)	−0.696	(0.472)
Years.noclaims.range21–40	−0.160 ***	(0.046)	0.045	(0.059)
Years.noclaims.range41–60	−0.136 *	(0.065)	−0.379 ***	(0.086)
Years.noclaims.range61–80	−0.279	(0.194)	−0.159	(0.243)

Note: * $p < 0.1$; *** $p < 0.01$.

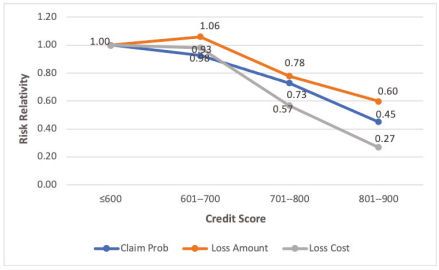
Table 4. Telematics variable coefficients and their standard errors obtained from the GLM model with both traditional policy variables and telematics variables.

	Dependent Variable			
	Num_Claim	**Num_Claim**	**AMT_Claim**	**AMT_Claim**
	Poisson	*Poisson*	*glm: Gamma link = log*	*glm: Gamma link = log*
	(1) Coefficients	**(2) Standard Errors**	**(3) Coefficients**	**(4) Standard Errors**
Annual.pct.driven	2.336 ***	(0.074)	−0.392 ***	(0.102)
Total.miles.driven	0.00003 ***	(0.00000)	−0.00001 ***	(0.00001)
Pct.drive1	0.392	(0.252)	−0.531	(0.437)
Pct.drive2	0.459 *	(0.199)	−0.439	(0.344)
Pct.drive3	0.478 *	(0.234)	−0.548	(0.410)
Pct.drive4	0.310 *	(0.158)	−0.300	(0.272)
Pct.drive.hrs	0.014 ***	(0.005)	−0.065 ***	(0.020)
Rush.ampm	−0.065 ***	(0.016)	−0.015	(0.021)
Avgdays.week	0.067 ***	(0.018)	−0.102 ***	(0.027)
Accel1	−0.133 ***	(0.039)	−0.054	(0.123)
Accel2	0.050 ***	(0.019)	0.027	(0.033)
Brake1	0.038	(0.030)	0.175	(0.135)
Brake2	0.088 ***	(0.011)	0.088 ***	(0.027)
Left.turn1	−0.030 *	(0.015)	0.045 *	(0.021)
Left.turn2	0.059 ***	(0.018)	−0.065 ***	(0.025)
Right.turn1	0.026	(0.020)	0.040	(0.060)
Right.turn2	−0.014	(0.027)	−0.046	(0.078)
Constant	−4.500 ***	(0.240)	9.502 ***	(0.316)
Observations	100,000		3864	
Log Likelihood	−15,608.270		−35,038.040	
Akaike Inf. Crit.	31,326.540		70,186.090	

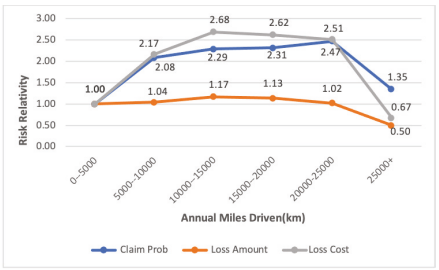
Note: * $p < 0.1$; *** $p < 0.01$.



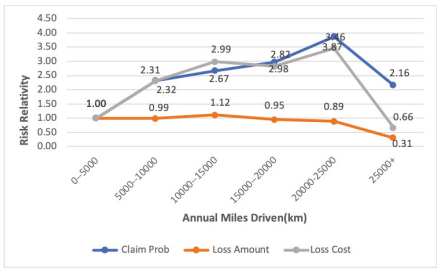
(**a**) With telematics variables



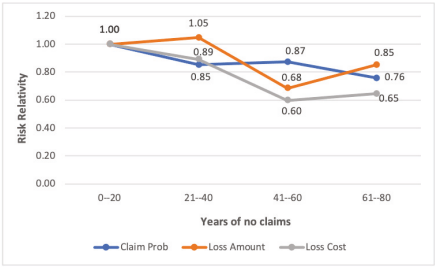
(**b**) Without telematics variables



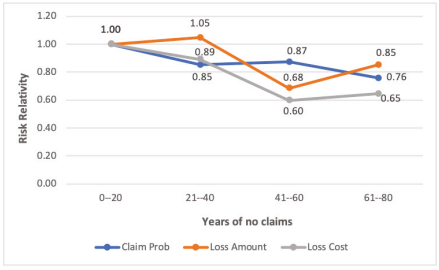
(**c**) With telematics variables



(**d**) Without telematics variable



(**e**) With telematics variable



(**f**) Without telematics variable

Figure 6. The comparison of estimated risk relativity for claim probability, claim amount and loss cost obtained from the predictive models with and without the use of telematics variables.

Table 5. Relativity estimate for the cases with and without telematics variables used in the predictive models for predicting the probability of claim occurrence and claim amounts. The overall relativity from both claim probability and claim amount in terms of loss cost are also obtained and compared. The bolded values in the table indicate more significant changes in relativities.

Variable	With Telematics Claim Prob (with Poisson)	Claim Amounts (with Gamma)	Loss Cost	Without Telematics Claim Prob (with Poisson)	Claim Amounts (with Gamma)	Loss Cost	Rel.Change
Insured.Age16–22	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Insured.Age23–35	0.62	0.97	0.60	0.69	0.91	0.63	−0.31
Insured.Age36–45	0.59	0.80	0.47	0.65	0.75	0.49	−0.28
Insured.Age46–65	0.71	0.82	0.58	0.71	0.79	0.57	−0.21
Insured.Age65+	0.57	0.87	0.50	0.53	0.89	0.47	−0.40
Female	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Male	1.05	1.01	1.06	0.99	1.03	1.03	0.02
Car.age$\leq$0	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Car.age1–5	0.81	0.93	0.75	0.91	0.91	0.83	−0.16
Car.age6–10	0.61	0.77	0.47	0.64	0.83	0.53	−0.36
Car.age11–15	0.46	0.58	0.27	0.41	0.64	0.26	−0.38
Car.age$\geq$16	0.28	0.60	0.16	0.24	0.66	0.16	−0.49
Married	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Single	1.05	0.99	1.04	1.03	0.99	1.03	0.05
Car.Use.Commercial	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Car.useCommute	0.91	1.06	0.97	0.81	1.05	0.85	−0.09
Car.useFarmer	0.55	0.45	0.24	0.42	0.44	0.18	−0.19
Car.usePrivate	0.95	0.93	0.89	0.81	0.96	0.77	−0.07
Credit.Score.$\leq$600	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Credit.score.range601–700	0.88	0.96	0.85	0.93	1.06	0.98	−0.21
Credit.score.range701–800	0.72	0.77	0.56	0.73	0.78	0.57	−0.22
Credit.score.range801–900	0.45	0.60	0.27	0.45	0.60	0.27	−0.33
Location.Cluster1	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Location.Cluster2	1.03	0.97	1.00	1.18	0.95	1.12	0.05
Location.Cluster3	0.34	0.51	0.17	0.35	0.45	0.16	−0.27
Location.Cluster4	1.06	1.27	1.34	1.23	1.36	1.67	−0.02
Location.Cluster5	1.35	0.89	1.20	1.48	0.95	1.40	0.25
Location.Cluster6	0.75	0.96	0.72	0.79	0.89	0.70	−0.17
Location.Cluster7	0.93	1.69	1.57	1.07	1.57	1.69	0.00
Location.Cluster8	0.45	4.04	1.81	0.49	3.55	1.74	**−1.73**
Location.Cluster9	0.91	1.01	0.91	1.00	0.99	1.00	−0.08
Location.Cluster10	1.16	1.26	1.47	1.20	1.26	1.51	0.20
Location.Cluster11	1.10	5.35	5.89	1.04	5.04	5.23	**0.85**
Location.Cluster12	0.56	1.13	0.64	0.60	1.07	0.65	−0.44
Location.Cluster13	0.85	1.07	0.91	0.93	1.03	0.96	−0.13
Location.Cluster14	1.01	1.13	1.14	1.11	1.21	1.34	−0.07
Annual.miles.0–5000	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Annual.miles.5000–10,000	2.08	1.04	2.17	2.32	0.99	2.31	**1.17**
Annual.miles.10,000–15,000	2.29	1.17	2.68	2.67	1.12	2.99	**1.57**
Annual.miles.15,000–20,000	2.31	1.13	2.62	2.98	0.95	2.82	**1.68**
Annual.miles.20,000–25,000	2.47	1.02	2.51	3.87	0.89	3.46	**1.62**
Annual.miles.25,000+	1.35	0.50	0.67	2.16	0.31	0.66	0.37
Years.noclaims.0–20	1.00	1.00	1.00	1.00	1.00	1.00	0.00
Years.noclaims21–40	0.85	1.05	0.89	0.85	1.04	0.88	−0.15
Years.noclaims41–60	0.87	0.68	0.60	0.79	0.66	0.52	−0.06
Years.noclaims61–80	0.76	0.85	0.65	0.60	0.79	0.48	−0.15

Overall, while the inclusion of telematics variables may not lead to substantial changes in loss cost relativities across most factor levels, the observed increase in risk relativity for certain variables underscores the significance of telematics data in enhancing risk assessment accuracy and refining insurance pricing models. These findings provide valuable insights for insurers seeking to leverage telematics technology to better understand and mitigate insurance risk, ultimately leading to more informed decision-making and improved competitiveness in the insurance market.

5. Some Discussions on Statistical Analysis

This study provides an in-depth exploration of the influence of telematics variables in conjunction with traditional rating variables within the framework of rate regulation. While some variables, such as gender and marital status, did not emerge as statistically significant in the model, their inclusion is crucial for assessing pricing fairness by estimating risk relativity across various demographic groups. From a regulatory perspective, retaining these variables is essential, as they ensure that risk relativity is adequately captured within the model. Although these variables may not strongly enhance predictive power, they play a critical role in helping maintain fairness in pricing, which is a central objective in rate regulation. It is important to distinguish between the objectives of rate regulation and those of insurance pricing models used by insurance companies. In pricing models, where numerous rating variables from diverse sources are employed for determining insurance premiums, variable selection is a key step in reducing redundancy and enhancing efficiency.

However, the goal of rate regulation is not merely to improve predictive performance but to establish benchmark values for regulatory risk factors. These benchmarks, often based on traditional policy variables selected through actuarial best practices, help ensure that pricing remains equitable and consistent across different demographic groups, or other criteria.

In this study, instead of focusing on variable selection for inclusion or exclusion, model selection criteria such as the Akaike Information Criterion (AIC) and log-likelihood were used to evaluate the impact of different model configurations, while keeping the same set of predictors. This approach allowed us to assess model performance without compromising the integrity of key variables that are essential for regulatory purposes.

In many European countries, the use of gender as a rating factor in insurance pricing is prohibited, and a similar approach is followed in Canada. However, in the Canadian context, gender and marital status are combined with age groups to form a rating factor known as 'Type of Use', which is widely and lawfully employed by auto insurance companies. From a rate regulation perspective, the inclusion of these variables is crucial as it allows for the derivation of their risk relativity.

Telematics variables may also correlate with traditional policy variables such as gender and marital status. For example, male drivers are often associated with higher speeds when making turns and may exhibit more aggressive driving behaviours than female drivers. These telematics variables offer significant promise for the regulation of UBI and represent a new frontier in understanding how driving pattern data can be integrated into insurance rate regulation. This study is one of the first attempts to investigate the impact of telematics variables on modelling claim frequency and severity specifically for rate regulation. The telematics variables employed in this study demonstrate considerable potential for regulating UBI.

6. Impact of Telematics Data on Pricing and Rate Regulation

The study of how telematics data affect pricing strategies and rate regulation framework from a risk and insurance perspective yields several significant impacts. To start, it enhances risk assessment outcomes by providing insurers and regulators with a more detailed understanding of individual driving behaviour. This enhancement allows for more accurate discrimination between high- and low-risk drivers, leading to more precise and fair pricing and underwriting decisions. Additionally, incorporating telematics data facilitates dynamic pricing models, such as usage-based or pay-as-you-drive insurance, where premiums are adjusted based on actual driving patterns. This approach promotes actuarial fairness in insurance pricing by aligning premiums charged more closely with individual risk profiles. Moreover, the integration of telematics data enables insurers to enhance loss prevention efforts. By identifying risky driving behaviours like speeding or harsh braking, insurers can proactively mitigate risks. Incentives or rewards for safe driving practices can be offered, ultimately reducing the frequency and severity of claims.

However, the use of telematics data in pricing and rate regulation also brings forth regulatory considerations. Regulators must balance the benefits of technology innovation and risk discrimination with consumer protection and fairness. This may involve developing guidelines or standards for the collection, usage, and protection of telematics data to ensure transparency and accountability in insurance practices. Furthermore, challenges and ethical considerations arise with the use of telematics data. Concerns about data privacy, consent, and potential biases in algorithmic decision-making may need to be addressed when implementing telematics-based pricing. Insurers and regulators must navigate these issues to maintain consumer trust and confidence in telematics-based insurance products.

On the other hand, the utilization of a predictive model incorporating telematics data for predicting claim counts and claim amounts can significantly impact the auto insurance industry from an actuarial fairness perspective. Such models enable insurers to enhance risk discrimination by analyzing individual driving behaviour. This precision allows for a more equitable distribution of premiums, aligning them more closely with the actual

risk posed by each insured driver. From an actuarial pricing perspective, it is expected that drivers should be responsible for their own risk. Consequently, actuarial fairness is bolstered as premiums become more reflective of expected claims costs, reducing the potential for cross-subsidization where lower-risk policyholders subsidize the premiums of their higher-risk counterparts. Moreover, telematics-based insurance models introduce incentives for safe driving practices, further reinforcing actuarial fairness. This approach not only benefits individual drivers but also contributes to fairer insurance pricing by aligning premiums with actual risk exposure. By providing insights into how premiums are calculated, policyholders can better understand and evaluate the fairness of pricing decisions, enhancing overall confidence in the insurance industry. Overall, while telematics-based predictive modelling holds promise for improving actuarial fairness in the auto insurance industry, it necessitates ongoing evaluation, refinement, and regulatory oversight to ensure equitable outcomes for all stakeholders.

7. Conclusions

Our comprehensive analysis of the results obtained through the proposed methodology sheds light on the effectiveness of leveraging telematics data in insurance analytics. The NSPCA approach employed in our study enabled us to distill meaningful insights from complex telematics data, enhancing predictive accuracy and interpretability, which is crucial for informed decision-making in the auto insurance industry. The findings from our analysis of telematics variables, such as daily driving patterns, extended driving durations, and driving intensities, provide a good understanding of driving behaviours. By categorizing driving habits into distinct groups based on days of the week and intensity levels, we gained valuable insights into the variability of risk profiles associated with different driving patterns. Additionally, our examination of various pricing strategies and their impact on traditional policy variables highlighted the complex interplay between telematics data and traditional risk factors.

The incorporation of telematics variables into the predictive modelling of claim counts and claim amounts could potentially lead to a more dynamic and personalized approach to premium calculation. The insight obtained from such predictive modelling enables insurers to differentiate between high and low-risk drivers more accurately than traditional rating factors like age, gender, and location alone. However, regulators may need to adapt existing frameworks to accommodate these advancements, ensuring that ratemaking practices remain fair and transparent for consumers. Our study revealed that, while the inclusion of telematics variables in predictive models may not lead to significant changes in loss cost relativities across most factor levels, it does offer valuable insights into the impact of vehicle usage patterns on insurance risk. The observed increase in risk relativity for certain variables underscores the importance of telematics data in refining risk assessment methodologies and informing more tailored pricing strategies.

Our study has unveiled that relying solely on telematics data for pricing could markedly alter the traditional rating variables commonly employed in current regulatory practices. Consequently, our research suggests that pricing insurance based solely on telematics data may not be advisable. Rather, we advocate for a pricing strategy that integrates telematics variables with existing rating factors. By incorporating these variables, insurers can achieve a more sustainable approach to discerning various levels of risk among drivers. This strategy ensures a balanced consideration of both traditional and telematics-derived factors, thereby fostering greater stability in insurance pricing practices. Our future work will focus on using advanced machine learning techniques, such as deep learning or ensemble methods, to develop more robust models capable of capturing the intricate relationships within telematics-derived factor and the response variables on which we focused. By refining these models, insurers can improve their accuracy and predictive power, leading to more informed decision-making in auto insurance pricing and rate regulation.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Mwongela, J.N. The Influence of Regulatory Framework on Insurance Penetration in Kenya. A Case Study of the Registered Insurance Companies in Nairobi County. Ph.D. Thesis, Kenya Methodist University, Nairobi, Kenya, 2022.
2. Fu, L.; Wu, C.S.P. General Iteration Algorithm for Classification Ratemaking. *Variance* **2007**, *1*, 193–213.
3. Branda, M. Optimization Approaches to Multiplicative Tariff of Rates Estimation in Non-life Insurance. *Asia-Pac. J. Oper. Res.* **2014**, *31*, 1450032. [CrossRef]
4. Masese, V.O. Application of generalized linear Models in Pricing Usage-Based Insurance. Ph.D. Thesis, University of Nairobi, Nairobi, Kenya, 2020.
5. Bian, Y.; Yang, C.; Zhao, J.L.; Liang, L. Good Drivers Pay Less: A Study of Usage-based Vehicle Insurance Models. *Transp. Res. Part A Policy Pract.* **2018**, *107*, 20–34. [CrossRef]
6. Che, X.; Liebenberg, A.; Xu, J. Usage-based Insurance—i=Impact on Insurers and Potential Implications for InsurTech. *N. Am. Actuar. J.* **2022**, *26*, 428–455. [CrossRef]
7. Śliwiński, A.; Kuryłowicz, Ł. Usage-based Insurance and Its Acceptance: An Empirical Approach. *Risk Manag. Insur. Rev.* **2021**, *24*, 71–91. [CrossRef]
8. Li, H.J.; Luo, X.G.; Zhang, Z.L.; Jiang, W.; Huang, S.W. Driving Risk Prevention in Usage-based Insurance Services Based on Interpretable Machine Learning and Telematics data. *Decis. Support Syst.* **2023**, 172, 113985. [CrossRef]
9. Cunha, L.; Bravo, J.M. Automobile Usage-Based-Insurance: Improving Risk Management Using Telematics Data. In Proceedings of the 2022 17th Iberian Conference on Information Systems and Technologies (CISTI), Madrid, Spain, 22–25 June 2022; IEEE: Piscatvey, NJ, USA, 2022; pp. 1–6.
10. Vavouranakis, P.; Panagiotakis, S.; Mastorakis, G.; Mavromoustakis, C.X. Smartphone-based telematics for usage based insurance. In *Advances in Mobile Cloud Computing and Big Data in the 5G Era*; Springer: Cham, Switzerland, 2017; pp. 309–339.
11. Guillen, M.; Nielsen, J.P.; Pérez-Marín, A.M. Near-miss Telematics in Motor Insurance. *J. Risk Insur.* **2021**, *88*, 569–589. [CrossRef]
12. El hafidy, A.; Rachad, T.; Idri, A.; Zellou, A. Gamified Mobile Applications for Improving Driving Behavior: A Systematic Mapping Study. *Mob. Inf. Syst.* **2021**, *2021*, 6677075. [CrossRef]
13. Coetzer, B.A. Usage-Based Insurance: Nudging towards Responsible Driving. Ph.D. Thesis, Stellenbosch University, Stellenbosch, South Africa, 2022.
14. Henckaerts, R.; Antonio, K. The Added Value of Dynamically Updating Motor Insurance Prices with Telematics Collected Driving Behavior Data. *Insur. Math. Econ.* **2022**, *105*, 79–95. [CrossRef]
15. Goniewicz, K.; Goniewicz, M.; Pawłowski, W.; Fiedor, P. Road Accident Rates: Strategies and Programmes for Improving Road Traffic Safety. *Eur. J. Trauma Emerg. Surg.* **2016**, *42*, 433–438. [CrossRef]
16. Elvik, R.; Vadeby, A.; Hels, T.; Van Schagen, I. Updated Estimates of the Relationship Between Speed and Road Safety at the Aggregate and Individual levels. *Accid. Anal. Prev.* **2019**, *123*, 114–122. [CrossRef] [PubMed]
17. Fan, C.K.; Wang, W.Y. A Comparison of Underwriting Decision Making Between Telematics-enabled UBI and Traditional Auto Insurance. *Adv. Manag. Appl. Econ.* **2017**, *7*, 17.
18. Yan, C.; Ou, Z.; Liu, W.; Xu, Q. Research on UBI Auto Insurance Pricing Model Based on Adaptive SAPSO to Optimize the Fuzzy Controller. *Int. J. Fuzzy Syst.* **2020**, *22*, 491–503. [CrossRef]
19. Holzapfel, J.; Peter, R.; Richter, A. Mitigating Moral Hazard with Usage-based Insurance. *J. Risk Insur.* **2023**. [CrossRef]
20. Eling, M.; Kraft, M. The Impact of Telematics on The Insurability of Risks. *J. Risk Financ.* **2020**, *21*, 77–109. [CrossRef]
21. Chaba, R. Influence of Telematics of UBI Insurance on The Management of The Fleet of Company Vehicles. *Arch. Motoryz.* **2021**, *92*, 69–82. [CrossRef]
22. Wiegers, W.A. The Use of Age, Sex, and Marital Status as Rating Variables in Automobile Insurance. *Univ. Tor. Law J.* **1989**, *39*, 149–210. [CrossRef]
23. Lemaire, J.; Park, S.C.; Wang, K.C. The Use of Annual Mileage as a Rating Variable. *Astin Bull. J. IAA* **2016**, *46*, 39–69. [CrossRef]
24. Verbelen, R.; Antonio, K.; Claeskens, G. Unravelling The Predictive Power of Telematics Data in Car Insurance Pricing. *J. R. Stat. Soc. Ser. C Appl. Stat.* **2018**, *67*, 1275–1304. [CrossRef]
25. Maillart, A. Toward an Explainable Machine Learning Model for Claim Frequency: A Use Case in Car Insurance Pricing with Telematics Data. *Eur. Actuar. J.* **2021**, *11*, 579–617. [CrossRef]
26. Duong, T.D.; Duong, V.N. Non-negative Sparse Principal Component Analysis for Multidimensional Constrained Optimization. In Proceedings of the PRICAI 2008: Trends in Artificial Intelligence: 10th Pacific Rim International Conference on Artificial Intelligence, Hanoi, Vietnam, 15–19 December 2008; Proceedings 10; Springer: Berlin/Heidelberg, Germany, 2008; pp. 103–114.
27. Duong, T.D. Non-Negative Sparse Principal Component Analysis. *J. Tech. Educ. Sci.* **2009**, *4*, 31–38.
28. Guillen, M.; Nielsen, J.P.; Ayuso, M.; Pérez-Marín, A.M. The Use of Telematics Devices to Improve Automobile Insurance Rates. *Risk Anal.* **2019**, *39*, 662–672. [CrossRef] [PubMed]

29. Duval, F.; Boucher, J.P.; Pigeon, M. Enhancing Claim Classification with Feature Extraction from Anomaly-detection-derived Routine and Peculiarity Profiles. *J. Risk Insur.* **2023**, *90*, 421–458. [CrossRef]
30. Jiang, Q.; Shi, T. Auto Insurance Pricing Using Telematics Data: Application of a Hidden Markov Model. *N. Am. Actuar. J.* **2024**, 1–18. [CrossRef]
31. Chauhan, V.; Yadav, J. Bibliometric Review of Telematics-based Automobile Insurance: Mapping the Landscape of Research and Knowledge. *Accid. Anal. Prev.* **2024**, *196*, 107428. [CrossRef]
32. Hosein, P. A Data Science Approach to Risk Assessment for Automobile Insurance Policies. *Int. J. Data Sci. Anal.* **2024**, *17*, 127–138. [CrossRef]
33. Xiang, J.; Ghaffarpasand, O.; Pope, F.D. Mapping Urban Mobility Using Vehicle Telematics to Understand Driving Behaviour. *Sci. Rep.* **2024**, *14*, 3271. [CrossRef]
34. Ayuso, M.; Guillen, M.; Nielsen, J.P. Improving Automobile Insurance Ratemaking Using Telematics: Incorporating Mileage and Driver Behaviour Data. *Transportation* **2019**, *46*, 735–752. [CrossRef]
35. Ma, Y.L.; Zhu, X.; Hu, X.; Chiu, Y.C. The Use of Context-sensitive Insurance Telematics Data in Auto Insurance Rate Making. *Transp. Res. Part Policy Pract.* **2018**, *113*, 243–258. [CrossRef]
36. Gao, G.; Meng, S.; Wüthrich, M.V. Claims Frequency Modeling Using Telematics Car Driving Data. *Scand. Actuar. J.* **2019**, *2019*, 143–162. [CrossRef]
37. Weidner, W.; Transchel, F.W.; Weidner, R. Telematic Driving Profile Classification in Car Insurance Pricing. *Ann. Actuar. Sci.* **2017**, *11*, 213–236. [CrossRef]
38. Soleymanian, M.; Weinberg, C.B.; Zhu, T. Sensor Data and Behavioral Tracking: Does Usage-based Auto Insurance Benefit Drivers? *Mark. Sci.* **2019**, *38*, 21–43. [CrossRef]
39. So, B.; Boucher, J.P.; Valdez, E.A. Synthetic Dataset Generation of Driver Telematics. *Risks* **2021**, *9*, 58. [CrossRef]
40. Sigg, C.D.; Buhmann, J.M. Expectation-maximization for Sparse and Non-negative PCA. In Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 5–9 July 2008; pp. 960–967.
41. Casalino, G.; Del Buono, N.; Mencar, C. Nonnegative Matrix Factorizations for Intelligent Data Analysis. In *Non-Negative Matrix Factorization Techniques: Advances in Theory and Applications*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 49–74.
42. Bertsimas, D.; Cory-Wright, R.; Pauphilet, J. Solving Large-scale Sparse PCA to Certifiable (near) Optimality. *J. Mach. Learn. Res.* **2022**, *23*, 1–35.
43. Urzola, R.G. Jack of all Trades, Master of None: The Trade-offs in Sparse PCA Methods for Diverse Purposes. Ph.D. Thesis, Proefschrift AIO, Hulsberg, The Netherlands, 2023.
44. Forzani, L.; Arancibia, R.G.; Llop, P.; Tomassi, D. Supervised Dimension Reduction for Ordinal Predictors. *Comput. Stat. Data Anal.* **2018**, *125*, 136–155. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Cauchy–Logistic Unit Distribution: Properties and Application in Modeling Data Extremes

Vladica S. Stojanović [1,*], Tanja Jovanović Spasojević [2], Radica Bojičić [3], Brankica Pažun [4] and Zlatko Langović [5]

[1] Department of Informatics & Computer Sciences, University of Criminal Investigation and Police Studies, 11000 Belgrade, Serbia
[2] Department of Mathematics, Faculty of Sciences & Mathematics, University of Priština in Kosovska Mitrovica, 38220 Kosovska Mutrovica, Serbia; tanja.jovanovic@pr.ac.rs
[3] Department of Mathematics & Informatics, Faculty of Economics, University of Kosovska Mitrovica, 38220 Kosovska Mitrovica, Serbia; radica.bojicic@pr.ac.rs
[4] Department of Informatics, Mathematics and Statistics, Faculty of Engineering Management, 11000 Belgrade, Serbia; brankica.pazun@fim.rs
[5] Department of Business Economy, Faculty of Hotel Management and Tourism, University of Kragujevac, 36210 Vrnjačka Banja, Serbia; zlatko.langovic@kg.edu.rs
* Correspondence: vladica.stojanovic@kpu.edu.rs

Abstract: This manuscript deals with a novel two-parameter stochastic distribution, obtained by transforming the Cauchy distribution, using generalized logistic mapping, into a unit interval. In this way, according to the well-known properties of the Cauchy distribution, a unit random variable with significantly accentuated values at the ends of the unit interval is obtained. Therefore, the proposed stochastic distribution, named the Cauchy–logistic unit distribution, represents a stochastic model that may be suitable for modeling phenomena and processes with emphasized extreme values. Key stochastic properties of the CLU distribution are examined, such as moments, entropy, modality, and symmetry conditions. In addition, a quantile-based parameter estimation procedure, an asymptotic analysis of the thus obtained estimators, and their Monte Carlo simulation study are conducted. Finally, the application of the proposed distribution in stochastic modeling of some real-world data with emphasized extreme values is provided.

Keywords: unit distributions; Cauchy distribution; generalized logistic map; stochastic properties; parameter estimation; quantiles; application

MSC: 60E05; 62E10; 62F10

1. Introduction

One of the important tools of contemporary probability theory is the so-called unit stochastic distributions, primarily used as stochastic models that describe proportional (i.e., percentage-based) variables. A proportional analysis is often used in various applied fields, such as econometrics, medicine, and engineering, where random variables (RVs) with a distribution defined on a unit interval are used to model the behavior of some real-world phenomena (see, as the most recent example, [1–9]). At the same time, due to the limitation of data within the $(0,1)$ interval, the modeling procedure itself is significantly different from the usual, traditional stochastic modeling. A typical approach to the construction of these stochastic distributions is usually based on the application of continuously differentiable

transformations from distributions defined on infinite intervals to unit intervals. Therefore, various transformations of continuous distributions into a unit interval are applied, such as Log-logistic distribution [10], Gompertz distribution [11], Teisser distribution [12], inverse Teisser distribution [13], Burr Type-X distribution [14], Maxwell–Boltzmann distribution [15], or unit distributions close to the exponential one [16,17]. In addition, recently, semi-logistic mappings have been frequently used in these types of transformations [18,19].

On the other hand, most unit distributions are limited, for instance, by only one form of (a)symmetry, or the property of "vanishing" data at the ends of the unit interval, which is inconvenient when it comes to phenomena with emphasized extreme values. To this end, a new unit distribution, called the Cauchy–logistic unit (CLU) distribution, is proposed here. Using similar ideas as in [20,21], the CLU distribution is based on a general logistic mapping of the Cauchy distribution to the unit interval. As is known, the Cauchy distribution has ″heavy tails", so the CLU distribution could be used in modeling empirical data on a unit interval with emphasized extremes, especially at the ends of the unit interval. In this way, it can be an alternative model to well-known unitary distributions such as Beta-type distributions [22], cumulative distribution function-quantile (CDF-Q) distributions [23,24], or the recently proposed, bounded truncated Cauchy power exponential (BTCPE) distribution [25].

The procedure for obtaining the CLU distribution is presented in the next Section 2, along with its key stochastic properties, such as moment-based features, (a)symmetry and modality conditions, entropy, hazard rates, quantile functions, etc. Section 3 discusses the methods of estimating the parameters of the CLU distribution, using the order and quantile statistics, as well as the numerical study and asymptotic properties of the estimators thus obtained. An application of the CLU distribution in fitting some real-world data with emphasized extremes is presented in Section 4, where it is also compared with some of the previously known unit stochastic distributions. Finally, Section 5 contains some concluding remarks and highlights.

2. Cauchy–Logistic Unit Distribution. Definition and Main Properties

We start from the random variable (RV) Y with a symmetric Cauchy distribution, defined by the probability density function (PDF):

$$g(y;\theta) = \frac{\theta}{\pi(\theta^2 + y^2)}, \qquad -\infty < y < +\infty, \tag{1}$$

where $\theta > 0$ is the scale parameter. For some $\lambda > 0$, let us define the so-called generalized logistic map $x = \varphi(y;\lambda) = (1 + \exp(y))^{-1/\lambda}$, which is an obvious bijective, continuous transformation $\varphi : (-\infty, +\infty) \times (0, +\infty) \to (0, 1)$, with the following limits:

$$\lim_{y \to -\infty} \varphi(y;\lambda) = 1^-, \qquad \lim_{y \to +\infty} \varphi(y;\lambda) = 0^+.$$

According to Equation (1) and the inverse transformation $\varphi^{-1}(x;\lambda) = \ln(x^{-\lambda} - 1)$, the PDF of the new RV X, after some computations, is obtained as follows:

$$f(x;\theta,\lambda) = g\Big(\varphi^{-1}(x;\lambda);\theta\Big) \cdot \left|\frac{\partial \varphi^{-1}(x;\lambda)}{\partial x}\right| = \frac{\lambda\theta}{\pi x\big(1 - x^{\lambda}\big)\big(\theta^2 + \ln^2\big(x^{-\lambda} - 1\big)\big)}, \tag{2}$$

where $x \in (0, 1)$ and $\int_0^1 f(x;\theta,\lambda)\mathrm{d}x = 1$. Therefore, we say that the RV X, whose PDF is given by Equation (2), has a Cauchy–logistic unit (CLU) distribution, with the parameters $\theta, \lambda > 0$, or abbr. $X : \mathcal{C}(\theta,\lambda)$. Based on that, it is obvious that the CLU distribution is a

two-parameter distribution, where, in addition to the scale parameter $\theta > 0$, there is also a shape parameter $\lambda > 0$. Due to them, the CLU distribution has various forms, asymmetries, and flexibilities, as shown in Figure 1a. Furthermore, it is easy to verify that the following boundary conditions hold:

$$\lim_{x \to 0^+} f(x;\theta,\lambda) = \lim_{x \to 1^-} f(x;\theta,\lambda) = +\infty,$$

that is, the CLU distribution can be suitable for modeling phenomena with accentuated extremes at the ends of the unit interval.

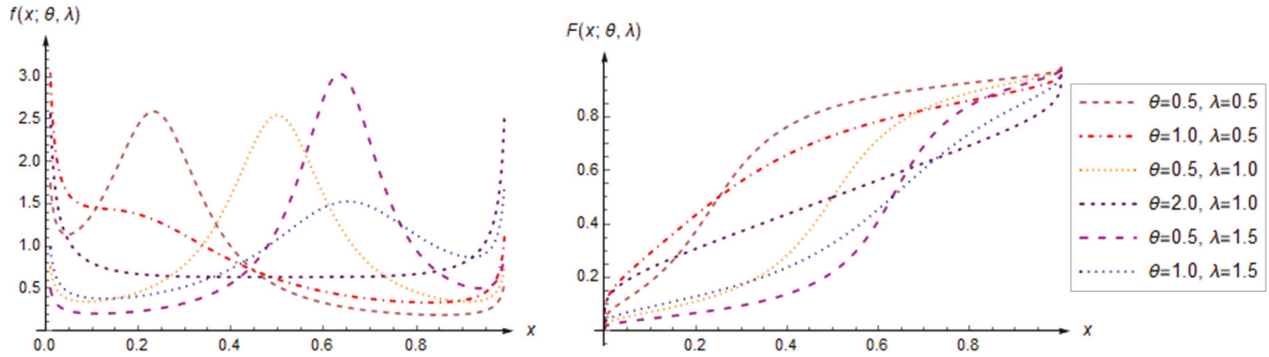
Additionally, the right plots (Figure 1b) show the cumulative distribution function (CDF) of the CLU distribution, obtained from Equation (1), after some computations, as follows:

$$F(x;\theta,\lambda) = P\{X < x\} = \int_0^x f(u;\theta,\lambda)\mathrm{d}u = \frac{1}{2} - \frac{1}{\pi}\arctan\left(\frac{\ln\left(x^{-\lambda}-1\right)}{\theta}\right). \tag{3}$$

Here, $x \in (0,1)$, and function $F(x;\theta,\lambda)$ is well defined outside the unit interval, since for all parameter values $\theta, \lambda > 0$ is valid:

$$\lim_{x \to 0^+} F(x;\theta,\lambda) = 0^+, \qquad \lim_{x \to 1^-} F(x;\theta,\lambda) = 1^-.$$

Based on the above, some key features of the CLU distribution are discussed below.



(**a**) (**b**)

Figure 1. Plots of the PDFs $f(x;\theta,\lambda)$ (**a**) and CDFs $F(x;\theta,\lambda)$ (**b**) of the CLU distribution for some values of parameters $\theta, \lambda > 0$.

2.1. Moment-Based Features

Firstly, the PDF and CDF of the CLU distribution can be used to obtain the rth incomplete moments:

$$M_r(x;\theta,\lambda) := \int_0^x u^r f(u;\theta,\lambda)\mathrm{d}u = \int_0^x u^r \mathrm{d}F(u;\theta,\lambda), \tag{4}$$

where r is an integer and $x \in (0,1)$. According to

$$0 < M_r(x;\theta,\lambda) \leq \int_0^x f(u;\theta,\lambda)\mathrm{d}u \leq 1,$$

the integral in Equation (4) converges, but it cannot be expressed in a closed form. The following statement gives a series expansion for incomplete moments, suitable for its numerical approximation.

Theorem 1. *The rth incomplete moment of the CLU distributed RV X can be expressed in the following way:*

$$M_r(x;\theta,\lambda) = \begin{cases} U(x;\theta,\lambda) + \dfrac{\theta r}{\lambda\pi}\displaystyle\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j} W_1\left(\theta\left(j+\dfrac{r}{\lambda}\right), \dfrac{\ln(x^{-\lambda}-1)}{\theta}\right), & 0 < x \le 2^{-1/\lambda} \\ U(x;\theta,\lambda) + \dfrac{\theta r}{\lambda\pi}\displaystyle\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}\left[W_1\left(\theta\left(j+\dfrac{r}{\lambda}\right),0\right) - W_2\left(\theta(j+1), -\dfrac{\ln(x^{-\lambda}-1)}{\theta}\right)\right], & x > 2^{-1/\lambda} \end{cases} \tag{5}$$

where

$$U(x;\theta,\lambda) := -\frac{x^r}{\pi}\arctan\left(\frac{\ln(x^{-\lambda}-1)}{\theta}\right),$$

and

$$W_1(a,b) := \int_b^{+\infty} \mathrm{e}^{-at}\arctan(t)\mathrm{d}t, \quad W_2(a,b) := \int_0^b \mathrm{e}^{-at}\arctan(t)\mathrm{d}t, \tag{6}$$

with $a > 0$ and $b \ge 0$.

Proof. Applying integration by parts in Equation (4), and then changing the variable $t = \ln(x^{-\lambda}-1)/\theta$, we obtain

$$\begin{aligned} M_r(x;\theta,\lambda) &= x^r F(x;\theta,\lambda) - r\int_0^x u^{r-1}F(u;\theta,\lambda)\mathrm{d}u \\ &= U(x;\theta,\lambda) + \frac{r}{\pi}\int_0^x u^{r-1}\arctan\left(\frac{\ln\left(u^{-\lambda}-1\right)}{\theta}\right)\mathrm{d}u \\ &= U(x;\theta,\lambda) + \frac{\theta r}{\lambda\pi}\int_{\theta^{-1}\ln\left(x^{-\lambda}-1\right)}^{+\infty} \mathrm{e}^{\theta t}\left(1+\mathrm{e}^{\theta t}\right)^{-r/\lambda-1}\arctan(t)\mathrm{d}t. \\ &= U(x;\theta,\lambda) + \frac{\theta r}{\lambda\pi}\int_{\theta^{-1}\ln\left(x^{-\lambda}-1\right)}^{+\infty} \mathrm{e}^{-\theta rt/\lambda}\left(1+\mathrm{e}^{-\theta t}\right)^{-r/\lambda-1}\arctan(t)\mathrm{d}t. \end{aligned} \tag{7}$$

Now, the following two cases can be observed:

(i) When $\theta^{-1}\ln\left(x^{-\lambda}-1\right) \ge 0$, or equivalently $0 < x \le 2^{-1/\lambda}$, we have $t \ge 0$, that is, $0 < \mathrm{e}^{-\theta t} \le 1$. Thus, by applying the generalized binomial formula in the last integral in Equation (7), one obtains

$$M_r(x;\theta,\lambda) = U(x;\theta,\lambda) + \frac{\theta r}{\lambda\pi}\int_{\theta^{-1}\ln\left(x^{-\lambda}-1\right)}^{+\infty}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}\mathrm{e}^{-\theta(j+r/\lambda)t}\arctan(t)\mathrm{d}t. \tag{8}$$

At the same time, for some fixed and positive $x \le 2^{-1/\lambda}$, i.e., $x^{-\lambda}-1 \ge 1$, the last sum and integral are absolutely convergent and are valid:

$$\begin{aligned} \sum_{j=0}^{+\infty}\int_{\theta^{-1}\ln(x^{-\lambda}-1)}^{+\infty}\left|\binom{-r/\lambda-1}{j}\mathrm{e}^{-\theta(j+r/\lambda)t}\arctan(t)\mathrm{d}t\right| &\le \frac{\pi}{2}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}\int_{\theta^{-1}\ln(x^{-\lambda}-1)}^{+\infty}\mathrm{e}^{-\theta(j+r/\lambda)t}\mathrm{d}t \\ &\le \frac{\pi\lambda}{2r\theta}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}(x^{-\lambda}-1)^{-(j+r/\lambda)} \\ &\le \frac{\pi\lambda}{r\theta\,2^{r/\lambda+2}} < +\infty. \end{aligned}$$

Therefore, the sign of the integral and the sign of the sum in Equation (8) can be inverted (see, e.g., Theorem 1.38 in [26]), after which it obviously gives the first expression in Equation (5).

(ii) In the second case, when $\theta^{-1}\ln\left(x^{-\lambda}-1\right) < 0$, that is, $2^{-1/\lambda} < x < 1$, using a similar procedure as above, we have

$$M_r(x;\theta,\lambda) = U(x;\theta,\lambda) + \frac{\theta r}{\lambda\pi}\int_0^{+\infty} \mathrm{e}^{-\theta r t/\lambda}\left(1+\mathrm{e}^{-\theta t}\right)^{-r/\lambda-1}\arctan(t)\mathrm{d}t + \frac{\theta r}{\lambda\pi}\int_{\theta^{-1}\ln\left(x^{-\lambda}-1\right)}^{0} \mathrm{e}^{\theta t}\left(1+\mathrm{e}^{\theta t}\right)^{-r/\lambda-1}\arctan(t)\mathrm{d}t,$$

that is, after substitution $z=-t$,

$$M_r(x;\theta,\lambda) = U(x;\theta,\lambda) + \frac{\theta r}{\lambda\pi}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j} W_1\left(\theta\left(j+\frac{r}{\lambda}\right),0\right) - \frac{\theta r}{\lambda\pi}\int_0^{-\theta^{-1}\ln\left(x^{-\lambda}-1\right)}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}\mathrm{e}^{-\theta(j+1)z}\arctan(z)\mathrm{d}t. \tag{9}$$

Similarly to before, for some fixed $x\in(2^{-1/\lambda},1)$, i.e., $0<x^{-\lambda}-1<1$, the following applies:

$$\begin{aligned}\sum_{j=0}^{+\infty}\int_0^{-\theta^{-1}\ln(x^{-\lambda}-1)}\left|\binom{-r/\lambda-1}{j}\mathrm{e}^{-\theta(j+1)z}\arctan(z)\mathrm{d}z\right| &\le \frac{\pi}{2}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}\int_0^{\theta^{-1}\ln(x^{-\lambda}-1)}\mathrm{e}^{-\theta(j+1)z}\mathrm{d}z\\ &\le \frac{\pi}{2\theta}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}\left(1-(x^{-\lambda}-1)^{j+1}\right)\\ &\le \frac{\pi}{\theta\,2^{r/\lambda+2}} < +\infty.\end{aligned}$$

Thus, by changing the order of integral and sum in the last summand of Equation (9), the second equality in Equations (5) follows. □

Remark 1. *Note that to apply the procedure for obtaining incomplete moments, described in the previous theorem, it is necessary to calculate the integrals defined by Equations* (6)*. This can be achieved after some algebraic calculation, which yields the following equality for the corresponding indefinite integral:*

$$\int \mathrm{e}^{-ax}\arctan(x)\,\mathrm{d}x = -\frac{\mathrm{i}\mathrm{e}^{-\mathrm{i}a}\mathrm{Ei}(-a(x-\mathrm{i})) - \mathrm{i}\mathrm{e}^{\mathrm{i}a}\mathrm{Ei}(-a(x+\mathrm{i})) + 2\mathrm{e}^{-ax}\arctan(x)}{2a} + C,$$

where $\mathrm{i}=\sqrt{-1}$ *and* $\mathrm{Ei}(z) = -\int_{-z}^{\infty}\mathrm{e}^{-t}/t\,\mathrm{d}t$ *is the exponential integral. At the same time, the above indefinite integral is real-valued, because:*

$$\mathrm{i}\mathrm{e}^{-\mathrm{i}a}\mathrm{Ei}(-a(x-\mathrm{i})) - \mathrm{i}\mathrm{e}^{\mathrm{i}a}\mathrm{Ei}(-a(x+\mathrm{i})) = 2\mathrm{Im}\left(\mathrm{e}^{\mathrm{i}a}\mathrm{Ei}(-a(x+\mathrm{i}))\right).$$

According to Theorem 1, one can easily obtain the rth moments of the CLU distribution:

$$\mu_r(\theta,\lambda) := E(X^r) = \int_0^1 x^r f(x;\theta,\lambda)\mathrm{d}x = \int_0^1 x^r \mathrm{d}F(x;\theta,\lambda). \tag{10}$$

They can be viewed as limiting cases of incomplete moments, when $x \to 1$, and similarly as in [21], using the second one of Equations (5), it immediately follows:

Corollary 1. *The rth moment of the CLU of the distributed RV X can be expressed as*

$$\mu_r(\theta,\lambda) = \frac{1}{2} + \frac{\theta r}{\lambda\pi}\sum_{j=0}^{+\infty}\binom{-r/\lambda-1}{j}\Big[\omega_{j+r/\lambda}(\theta) - \omega_{j+1}(\theta)\Big], \tag{11}$$

where:

$$\omega_j(\theta) := \int_0^{+\infty} \mathrm{e}^{-j\theta t}\arctan(t)\mathrm{d}t = \frac{\mathrm{Ci}(j\theta)\sin(j\theta) + \cos(j\theta)(\pi/2 - \mathrm{Si}(j\theta))}{j},$$

and $\mathrm{Ci}(z) = -\int_z^{\infty}\cos(t)/t\,\mathrm{d}t$ *and* $\mathrm{Si}(z) = \int_0^z \sin(t)/t\,\mathrm{d}t$ *are the cosine and sine integrals, respectively.*

Remark 2. *Note that Equation* (11) *can be used for (approximate) calculation of the mean value* $E(X) = \mu_1$ *and the variance* $\mathrm{Var}(X) = \mu_2 - \mu_1^2$. *For instance, when* $\theta = \lambda = 1$, *it is easy to obtain the (exact) value of the mean* $E(X) = 1/2$, *as well as the approximate value of the variance (calculated by summing* 10^6 *terms in Equation* (11)) $\mathrm{Var}(X) \approx 0.091926$. *Similarly, with arbitrary precision, one can calculate the skewness and the kurtosis of the CLU distributed RV X, respectively,*

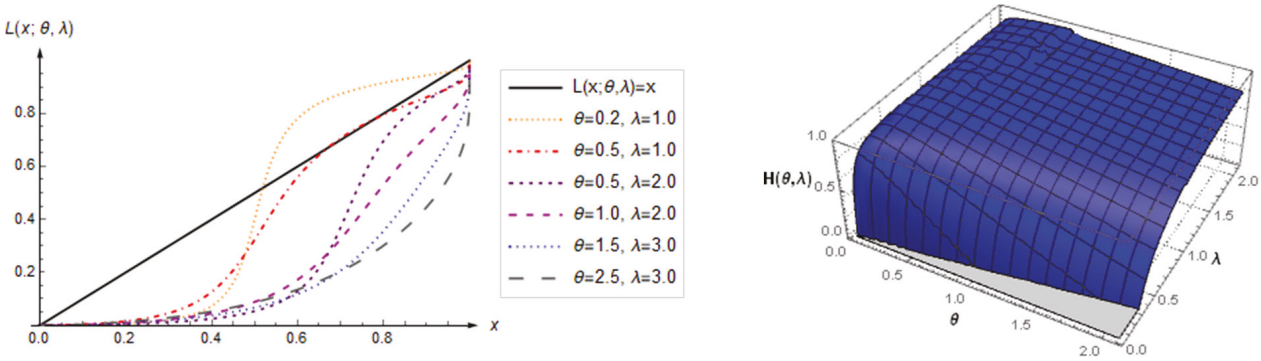
$$S(\theta,\lambda) = \frac{E(X-\mu_1(\theta,\lambda))^3}{[\mathrm{Var}(X)]^{3/2}} = \frac{\mu_3(\theta,\lambda) - 3\mu_1(\theta,\lambda)\mathrm{Var}(X) - (\mu_1(\theta,\lambda))^3}{[\mathrm{Var}(X)]^{3/2}},$$

$$K(\theta,\lambda) = \frac{E(X-\mu_1(\theta,\lambda))^4}{[\mathrm{Var}(X)]^{2}} = \frac{\mu_4(\theta,\lambda) - 4\mu_3(\theta,\lambda)\mu_1 + 6\mu_2(\theta,\lambda)(\mu_1(\theta,\lambda))^2 - 3(\mu_1(\theta,\lambda))^4}{[\mathrm{Var}(X)]^{2}}.$$

Remark 3. *The ratio of the incomplete and ordinary first-order moments defines the so-called Lorenz curve:*

$$L(x;\theta,\lambda) := \frac{M_1(x;\theta,\lambda)}{\mu_1(\theta,\lambda)} = \frac{1}{\mu_1(\theta,\lambda)}\int_0^x u f(u;\theta,\lambda)\mathrm{d}u,$$

which is often used as a tool to measure income inequality. Figure 2a shows the plots of Lorenz curves for some values of the parameters of the CLU distribution, with those "closer" to the diagonal line indicating a more even distribution of income, and vice versa. As can be seen, the CLU distribution shows the ability to describe significant inequality in the distribution of income, and this is in accordance with its definition, i.e., with the possibility of describing extreme data values.



(**a**) (**b**)

Figure 2. (**a**) Lorenz curves of the CLU distribution obtained for some parameter values; (**b**) 3D plot of the parameter dependence of Shannon entropy.

The entropy of the CLU distribution, as an important measure of uncertainty, is briefly discussed below. One of the frequently used, generalized ways of defining the entropy of RVs is the so-called Rényi entropy, given by the following equation:

$$H_\alpha(X) := \frac{1}{1-\alpha} \ln \int_0^1 (f(x;\theta,\lambda))^\alpha \mathrm{d}x,$$

where $\alpha \in (0, +\infty)$ and $\alpha \neq 1$. According to the definition of the CLU distributed RV $X : \mathcal{C}(\theta, \lambda)$, after some algebraic calculations similar to the proof of Theorem 1, one obtains

$$H_\alpha(X) := \ln\left(\frac{\theta}{\lambda}\right) - \frac{\alpha \ln \pi}{1-\alpha} + \frac{1}{1-\alpha} \ln G(\alpha;\theta,\lambda),$$

where

$$G(\alpha;\theta,\lambda) := \sum_{j=0}^{+\infty} \binom{(1+1/\lambda)(\alpha-1)}{j} \left(\int_0^{+\infty} \frac{\mathrm{e}^{-\theta t\left(j+\frac{1-\alpha}{\lambda}\right)} \mathrm{d}t}{(1+t^2)^\alpha} + \int_0^{+\infty} \frac{\mathrm{e}^{-\theta z(j+1-\alpha)} \mathrm{d}z}{(1+z^2)^\alpha} \right).$$

Based on the above, the well-known Shannon entropy can be obtained as follows:

$$\begin{aligned} H(X) := \mathrm{E}(-\ln f(X)) &= \lim_{\alpha \to 1} H_\alpha(X) = \ln\left(\frac{\theta}{\lambda\pi}\right) + \lim_{\alpha \to 1} \frac{1}{1-\alpha} \ln\left(\frac{1}{\pi} G(\alpha;\theta,\lambda)\right) \\ &= \ln\left(\frac{\theta}{\lambda\pi}\right) - \frac{1}{\pi} \cdot \frac{\partial\, G(\alpha;\theta,\lambda)}{\partial\, \alpha}\bigg|_{\alpha=1}, \end{aligned}$$

where $G(1;\theta,\lambda) = \int_{-\infty}^{+\infty} \mathrm{d}t/(1+t^2) = \pi$, and the listed partial derivative exists but is omitted here due to its complexity. In Figure 2b, a 3D plot of the Shannon entropy is shown, in dependence on parameters (θ, λ). Note that high values of Shannon entropy are obtained even with a slight increase in the shape parameter $\lambda > 0$. This (again) corresponds to the interpretation of the CLU distribution as a unit distribution with pronounced extremes.

2.2. Hazard Rate and Quantile Function

In this part, two more important functions that can further characterize the properties of the CLU distribution are considered. First, we introduce the hazard rate function (HRF), which is easily obtained according to Equations (2) and (3), as follows:

$$\begin{aligned} h(x;\theta,\lambda) &:= \frac{f(x;\theta,\lambda)}{1-F(x;\theta,\lambda)} \\ &= \frac{\theta\lambda}{x\left(1-x^\lambda\right)\left(\theta^2 + \ln^2\left(x^{-\lambda}-1\right)\right)\left(\frac{\pi}{2} + \arctan\left(\frac{\ln\left(x^{-\lambda}-1\right)}{\theta}\right)\right)}. \end{aligned} \tag{12}$$

Similar to the PDF of the CLU distribution, given by Equation (2), one can easily see that the function $h(x;\theta,\lambda)$ satisfies the following boundary conditions:

$$\lim_{x \to 0^+} h(x;\theta,\lambda) = \lim_{x \to 1^-} h(x;\theta,\lambda) = +\infty.$$

This means that the HRF has two asymptotic points at $x = 0$ and $x = 1$, i.e., in addition to the usual tendency to increase with increasing values of $x \in (0, 1)$, there is also the so-called decreasing failure rate (DFR). It describes the phenomenon where the probability of failure decreases in a unit interval, with the possibility that in some cases a sudden jump in the probability of failure is possible, as can be seen in Figure 3.

Additionally, after some simple algebraic calculations, the cumulative hazard rate function (CHRF), reversed hazard rate function (RHRF), and odds function (Odds) can be obtained, respectively, as follows:

$$G(x;\theta,\lambda) := \int_0^x h(u;\theta,\lambda)\mathrm{d}u = -\ln\left(1-F(x;\theta,\lambda)\right) = -\ln\left(\frac{1}{2}+\frac{1}{\pi}\arctan\left(\frac{\ln\left(x^{-\lambda}-1\right)}{\theta}\right)\right)$$

$$r(x;\theta,\lambda) := \frac{f(x;\theta,\lambda)}{F(x;\theta,\lambda)} = \frac{\theta\lambda}{x\left(1-x^{\lambda}\right)\left(\theta^2+\ln^2\left(x^{-\lambda}-1\right)\right)\left(\frac{\pi}{2}-\arctan\left(\frac{\ln\left(x^{-\lambda}-1\right)}{\theta}\right)\right)}$$

$$d(x;\theta,\lambda) := \frac{1-F(x;\theta,\lambda)}{F(x;\theta,\lambda)} = \frac{\pi}{\frac{\pi}{2}-\arctan\left(\frac{\ln\left(x^{-\lambda}-1\right)}{\theta}\right)}-1,$$

where $0 < x < 1$. As is known (see, for more details, [27–29]), the CHRF represents total accumulated risk up to a certain time (x), while the RHRF gives the conditional probability of failure provided that it has already occurred. As can be easily seen, in the case of the CLU distribution, the CHRF is an increasing function, while the RHRF has infinite values at both ends of the unit interval. At the same time, due to the previously mentioned properties of the CLU distribution, both functions have a very pronounced and fast convergence towards infinity when $x \to 1$. This means that the probability of a failure occurring increases rapidly as time approaches the end of the observation period. Finally, in survival analysis, the Odds represent the ratio of the likelihood that a certain failure will not occur (in a given time period) compared with the probability that it will occur. It is clear that the Odds is a monotonically decreasing function, tending to zero when $x \to 1$. This can also be seen in Figure 3, where plots of all the previously mentioned reliability functions are shown.

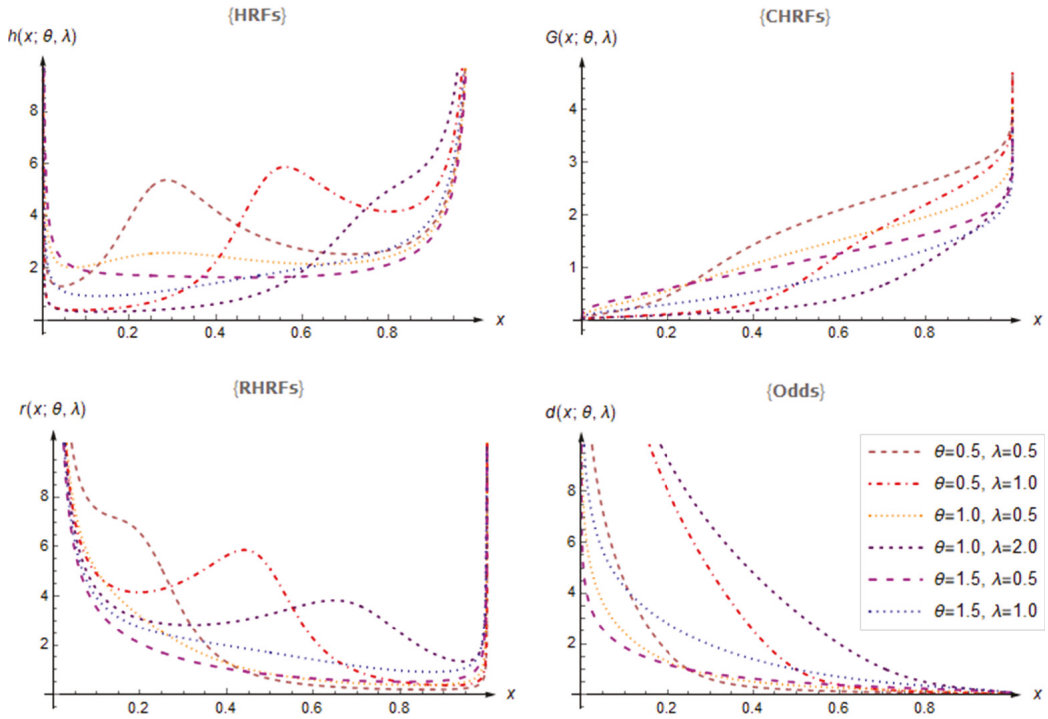


Figure 3. Plots of the HRFs $h(x;\theta,\lambda)$, CHRFs $G(x;\theta,\lambda)$, RHRFs $r(x;\theta,\lambda)$, and Odds $d(x;\theta,\lambda)$ for some values of the CLU distribution parameters $\theta, \lambda > 0$.

Further, by using Equation (3), the quantile function (QF) can be easily obtained, as follows:

$$Q(p;\theta,\lambda) := F^{-1}(p;\theta,\lambda) = \left(\mathrm{e}^{-\theta\tan\left(\pi\left(p-\frac{1}{2}\right)\right)}+1\right)^{-1/\lambda} = \left(\mathrm{e}^{\theta\cot(\pi p)}+1\right)^{-1/\lambda}, \tag{13}$$

where $p = F(x;\theta,\lambda)$. As is well known, the QF plays an important role in stochastic theory by mapping probabilities (i.e., quantiles) on them corresponding values in a given distribution. Additionally, in our case it can be a very useful tool for some additional ways of examining the shape properties of the CLU distribution, which are listed below.

Theorem 2. *Let $X : \mathcal{C}(\theta,\lambda)$ be the RV with the CLU distribution, whose QF is given by Equation* (13)*. Then, the following statements hold:*

(i) The RV X is symmetrically distributed if and only if $\lambda = 1$. Otherwise, X is positively asymmetric when $0 < \lambda < 1$ and negatively asymmetric when $\lambda > 1$.

(ii) The RV X is unimodal if and only if its parameters (θ,λ) satisfy the equality:

$$\lambda = \frac{\theta e^{\theta\cot(\pi p_0)}}{\theta+\sin(2\pi p_0)\left(e^{\theta\cot(\pi p_0)}+1\right)}, \tag{14}$$

wherein is $\partial^3 Q(p_0;\lambda,\theta)/\partial p^3 > 0$, $p_0 = F(x_0;\theta,\lambda)$ and $x_0 \in (0,1)$ is the mode of the CLU distribution. Otherwise, the PDF of X is a convex, bathtub-shaped function.

Proof. (i) Substituting the quantile $p = 1/2$ in the QF $Q(p;\theta,\lambda)$, given by Equation (13), the median $m = Q(1/2;\theta,\lambda) = 2^{-1/\lambda}$ is simply obtained. Thus, the RV X is symmetrically distributed if and only if the median is equal to $1/2$, that is, when $\lambda = 1$. The positive and negative asymmetry conditions are also easily obtained from here by solving the inequalities $Q(1/2;\theta,\lambda) < 1/2$ and $Q(1/2;\theta,\lambda) > 1/2$, respectively.

(ii) Using the rule for the derivative of the inverse function for the derivatives up to the third order of the QF $Q(p;\theta,\lambda)$, one obtains

$$\begin{aligned}
\frac{\partial Q(p;\theta,\lambda)}{\partial p} &= \frac{1}{\partial F(x;\theta,\lambda)/\partial x} = \frac{1}{f(x;\theta,\lambda)},\\
\frac{\partial^2 Q(p;\theta,\lambda)}{\partial p^2} &= -\frac{\partial f(x;\theta,\lambda)/\partial x}{f^2(x;\theta,\lambda)} \qquad\qquad .\\
\frac{\partial^3 Q(p;\theta,\lambda)}{\partial p^3} &= \frac{2(\partial f(x;\theta,\lambda)/\partial x)^2 - f(x;\theta,\lambda)\partial^2 f(x;\theta,\lambda)/\partial x^2}{f^3(x;\theta,\lambda)}.
\end{aligned}$$

Thus, $x = x_0$ is the mode of RV X, that is, the point of maxima of the PDF $f(x;\theta,\lambda)$ if and only if the value $p_0 = F(x_0;\theta,\lambda)$ satisfies

$$\frac{\partial^2 Q(p_0;\theta,\lambda)}{\partial p^2} = 0, \qquad \frac{\partial^3 Q(p_0;\theta,\lambda)}{\partial p^3} = -\frac{\partial^2 f(x_0;\theta,\lambda)/\partial x^2}{f^2(x_0;\theta,\lambda)} > 0. \tag{15}$$

After some algebraic computation, it is obtained that

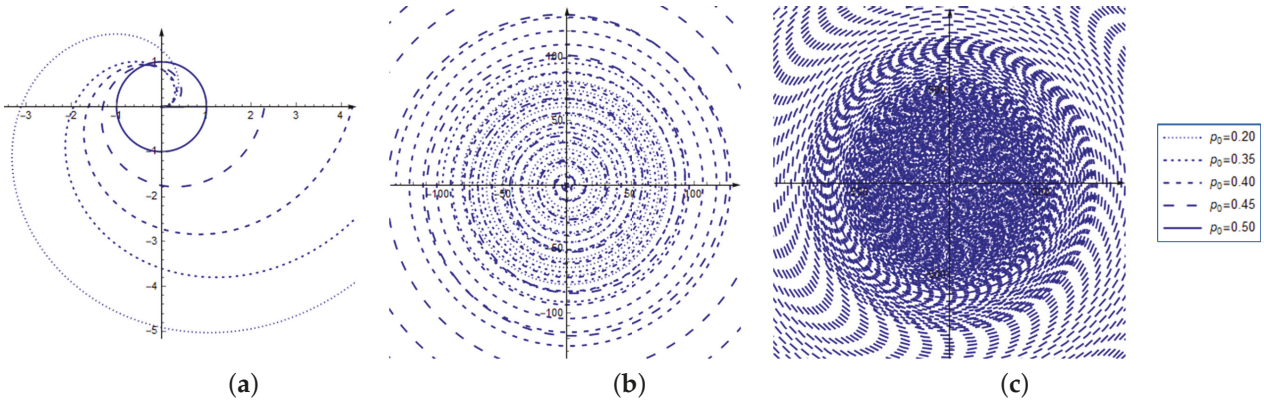
$$\frac{\partial^2 Q(p;\theta,\lambda)}{\partial p^2} = -\frac{\pi^2\theta\mathrm{e}^{\theta\cot(\pi p)}\left(\lambda(\theta+\sin(2\pi p))+\mathrm{e}^{\theta\cot(\pi p)}(\lambda\sin(2\pi p)-\theta)\right)}{\lambda^2\sin^4(\pi p)\left(\mathrm{e}^{\theta\cot(\pi p)}+1\right)^{\frac{1}{\lambda}+2}} \tag{16}$$

so that Equations (15) and (16) obviously give the statement of the theorem. □

Remark 4. *Note that the functional dependence in Equation* (14) *is periodic, with respect to* $p_0 \in (0,1)$*, as can be seen in Figure 4. Here, a polar plot of dependence of parameters* (θ, λ) *is presented that gives a unimodal CLU distribution, for different* p_0 *and various angular "views". As can be observed, these dependencies have a spiral shape, whereas in the case of* $\lambda = 1$ *(Figure 4a), a circle, as a trivial spiral, is obtained. As shown in the previous theorem, each* $\theta > 0$ *then yields a symmetric CLU distribution. Thereby, the value* $\lambda = 1$ *implies* $p_0 = 1/2$*, and after some calculation, one obtains*

$$\frac{\partial^3 Q(1/2;\theta,1)}{\partial p^3} = -\frac{\pi^3}{8}\theta\Big(\theta^2 - 4\Big).$$

Thus, when $0 < \theta < 2$*, the CLU distribution is unimodal, while for* $\theta \geq 2$ *it has a convex bathtub shape, which can also be seen in Figure 1a.*



(**a**) (**b**) (**c**)

Figure 4. Polar plots of the parameters dependence which gives the unimodal CLU distribution, with fixed $p_0 \in (0,1)$ and three various angular intervals: (**a**) $\theta \in (0, 2\pi)$; (**b**) $\theta \in (0, 25\pi)$; (**c**) $\theta \in (0, 150\pi)$.

Remark 5. *Using the QF* $Q(p;\theta,\lambda)$*, defined by Equation* (13)*, some more measures of the shape of the CLU distribution can be studied. These are, for instance, Galton skewness (GS) which measures the degree of long tail, and Moors kurtosis (MK) which measures the degree of weight of the tail of the distribution (see, e.g., Nasiru et al. [5,8]). In the case of the CLU distribution, these two measures, after some calculation, can be expressed as follows:*

$$\begin{aligned} GS(\theta,\lambda) &= \frac{Q(3/4;\theta,\lambda) - 2Q(1/2;\theta,\lambda) + Q(1/4;\theta,\lambda)}{Q(3/4;\theta,\lambda) - Q\Big(1/4;\theta,\lambda\Big)} \\ &= \frac{\left(\mathrm{e}^{-\theta}+1\right)^{-1/\lambda} + \left(\mathrm{e}^{\theta}+1\right)^{-1/\lambda} - 2^{(\lambda-1)/\lambda}}{\left(\mathrm{e}^{-\theta}+1\right)^{-1/\lambda} - \left(\mathrm{e}^{\theta}+1\right)^{-1/\lambda}}, \\ MK(\theta,\lambda) &= \frac{Q(7/8;\theta,\lambda) - Q(5/8;\theta,\lambda) + Q(3/8;\theta,\lambda) - Q(1/8;\lambda,\theta)}{Q(3/4;\theta,\lambda) - Q(1/4;\theta,\lambda)} \\ &= \frac{\left(\mathrm{e}^{\theta\tan\frac{\pi}{8}}+1\right)^{-\frac{1}{\lambda}} - \left(\mathrm{e}^{-\theta\tan\frac{\pi}{8}}+1\right)^{-\frac{1}{\lambda}} + \left(\mathrm{e}^{-\theta\cot\frac{\pi}{8}}+1\right)^{-\frac{1}{\lambda}} - \left(\mathrm{e}^{\theta\cot\frac{\pi}{8}}+1\right)^{-\frac{1}{\lambda}}}{\left(\mathrm{e}^{-\theta}+1\right)^{-\frac{1}{\lambda}} - \left(\mathrm{e}^{\theta}+1\right)^{-\frac{1}{\lambda}}}. \end{aligned}$$

Figure 5 shows 3D plots of the dependences of both these coefficients with respect to the parameters (θ,λ)*. Let us notice, for instance, in the same way as in Theorem 2, that the coefficient* $GS(\theta,\lambda)$ *has positive values (i.e., positive asymmetry is valid) for the values of the parameter* $\lambda \in (0,1)$ *and vice versa.*

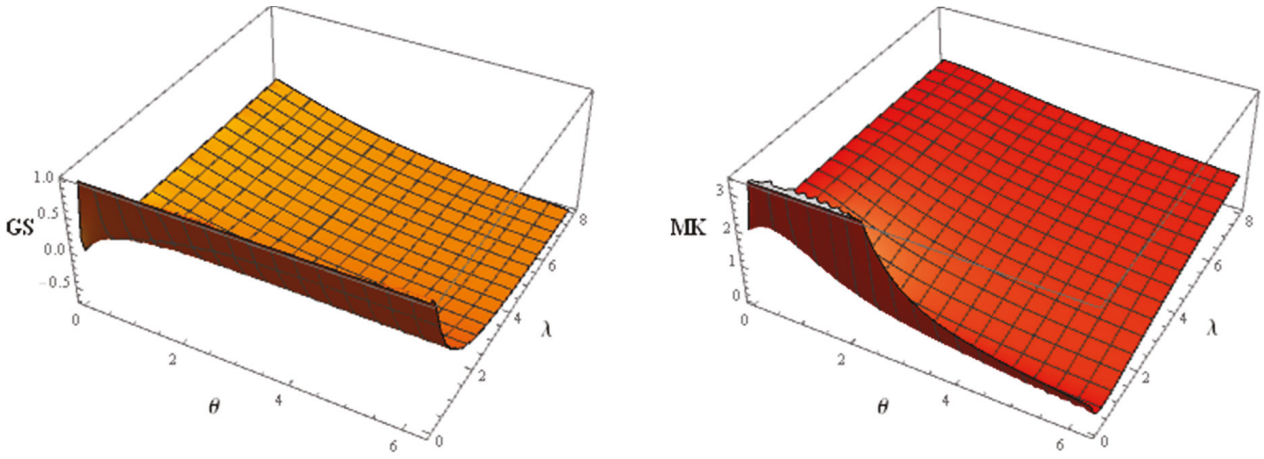


Figure 5. Thre-dimensional graphs of the dependence of the Galton skewness and Moors kurtosis with respect to the parameters (θ, λ).

3. Parameters Estimation and Simulation Study

In this section, the procedure for estimating the parameters (θ, λ) of the CLU-distributed RV X is presented. It is worth pointing out that, based on the previously mentioned properties of the CLU distribution, the usage of some common estimation procedures has some difficulties. For instance, due to the complexity of calculating the moments, that is, the likelihood function, the method of moments, or the maximum likelihood estimation methods are somewhat unsuitable. Therefore, similar to [20,21], the estimation methods based on the quantiles of the CLU distribution, i.e., the values of the QF $Q(p; \theta, \lambda)$, are discussed here.

3.1. Estimation of CLU Distribution Parameters

First, for a random sample $X_1, X_2, \ldots, X_n$ of the length n, let us define the corresponding order statistics $X_{(1)} \leq X_{(2)} \leq \ldots X_{(n)}$, whose PDFs are given with the following well-known expression:

$$f_{X_{(i)}}(x; \theta, \lambda) = \frac{n! f(x; \theta, \lambda)}{(i-1)!(n-i)!} (F(x; \theta, \lambda))^{i-1} (1 - F(x; \theta, \lambda))^{n-i}, \tag{17}$$

where $i = 1, \ldots, n$. Then, similar to the proof of Theorem 2, by substituting $p = j/4$, $j = 1, 2, 3$ into the QF $Q_p := Q(p; \theta, \lambda)$, the quartiles of the RV X are obtained:

$$Q(1/4; \theta, \lambda) = \left(e^{\theta} + 1\right)^{-1/\lambda}, \qquad Q(3/4; \theta, \lambda) = \left(e^{-\theta} + 1\right)^{-1/\lambda}, \tag{18}$$

while the second quartile is the median $m = Q(1/2; \theta, \lambda) = 2^{-1/\lambda}$. On the other hand, the sample quantiles are given as follows:

$$\widehat{Q}_p := X_{(i_p)}, \qquad i_p = \begin{cases} np, & np \text{ is integer,} \\ 1 + [np], & \text{otherwise,} \end{cases}$$

where $[np]$ is the integer part of np. Obviously, the sample quantiles are the order statistics, so their distribution is simply obtained according to Equation (17). Furthermore, sample quartiles $\widehat{Q}_{j/4}$, $j = 1, 2, 3$ can be used as the estimators of theoretical ones, and based on

them, the estimator of the shape parameter λ is given by $\widehat{\lambda} = -\ln 2/\ln \widehat{m}$, where $\widehat{m} = \widehat{Q}_{1/2}$ is the sample median. Also, according to Equation (13), it follows that

$$\left(\frac{Q(3/4,\lambda,\theta)}{Q(1/4,\lambda,\theta)}\right)^{\lambda} = \mathrm{e}^{\theta},$$

and using estimator $\widehat{\lambda}$, for the estimator of the scale parameter θ, one obtains

$$\widehat{\theta} = \ln\Big(\widehat{Q}_{3/4}/\widehat{Q}_{1/4}\Big)^{\widehat{\lambda}} = \widehat{\lambda}\Big(\ln \widehat{Q}_{3/4} - \ln \widehat{Q}_{1/2}\Big). \tag{19}$$

Note that this estimator is close to the so-called Hodges–Lehmann estimator for the scale parameter of the Cauchy distribution [30]. In addition, both obtained estimators have certain asymptotic and efficiency properties, which can be expressed as follows:

Theorem 3. *Statistics $(\widehat{\theta},\widehat{\lambda})$ are consistent and asymptotic normal (AN) estimators for the true parameters (θ,λ).*

Proof. Using some general asymptotic sample quantile theory [31–33], we first prove the consistency of the proposed estimators. To this end, for any $0 < p < 1$, note that the CDF $p = F(x;\theta,\lambda)$ is a differentiable, increasing function. Thus, quantile $Q_p := Q(p;\theta,\lambda)$ is uniquely determined by Equation (13), so according to Bahadur's representation of sample quantiles (see, e.g., Theorem 1 in [32] or [33], pp. 91–92), it is valid that

$$\widehat{Q}_p = Q_p + \frac{p - F_n(Q_p)}{f(Q_p;\theta,\lambda)} + O\Big(n^{-3/4}(\ln n)^{3/4}\Big), \tag{20}$$

where $F_n(x) := n^{-1}\sum_{i=1}^{n} I(X_i < n)$ is the empirical CDF. As is well known, for each $x \in \mathbb{R}$, the almost sure and uniform convergence $F_n(x) \xrightarrow{as} F(x;\theta,\lambda)$, $n \to \infty$ holds. By applying this convergence to Equation (20), when $x = Q_p$, it is obtained that

$$\widehat{Q}_p \xrightarrow{as} Q_p, \qquad n \to \infty,$$

i.e., the sample quantiles are consistent estimators of the theoretical ones. Finally, as estimators $(\widehat{\theta},\widehat{\lambda})$ are continuous functions of sample quartiles $\widehat{Q}_{j/4}$, $j = 1,2,3$, by using the property of continuity of almost sure convergence (see, e.g., [33], p. 24), it follows that

$$(\widehat{\theta},\widehat{\lambda}) \xrightarrow{as} (\theta,\lambda), \qquad n \to \infty.$$

For the proof of the AN property, notice that according to Equation (20), under the same assumptions as above, the following convergence in distribution is valid:

$$\sqrt{n}\Big(\widehat{Q}_p - Q_p\Big) \xrightarrow{d} \mathcal{N}\left(0, \frac{p(1-p)}{[f(Q_p;\theta,\lambda)]^2}\right), \quad n \longrightarrow \infty. \tag{21}$$

Thus, for the sample median $\widehat{m} = \widehat{Q}_{1/2}$, Equation (21) becomes

$$\sqrt{n}(\widehat{m} - m) \xrightarrow{d} \mathcal{N}(0, V_0), \quad n \longrightarrow \infty,$$

where

$$V_0 = \frac{1}{[2f(m;\theta,\lambda)]^2} = \frac{\pi^2\theta^2 4^{-\frac{1}{\lambda}-2}}{\lambda^2}.$$

According to this and the continuity of convergence in distribution (see, e.g., [33], p. 118), for the estimator $\widehat{\lambda} = -\ln 2/\ln \widehat{m}$, the following holds:

$$\sqrt{n}(\widehat{\lambda} - \lambda) \xrightarrow{d} \mathcal{N}\left(0, \sigma_{\lambda}^2\right), \qquad n \to \infty, \tag{22}$$

wherein

$$\sigma_{\lambda}^2 := \mathrm{Var}(\widehat{\lambda}) = \left[-\frac{\partial}{\partial m}\left(\frac{\ln 2}{\ln m}\right)\right]^2 \cdot V_0 = \left(\frac{\pi\theta\lambda}{4\ln 2}\right)^2.$$

In a similar way, for the estimator $\widehat{\theta}$, given by Equation (19), one obtains

$$\sqrt{n}(\widehat{\theta} - \theta) \xrightarrow{d} \mathcal{N}\left(0, \sigma_{\theta}^2\right), \qquad n \to \infty, \tag{23}$$

where, according to Equations (18)–(21), and after some computations, we obtain

$$\begin{aligned}
\sigma_{\theta}^2 &:= \mathrm{Var}(\widehat{\theta}) = \lambda^2\left(\frac{\mathrm{Var}(\widehat{Q}_{1/4})}{Q_{1/4}^2} + \frac{\mathrm{Var}(\widehat{Q}_{3/4})}{Q_{3/4}^2}\right) \\
&= \frac{3\lambda^2}{16}\left(\frac{1}{Q_{1/4}^2 f^2(Q_{1/4},\theta,\lambda)} + \frac{1}{Q_{3/4}^2 f^2(Q_{3/4},\theta,\lambda)}\right) \\
&= \frac{3\lambda^2}{16}\left(\left(\mathrm{e}^{\theta}+1\right)^{2/\lambda}\frac{4\pi^2\mathrm{e}^{2\theta}\theta^2}{\lambda^2\left(\mathrm{e}^{\theta}+1\right)^{2+2/\lambda}} + \left(\mathrm{e}^{-\theta}+1\right)^{2/\lambda}\frac{4\pi^2\theta^2}{\left(\mathrm{e}^{\theta}+1\right)^2\lambda^2\left(\mathrm{e}^{-\theta}+1\right)^{2/\lambda}}\right) \\
&= \frac{3\pi^2\theta^2\left(\mathrm{e}^{2\theta}+1\right)}{4\left(\mathrm{e}^{\theta}+1\right)^2}.
\end{aligned} \tag{24}$$

Therefore, the convergences in Equations (22) and (23) confirm the AN properties of the proposed estimators. □

Remark 6. *Note that the estimator $\widehat{\theta}$, calculated using Equation* (19) *and the previously obtained value of $\widehat{\lambda}$, has a variance, given by Equation* (24)*, which does not depend on λ. This is one reason that justifies the use of this estimator.*

3.2. Simulation Study

In the following, using Monte Carlo simulations of samples from the CLU distribution, the proposed estimators for each of the samples are calculated, and their statistical analysis is performed. To this end, three different types of samples are considered:

(i) Sample I is drawn from a positively asymmetric bathtub-shaped CLU distribution $\mathcal{C}(\theta,\lambda)$, with parameters $\theta = 0.3$, $\lambda = 0.2$.

(ii) Sample II is drawn from a symmetric unimodal CLU distribution $\mathcal{C}(\theta,\lambda)$, with parameters $\theta = 0.4$, $\lambda = 1$.

(iii) Sample III is drawn from a negatively asymmetric unimodal CLU distribution $\mathcal{C}(\theta,\lambda)$, with parameters $\theta = 0.5$, $\lambda = 2$.

Note that the simulated values of all samples are generated by the R-package "distr" [34], according to which the estimates $\widehat{\lambda}$ and $\widehat{\theta}$ are simply calculated. For each of them, four different sample sizes are considered, equal to $n = 50$, $n = 150$, $n = 500$, and $n = 1500$, respectively. Some realizations of these samples are shown in Figure 6 below, and it is worth pointing out that the choice of sample lengths, among others, was taken as "close" to some real-world data series that will be discussed later. The results of Monte

Carlo simulations and statistical analyses of estimated parameters obtained based on the realization of the above samples are presented in Tables 1–3.

Table 1. Summary statistics, estimation errors, and AN testing of parameter estimates of the CLU distribution. (Sample I with the true parameters: $\theta = 0.3, \lambda = 0.2$).

Statistics	$n = 50$		$n = 150$		$n = 500$		$n = 1500$	
	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$
Min.	0.1603	0.1512	0.1622	0.2108	0.1974	0.2488	0.2025	0.2622
Mean	0.2056	0.2541	0.2050	0.2520	0.2049	0.2913	0.2017	0.2954
Max.	0.2325	0.4136	0.2196	0.3581	0.2131	0.3366	0.2126	0.3221
SD	9.19×10^{-3}	0.0518	5.34×10^{-3}	0.0308	2.98×10^{-3}	0.0161	1.85×10^{-3}	9.91×10^{-3}
MSEE	0.0107	0.0459	7.30×10^{-3}	0.0480	5.75×10^{-3}	0.0446	5.36×10^{-3}	0.0359
FEE (%)	5.3595	16.920	3.6811	16.755	3.6493	16.006	2.8732	14.883
AD	0.4139	0.7572 *	0.3703	0.6934	0.2616	0.6417	0.2060	0.3913
(*p*-value)	(0.3335)	(0.0481)	(0.4216)	(0.0692)	(0.7027)	(0.0929)	(0.8688)	(0.3769)
D	0.0452	0.0575	0.0484	0.0560	0.0477	0.0549	0.0473	0.0459
(*p*-value)	(0.4095)	(0.1098)	(0.3032)	(0.1312)	(0.3245)	(0.1498)	(0.3369)	(0.3821)

* $0.01 < p < 0.05$.

Table 2. Summary statistics, estimation errors, and AN testing of parameter estimates of the CLU distribution. (Sample II with the true parameters: $\theta = 0.4, \lambda = 1$).

Statistics	$n = 50$		$n = 150$		$n = 500$		$n = 1500$	
	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$
Min.	0.8107	0.1940	0.9086	0.2842	0.9471	0.3139	0.9648	0.3308
Mean	1.0039	0.3748	0.9993	0.3768	1.0013	0.3840	0.9995	0.3984
Max.	1.1577	0.6548	1.1105	0.6078	1.0474	0.4427	1.0310	0.4186
SD	0.0604	0.0884	0.0388	0.0470	0.0202	0.0251	0.0122	0.0164
MSEE	0.0561	0.0252	0.0388	0.0232	0.0202	0.0216	0.0122	0.0160
FEE (%)	5.6094	6.2999	3.8661	5.7877	2.0188	5.4108	1.2209	4.0024
AD	0.5214	1.0854 **	0.4392	0.6846	0.3166	0.6818	0.1565	0.2659
(*p*-value)	(0.1827)	(7.41×10^{-3})	(0.2902)	(0.0727)	(0.5375)	(0.0739)	(0.9536)	(0.6881)
D	0.0566	0.0700 *	0.0435	0.0680 *	0.0407	0.0494	0.0344	0.0417
(*p*-value)	(0.1220)	(0.0184)	(0.4719)	(0.0252)	(0.5779)	(0.2732)	(0.8152)	(0.5410)

* $0.01 < p < 0.05$; ** $p < 0.01$.

Table 3. Summary statistics, estimation errors, and AN testing of parameter estimates of the CLU distribution. (Sample III with the true parameters: $\theta = 0.5, \lambda = 2$).

Statistics	$n = 50$		$n = 150$		$n = 500$		$n = 1500$	
	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$	$\widehat{\lambda}$	$\widehat{\theta}$
Min.	1.613	0.2653	1.769	0.3186	1.823	0.3830	1.891	0.4236
Mean	1.984	0.4741	1.985	0.4735	1.988	0.4763	2.004	0.4908
Max.	2.425	0.8680	2.264	0.7305	2.105	0.5807	2.068	0.5277
SD	0.1526	0.1155	0.0878	0.0696	0.0509	0.0339	0.0304	0.0203
MSEE	0.1526	0.0259	0.0884	0.0265	0.0529	0.0237	0.0342	0.0233
FEE (%)	7.6282	5.1816	4.4202	5.3052	2.6455	4.7322	1.7102	4.6517
AD	0.5397	1.4809 **	0.4006	0.5076	0.1886	0.6136	0.1471	0.5482
(*p*-value)	(0.1644)	(7.83×10^{-4})	(0.3583)	(0.1977)	(0.9005)	(0.1091)	(0.9659)	(0.1565)
D	0.0525	0.0693 *	0.0481	0.0494	0.0370	0.0505	0.0307	0.0490
(*p*-value)	(0.1961)	0.0208	(0.3113)	(0.2737)	(0.7233)	(0.2440)	(0.9168)	(0.2865)

* $0.01 < p < 0.05$; ** $p < 0.01$.

In order to check the effectiveness of proposed estimators, $S = 200$ independent simulations of each sample were conducted, and the results of the statistical analysis of

the obtained estimates are presented in Tables 1–3. In other words, Tables 1–3 contain their summary statistics, as well as the mean square errors of estimation (MSEE), fractional errors of estimation (FEE), and the results of the Anderson–Darling (AD) and Lilliefors (D) normality tests. According to this, it can be observed that the proposed estimators are efficient, as the SD, MSEE, and FEE values decrease with increasing sample size. Therefore, the estimates $\widehat{\lambda}$ are more stable and accurate than $\widehat{\theta}$, since the estimates $\widehat{\theta}$ are obtained by a two-stage procedure, that is, using Equation (19) and previously calculated values of $\widehat{\lambda}$.

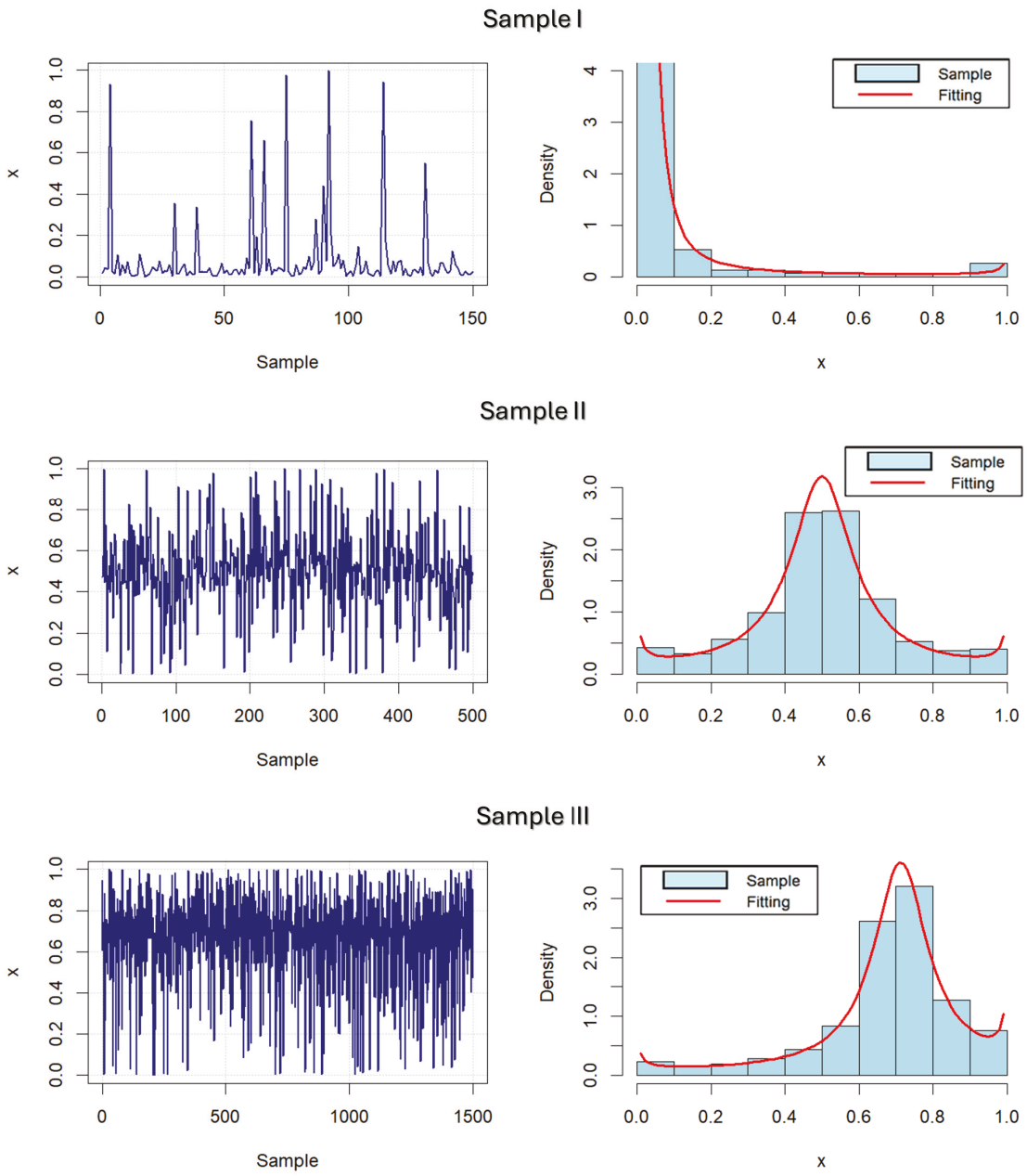


Figure 6. Left panels: realizations of samples with CLU distribution. Right panels: empirical and fitted PDFs of the RV $X : \mathcal{C}(\theta, \lambda)$.

Some confirmation of this can be seen in Figure 6, where sample realizations, as well as their empirical and fitted distributions, are shown. The AN properties of the obtained estimates also confirm similar conclusions. The test statistics and the appropriate p-values, computed using the R-package "nortest" [35], show that for smaller samples, the estimates

of $\widehat{\theta}$ have a less pronounced AN property than $\widehat{\lambda}$. Nevertheless, the AN property was confirmed in most samples, as can be also seen in Figure 7. Here, in order to compare the empirical distribution of the obtained samples with the Gaussian distribution, Q-Q plots of estimates of both parameters for samples of size $n = 500$ are shown.

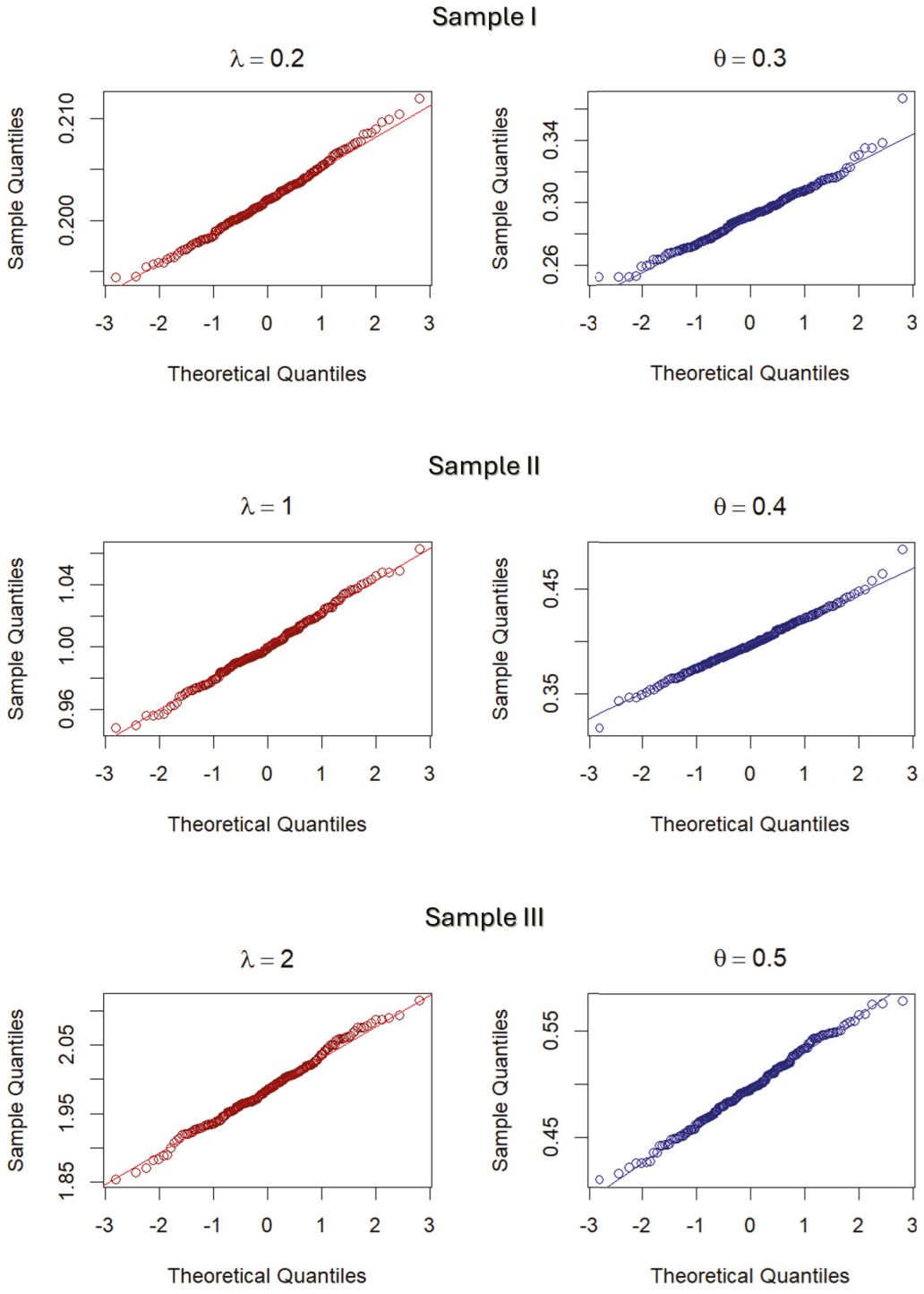


Figure 7. Q-Q plots of parameter estimates $\widehat{\lambda}$ (**left**) and $\widehat{\theta}$ (**right**) for samples of length $n = 500$.

4. Applications of the CLU Distribution

Some possibilities for the application of LCU distribution in real-world data modeling are discussed here, as well as a comparison of the proposed model with some existing

ones. To this end, the following three real-world data sets were observed whose empirical distributions have a presence of "heavy tails":

(i) The first set of data, named Series A, was obtained by researching the percentage of use of certain types of antibiotics (aminoglycosides, penicillins, and cephalosporins), before and during the SarS CoV virus pandemic. In this way, by using the official data of the General Hospital in the city of Vranje, the regional center of southern Serbia, a data set of length $n = 324$ was obtained.

(ii) The second one, named Series B, represents the percentage of time spent using the service by end users and is part of the training data set relating to network and telecommunication traffic in India. This data set, also observed in [21], was collected by [36] and consists of $n = 542$ monthly end-user costs (in Indian rupees), normalized to their maximum and minimum values, yielding a data set within a unit interval.

(iii) The third, Series C, obtained according to the official data of the National Association of Securities Dealers in Automated Quotations (NASDAQ) [37], refers to the daily log-returns of stock prices in the period from 29 May 2018 to 24 May 2023 and represents a time series of length $n = 1257$. Notice that its dynamics were examined earlier in [38], where it is shown that it can be viewed as independent realizations of the RV with Cauchy distribution. Therefore, this series is transformed here by the previously mentioned logistic function, that is, it represents the input of the logistic regression model, while the output is the realization of the RV with the CLU distribution.

It is also worth noting that data with larger lengths are observed, as the presence of "heavy tails" in their empirical distributions is more noticeable (see Figure 8 below). To check the efficiency of the CLU distribution, it is compared with the following three previously known unit distributions:

(i) Beta distribution, whose PDFs is

$$g_1(x; a, b) = \frac{x^{a-1}(1-x)^{b-1}}{B(a,b)},$$

where $0 < x < 1$; $a, b > 0$ are the distribution parameters, and $B(a, b)$ is the beta function.

(ii) Bounded truncated Cauchy power exponential (BTCPE) distribution, introduced by [25], whose PDF is

$$g_2(x; a, b) = \frac{4abx^{a-1}(1-x^a)^{b-1}}{\pi\big(1+(1-x^a)^{2b}\big)},$$

where $0 < x < 1$ and $a, b > 0$.

(iii) The unit-Weibull (UW) distribution, proposed by [39], whose PDF is:

$$g_3(x; a, b) = \frac{ab(-\ln x)^{b-1}\exp(-a(-\ln x)^b)}{x},$$

where $0 < x < 1$ and $a, b > 0$.

To obtain the estimated parameter values of the Beta distribution, the method of moments (MM) is applied (see, e.g., [40]). According to this, the resulting estimates are

$$\widehat{a} = \bar{x}\Big(\frac{\bar{x}(1-\bar{x})}{\bar{v}} - 1\Big), \qquad \widehat{b} = (1-\bar{x})\Big(\frac{\bar{x}(1-\bar{x})}{\bar{v}} - 1\Big),$$

where $\bar{x} = n^{-1}\sum_{i=1}^{n} x_i$ is the sample mean, $\bar{v} = (n-1)^{-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$ is the sample variance, and $\bar{v} < \bar{x}(1-\bar{x})$. From the other side, for the other two distributions, the maximum

likelihood (ML) estimation method is used, where the parameters $a, b > 0$ are estimated by maximizing the following log-likelihood function:

$$\ell_j(a, b|x_1, \ldots, x_n) = \sum_{i=1}^{n} \ln g_j(x_i; a, b), \quad j = 2, 3. \tag{25}$$

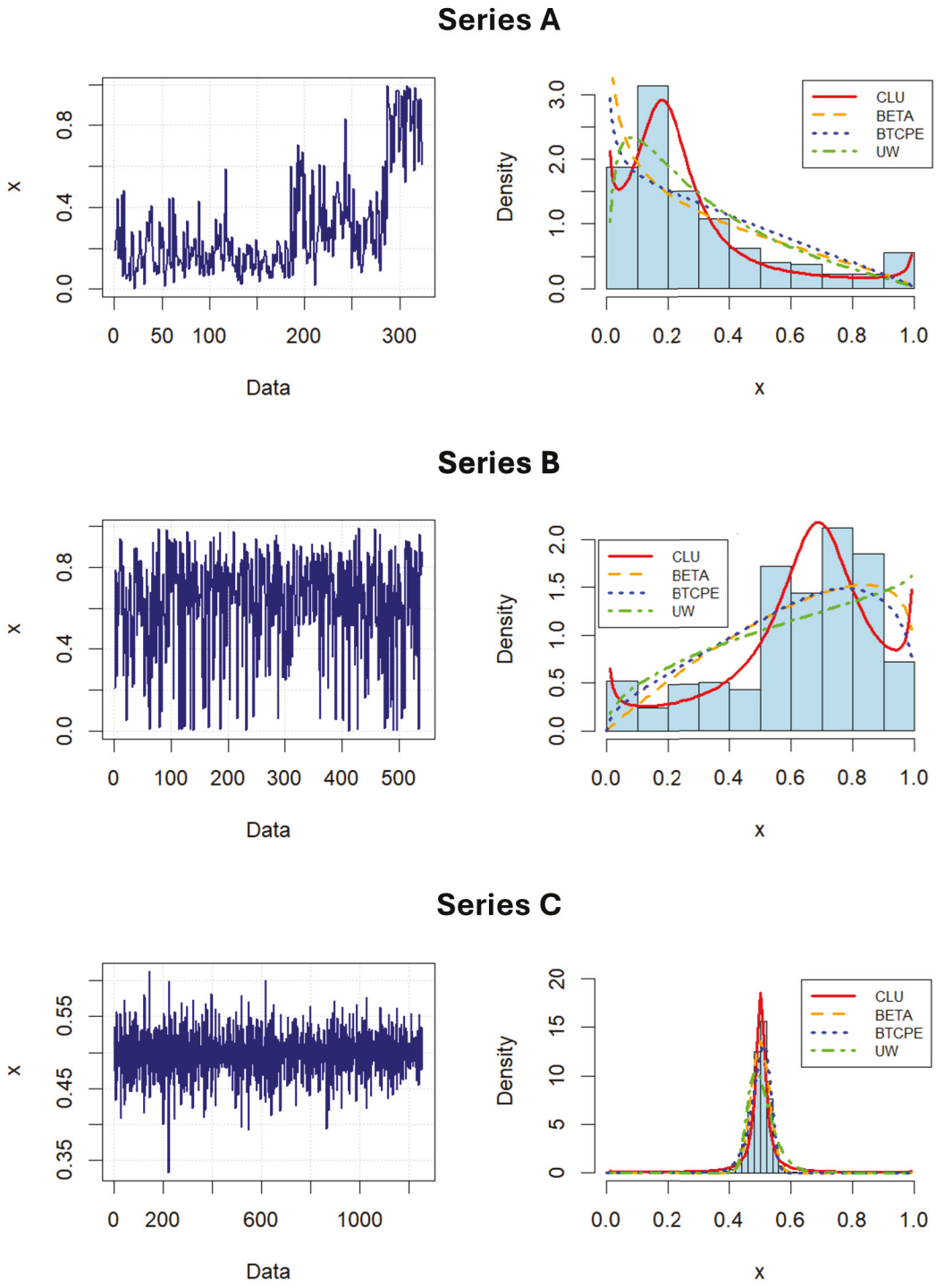


Figure 8. Left panels: observed values of the real-world data series. **Right** panels: empirical and fitted PDFs, obtained with CLU, Beta, and Kumaraswamy distributions.

For numerical optimization of the log-likelihood functions given by Equation (25), the R-function "optim()" within the "stats" package was used. As is well known, the proposed estimators for both of the above distributions have desirable properties, such as consistency and AN features. Therefore, one of the reasons for choosing such estimation methods is to compare the CLU distribution with other, well-known distributions, also in terms of different estimation procedures.

The observations of the aforementioned series, along with fitted PDFs obtained by the previously described parameter estimation procedures, are also shown in Figure 8. As can be seen, all series have pronounced and persistent fluctuations, which create "heavy tails" in their distributions. In particular, it is noticeable that the empirical distribution of Series A is significantly positively skewed, and Series B has a negatively asymmetric distribution, while Series C has an approximately symmetric empirical distribution. This can be confirmed by the estimated parameter values for each series, as well as for all competing distributions, which are shown in Tables 4–6. Thus, for instance, Series C has an estimated value of the shape parameter $\widehat{\lambda} \approx 1$, which indicates the possibility of symmetry of its distribution.

In the following, $S = 1000$ independent Monte Carlo simulations were conducted for all competing models. Agreement between the distributions of the real and fitted data was checked using the MSEE error statistics, the Akaike information criterion (AIC), the Bayesian information criterion (BIC), and the Hannan–Quinn Information Criterion (HQIC). In addition, two-sample Kolmogorov–Smirnov (KS), Anderson–Darling (AD), and Cramér–von Mises (CVM) tests of asymptotic distributions equivalence were conducted, and their test statistics, along with the corresponding p-values, are also shown in Tables 4–6. According to the presented results, it can be seen that the MSEE, AIC, and BIC values are significantly lower in the case of CLU distribution. More specifically, in the case of Series A with only two distributions, CLU and UW, the hypothesis of equivalence of the empirical and fitted distributions is not rejected at either significance level, $p > 0.01$ or $p > 0.05$.

Table 4. Estimated parameters of the competing distributions for Series A, along with corresponding estimation errors, fit, and test statistics.

Parameters/Statistics	Distributions			
	CLU	**BETA**	**BTCPE**	**UW**
λ/a	0.4303	0.7160	0.7345	0.4057
θ/b	0.4821	1.7147	1.7975	1.6707
MSEE	0.0111	0.0311	0.0387	0.0139
AIC	−251.6	−190.6	−115.1	−184.8
BIC	−236.0	−175.1	−99.54	−169.2
HQIC	−240.6	−179.6	−104.1	−173.8
KS	0.0679	0.1235 *	0.1111 *	0.0926
(p-value)	(0.4439)	(0.0143)	(0.0366)	(0.1243)
AD	1939.3	11,368.7 **	7399.3 **	2922.9
(p-value)	(0.1810)	(2.50×10^{-4})	(5.00×10^{-4})	(0.0585)
CVM	0.5027	4.4107 **	2.9059 *	1.2942
(p-value)	(0.4785)	(2.50×10^{-3})	(0.0110)	(0.1305)

* $0.01 < p < 0.05$; ** $p < 0.01$.

Table 5. Estimated parameters of the competing distributions for Series B, along with corresponding estimation errors, fit, and test statistics.

Parameters/Statistics	Distributions			
	CLU	**BETA**	**BTCPE**	**UW**
λ/a	1.7560	1.9597	1.5014	1.5054
θ/b	0.7675	1.1883	1.2770	0.9793
MSEE	8.36×10^{-3}	0.0123	0.0252	0.0254
AIC	−183.2	−145.6	−112.6	−78.21
BIC	−166.6	−129.0	−96.07	−61.62
HQIC	−171.9	−134.2	−101.3	−66.85
KS	0.0701	0.0867 *	0.1236 **	0.1550 **
(p-value)	(0.1393)	(0.0340)	(5.06×10^{-4})	(4.44×10^{-6})
AD	2988.6	6560.9 *	7800.6 *	15,499.5 **
(p-value)	(0.2100)	(0.0235)	(0.0140)	(1.50×10^{-3})
CVM	0.7280	1.3191	2.3889 *	5.068 **
(p-value)	(0.3060)	(0.0990)	(0.0215)	(1.00×10^{-3})

* $0.01 < p < 0.05$; ** $p < 0.01$.

Table 6. Estimated parameters of the competing distributions for Series C, along with corresponding estimation errors, fit, and test statistics.

Parameters/Statistics	Distributions			
	CLU	**BETA**	**BTCPE**	**UW**
λ/a	1.0047	145.3	16.11	26.00
θ/b	0.0676	145.3	57,122.4	10.03
MSEE	5.76×10^{-4}	6.42×10^{-4}	2.32×10^{-3}	4.47×10^{-3}
AIC	−5306.6	−5036.4	−4521.5	−4194.9
BIC	−5288.4	−5018.1	−4503.3	−4176.6
HQIC	−5294.8	−5024.5	−4509.7	−4183.0
KS	0.0430	0.0494	0.0518	0.1178 **
(p-value)	(0.1960)	(0.0937)	(0.0692)	(5.34×10^{-8})
AD	10,132.8	11,432.6	10,628.3	12,7017.9 **
(p-value)	(0.0980)	(0.0725)	(0.0735)	(2.50×10^{-4})
CVM	1.3898	1.6240	1.8286 *	14.72614 **
(p-value)	(0.1175)	(0.0600)	(0.0405)	(2.50×10^{-4})

* $0.01 < p < 0.05$; ** $p < 0.01$.

At the same time, it is noticeable that in the case of Series B, the CLU distribution has slightly better fitting characteristics compared with all other competing distributions. Thus, for instance, for the CLU distribution, the values of the MSEE, AIC, BIC, and HQIC statistics are relatively lower than for other distributions, while its testing results do not reject the hypothesis of equivalence of the empirical or theoretical distributions. Finally, when it comes to Series C, it can be noted that it has an approximately symmetric distribution, as well as that the CLU and Beta distributions have a similar, approximately equal fitting quality. Also, for these two distributions, the values of the corresponding test statistics, at the significant level $p > 0.01$, do not reject the null hypothesis of asymptotic equality of the empirical distributions with the corresponding fitted distributions. Nevertheless, similar to before, it is clear that the CLU distribution has better fitting characteristics compared with all other theoretical distributions. (See also Table A1 and Figure A1 in Appendix A).

5. Conclusions

This manuscript discusses a new unit stochastic distribution, obtained by the logistic transformation of the Cauchy distribution to the unit interval. In this way, the CLU

distribution was obtained, which is examined in terms of its basic stochastic properties, as well as the quantile-based procedure for estimating its parameters. The consistency and AN properties of this estimation procedure were also investigated, as well as a numerical study of its effectiveness. Finally, a practical application of the CLU distribution in fitting three real-world data sets is presented. In order to test the effectiveness of the CLU distribution in its fitting capabilities, it is compared with three well-known distributions, that is, with the Beta, BTCPE, and UW distributions. According to the results obtained in this way, it can be noted that the CLU distribution generally fits the empirical distributions of the observed data better. Moreover, the proposed distribution can be applied to fitting both asymmetric and symmetric distributions with "heavy tails".

All of this can certainly be a motive for further research, which can be carried out in several different directions. First of all, logistic mapping of some other continuous distributions (such as, for instance, Gaussian, Weibull, or exponential power distribution) can lead to new unit distributions with somewhat different stochastic properties. As already pointed out, this has been carried out in [20,21], where logistic transformations of the Laplacian and Gumbel distributions were performed. Additionally, as is well known, logistic regressions can be applied in machine learning and data science, and this may also be an additional motivation to further examine unit distributions based on logistic maps.

Author Contributions: Conceptualization, V.S.S., T.J.S. and R.B.; methodology, V.S.S., T.J.S. and R.B.; software, V.S.S., T.J.S. and B.P.; validation, V.S.S., T.J.S. and R.B.; formal analysis, V.S.S., T.J.S. and R.B.; data curation, V.S.S., B.P. and Z.L.; writing—original draft preparation, V.S.S., T.J.S. and R.B.; writing—review and editing, T.J.S., B.P. and Z.L.; visualization, V.S.S., R.B. and B.P.; supervision, B.P. and Z.L.; project administration, B.P. and Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in this study are publicly available on the "Kaggle" website: https://www.kaggle.com (accessed on 2 January 2025) and the National Association of Securities Dealers Automated Quotations (NASDAQ): https://nasdaq.com (accessed on 7 August 2024) .

Acknowledgments: The authors are very grateful to the anonymous reviewers whose valuable comments significantly improved this manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Table A1. Comparison of empirical quartiles (EQs) for observed real-world data and theoretical ones obtained using CLU, Beta, BTCPE, and UW distributions (The most appropriate ones are shown in bold).

Series	Quartile	EQs	CLU	BETA	BTCPE	UW
Series A	25%	**0.1241**	0.1217	0.0829	0.1298	**0.1259**
	50% (*Median*)	**0.1997**	**0.2054**	0.2335	0.2897	0.2537
	75%	**0.3807**	**0.3755**	0.4613	0.5010	0.4443
Series B	25%	**0.5188**	**0.5251**	0.4470	0.4180	0.3953
	50% (*Median*)	**0.6739**	**0.6696**	0.6511	0.6271	0.6292
	75%	**0.8032**	0.7889	0.8216	**0.7988**	0.8230
Series C	25%	**0.4836**	**0.4849**	0.4802	0.4789	0.4740
	50% (*Median*)	**0.5016**	**0.5016**	0.5000	0.5027	0.4982
	75%	**0.5173**	**0.5183**	0.5198	0.5220	0.5282

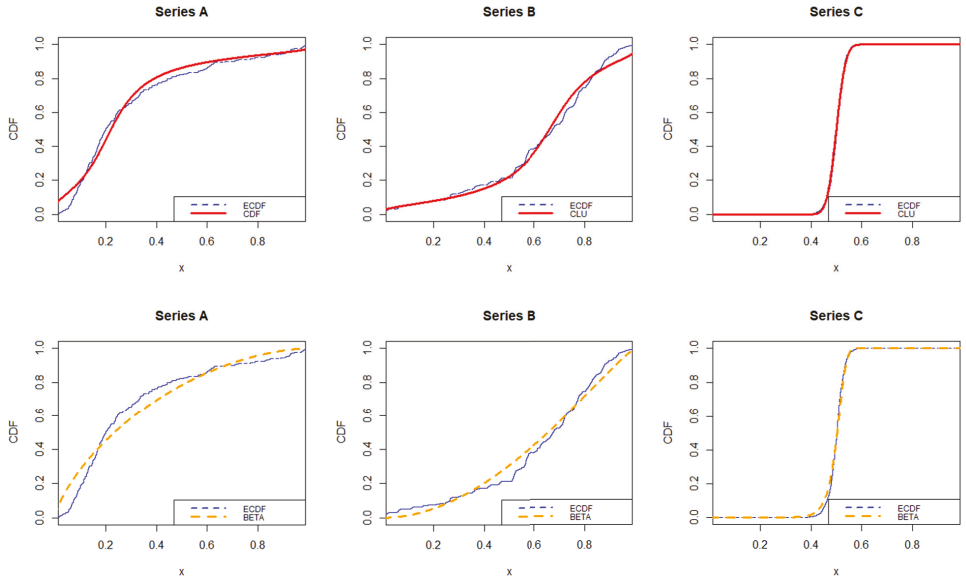


Figure A1. *Cont.*

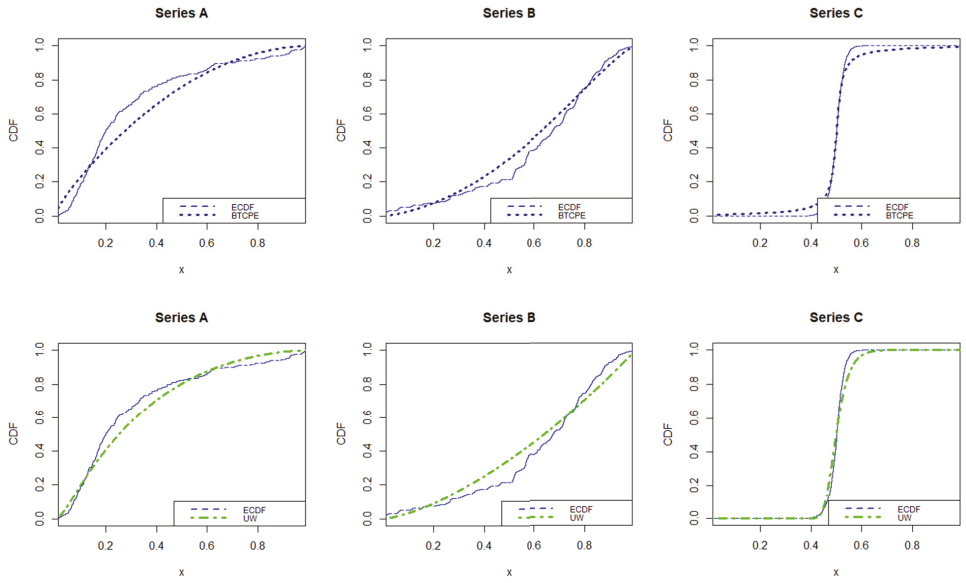


Figure A1. Comparison of ECDFs for observed real-world data and corresponding theoretical CDFs obtained using CLU, Beta, BTCPE, and UW distributions.

References

1. Afify, A.Z.; Nassar, M., Kumar, D.; Cordeiro, G.M. A New Unit Distribution: Properties and Applications. *Electron. J. Appl. Stat.* **2022**, *15*, 460–484.
2. Martínez-Flórez, G.; Azevedo-Farias, R.B.; Tovar-Falón, R. New Class of Unit-Power-Skew-Normal Distribution and Its Associated Regression Model for Bounded Responses. *Mathematics* **2022**, *10*, 3035. [CrossRef]
3. Korkmaz, M.Ç.; Korkmaz, Z.S. The Unit Log–log Distribution: A New Unit Distribution with Alternative Quantile Regression Modeling and Educational Measurements Applications. *J. Appl. Stat.* **2023**, *50*, 889–908. [CrossRef] [PubMed]
4. Condino, F.; Domma, F. Unit Distributions: A General Framework, Some Special Cases, and the Regression Unit-Dagum Models. *Mathematics* **2023**, *11*, 2888.. [CrossRef]
5. Nasiru, S.; Abubakari, A.G.; Chesneau, C. The Arctan Power Distribution: Properties, Quantile and Modal Regressions with Applications to Biomedical Data. *Math. Comput. Appl.* **2023**, *28*, 25. [CrossRef]
6. Alomair, G.; Akdoğan, Y.; Bakouch, H.S.; Erbayram, T. On the Maximum Likelihood Estimators' Uniqueness and Existence for Two Unitary Distributions: Analytically and Graphically, with Application. *Symmetry* **2024**, *16*, 610. [CrossRef]

7. Alotaibi, R.; Nassar, M.; Elshahhat, A. Estimation and Optimal Censoring Plan for a New Unit Log-Log Model via Improved Adaptive Progressively Censored Data. *Axioms* **2024**, *13*, 152. [CrossRef]
8. Nasiru, S.; Chesneau, C.; Ocloo, S.K. The Log-Cosine-Power Unit Distribution: A New Unit Distribution for Proportion Data Analysis. *Decis. Anal. J.* **2024**, *10*, 100397 . [CrossRef]
9. Alvarez, P.I.; Varela, H.; Cortés, I.E.; Venegas, O.; Gómez, H.W. Modified Unit-Half-Normal Distribution with Applications. *Mathematics* **2024**, *12*, 136. [CrossRef]
10. Shakhatreh, M.K.; Aljarrah, M.A. Bayesian Analysis of Unit Log-Logistic Distribution Using Non-Informative Priors. *Mathematics* **2023**, *11*, 4947. [CrossRef]
11. Alsadat, N.; Hassan, A.S.; Elgarhy, M.; Chesneau, C.; Mohamed, R.E. An Efficient Stress–Strength Reliability Estimate of the Unit Gompertz Distribution Using Ranked Set Sampling. *Symmetry* **2023**, *15*, 1121. [CrossRef]
12. Krishna, A.; Maya, R.; Chesneau, C.; Irshad, M.R. The Unit Teissier Distribution and Its Applications. *Math. Comput. Appl.* **2022**, *27*, 12. [CrossRef]
13. Alsadat, N.; Elgarhy, M.; Karakaya, K.; Gemeay, A.M.; Chesneau, C.; Abd El-Raouf, M.M. Inverse Unit Teissier Distribution: Theory and Practical Examples. *Axioms* **2023**, *12*, 502. [CrossRef]
14. Fayomi, A.; Hassan, A.S.; Baaqeel, H.; Almetwally, E.M. Bayesian Inference and Data Analysis of the Unit–Power Burr X Distribution. *Axioms* **2023**, *12*, 297. [CrossRef]
15. Biçer, C.; Bakouch, H.S.; Biçer, H.D.; Alomair, G.; Hussain, T.; Almohisen, A. Unit Maxwell-Boltzmann Distribution and Its Application to Concentrations PoGLUtant Data. *Axioms* **2024**, *13*, 226. [CrossRef]
16. Bakouch, H.S.; Hussain, T.; Tošić, M.; Stojanović, V.S.; Qarmalah, N. Unit Exponential Probability Distribution: Characterization and Applications in Environmental and Engineering Data Modeling. *Mathematics* **2023**, *11*, 4207. [CrossRef]
17. Alsadat, N.; Taniş, C.; Sapkota, L.P.; Kumar, A.; Marzouk, W.; Gemeay, A.M. Inverse Unit Exponential Probability Distribution: Classical and Bayesian Inference with Applications. *AIP Adv.* **2024**, *14*, 055108. [CrossRef]
18. Ramadan, A.T.; Tolba, A.H.; El-Desouky, B.S. A Unit Half-Logistic Geometric Distribution and Its Application in Insurance. *Axioms* **2022**, *11*, 676. [CrossRef]
19. Nasiru, S.; Chesneau, C.; Abubakari, A.G.; Angbing, I.D. Generalized Unit Half-Logistic Geometric Distribution: Properties and Regression with Applications to Insurance. *Analytics* **2023**, *2*, 438–462. [CrossRef]
20. Stojanović, V.S.; Jovanović Spasojević, T.; Jovanović, M. Laplace-Logistic Unit Distribution with Application in Dynamic and Regression Analysis. *Mathematics* **2024**, *12*, 2282. [CrossRef]
21. Stojanović, V.S.; Jovanović M.; Pažun, B.; Langović, Z.; Grujčić, Ž. Gumbel–Logistic Unit Distribution with Application in Telecommunications Data Modeling. *Symmetry* **2024**, *16*, 1513. [CrossRef]
22. McDonald, J.B.; Xu, Y.J. A generalization of the beta distribution with applications. *J. Econom.* **1995**, *66*, 133–152. . [CrossRef]
23. Smithson, M.; Shou, Y. CDF-quantile distributions for modelling random variables on the unit interval. *Brit. J. Math. Stat. Psy.* **2017** *70*, 412–438. . [CrossRef]
24. Smithson M.; Shou, Y. Flexible CDF-quantile distributions on the closed unit interval, with software and applications. *Commun. Stat. A-Theor.* **2024**, *53*, 3876–3898. . [CrossRef]
25. Nasiru, S.; Abubakari, A.G.; Chesneau, C. New Lifetime Distribution for Modeling Data on the Unit Interval: Properties, Applications and Quantile Regression. *Math. Comput. Appl.* **2022**, *27*, 105. . [CrossRef]
26. Rudin, W. *Real and Complex Analysis*, 3rd ed.; McGraw-Hill: New York, NY, USA, 1987. Available online: https://59clc.wordpress.com/wp-content/uploads/2011/01/real-and-complex-analysis.pdf (accessed on 5 December 2024).
27. Lawless, J.F. *Statistical Models and Methods for Lifetime Data*; 2nd ed.; John Wiley and Sons: Hoboken, NJ, USA, 2003. [CrossRef]
28. Finkelstein, M.S. On the Reversed Hazard Rate, *Reliab. Eng. Syst. Safe.* **2002**, *78*, 71–75. [CrossRef]
29. Unnikrishnan Nair, N.; Sankaran, P. G.: Odds Function and Odds Rate for Discrete Lifetime Distributions, *Commun. Stat. A-Theor.* **2015**, *44*, 4185–4202. [CrossRef]
30. Kravchuk, O.Y.; Pollett, P.K. Hodges-Lehmann scale estimator for Cauchy distribution. *Commun. Stat. A-Theor.* **2012**, *41* , 3621–3632. [CrossRef]
31. Zieliński, R. Uniform Strong Consistency of Sample Quantiles. *Stat. Prob. Lett.* **1998**, *37*, 115–119. [CrossRef]
32. Dudek, D.; Kuczmaszewska, A. Some Practical and Theoretical Issues Related to the Quantile Estimators. *Stat. Pap.* **2024**, *65*, 3917–3933. [CrossRef]
33. Serfling, R.J. *Approximation Theorems of Mathematical Statistics*, 2nd ed.; John Wiley and Sons: Hoboken, NJ, USA, 2002.
34. Ruckdeschel, P.; Kohl, M.; Stabla, T.; Camphausen, F. S4 Classes for Distributions. *R News* **2006**, *6*. Available online: https://CRAN.R-project.org/doc/Rnews (accessed on 1 August 2024).
35. Gross, L. Tests for Normality. *R Package Version* 1.0-2. 2013. Available online: http://CRAN.R-project.org/package=nortest (accessed on 1 August 2024).

36. Mirza, M.H. "Telecom_CHURN_DATASET" (Dataset). 2023. Available online: https://www.kaggle.com/datasets/mirzahasnine/telecom-churn-dataset (accessed on 2 January 2025).
37. Nasdaq. Available online: https://nasdaq.com (accessed on 7 August 2024).
38. Ljajko, E.; Stojanović, V.; Tošić, M.; Božović, I. Cauchy Split-BREAK Process: Asymptotic Properties and Application in Securities Market Analysis. *U.P.B. Sci. Bull. Series A* **2023**, *85*, 139–154.
39. J. Mazucheli, J.; Menezes, A.F.B.; Ghitany, M.E. The Unit-Weibull Distribution and Associated Inference. *J. Appl. Prob. Stat.* **2018**, *13*, 1–22.
40. Kachiashvili, K.J.; Melikdzhanjan, D.I. Estimators of the Parameters of Beta Distribution. *Sankhya B* **2019**, *81*, 350–373. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Probabilistic Forecasting of Crude Oil Prices Using Conditional Generative Adversarial Network Model with Lévy Process

Mohammed Alruqimi * and Luca Di Persio

Department of Computer Science, College of Mathematics, University of Verona, 37129 Verona, Italy; luca.dipersio@univr.it

* Correspondence: mohammed.alruqimi@univr.it

Abstract: Accurate crude oil price forecasting is essential, considering oil's critical role in the global economy. However, the crude oil market is significantly influenced by external, transient events, posing challenges in capturing price fluctuations' complex dynamics and uncertainties. Traditional time series forecasting models, such as ARIMA and LSTM, often rely on assumptions regarding data structure, limiting their flexibility to estimate volatility or account for external shocks effectively. Recent research highlights Generative Adversarial Networks (GANs) as a promising alternative approach for capturing intricate patterns in time series data, leveraging the adversarial learning framework. This paper introduces a Crude Oil-Driven Conditional GAN (CO-CGAN), a hybrid model for enhancing crude oil price forecasting by combining advanced AI frameworks (GANs), oil market sentiment analysis, and stochastic jump-diffusion models. By employing conditional supervised training, the inherent structure of the data distribution is preserved, thereby enabling more accurate and reliable probabilistic price forecasts. Additionally, the CO-CGAN integrates a Lévy process and sentiment features to better account for uncertainties and price shocks in the crude oil market. Experimental evaluations on two real-world oil price datasets demonstrate the superior performance of the proposed model, achieving a Mean Squared Error (MSE) of 0.000054 and outperforming benchmark models.

Keywords: crude oil-driven conditional generative adversarial network; Lévy–Merton jump-diffusion model; oil price forecasting; long short-term memory

MSC: 68T07

1. Introduction

Despite the significant success achieved by deep learning models in various fields, particularly in computer vision and natural language processing, their success in time series forecasting remains limited. Time series forecasting presents unique challenges primarily due to the high degree of uncertainty and the intricate dynamics influenced by external factors. These challenges are particularly amplified in the case of crude oil price forecasting, where the inherent volatility and sensitivity of oil prices to global events make the uncertainty and fluctuations even more pronounced. The fluctuations in oil prices directly impact the cost of production, transportation, and goods, influencing inflation and overall economic stability. Therefore, accurate forecasting of oil prices has been a critical concern over recent decades. Statistical methods such as autoregressive integrated moving average (ARIMA) models were commonly used in the early stages of oil price forecasting. These

methods are built on mathematical assumptions and rely on linear relationships within the data, making them effective for simple and stable patterns. However, as the need to capture the intricate and non-linear dynamics of oil price movements, the advent of deep learning (DL) models emerged as a more powerful alternative [1]. In the past decade, numerous machine learning (ML) models have been employed to enhance forecasting accuracy [2]. Recurrent Neural Networks (RNNs), particularly Gated Recurrent Units (GRUs) and Long Short-Term Memory (LSTM) networks, have gained popularity for predicting oil prices due to their ability to capture temporal dependencies [3]. However, several limitations arise due to the complex and volatile nature of the energy markets. These models may struggle to optimally integrate multiple exogenous data and capture sudden jumps in the data caused by exogenous factors. This can lead to challenges such as the model overfitting to noise rather than accurately capturing underlying patterns, especially when working with small or noisy financial datasets. Transformer-based models, which excel at capturing long-term dependencies, have gradually become popular in time series forecasting as well [4]. However, Transformers typically require larger datasets to be trained efficiently, and their performance in time series data is questionable [5]. The attention mechanism of the Transformer is somewhat against order, which is critical for time series forecasting. GANs [6] have gained increasing attention in the literature. GANs offer a novel approach with the potential for superior performance, mainly due to their advanced pattern recognition capabilities. These cutting-edge neural network architectures mitigate overfitting by introducing Gaussian-mixture noise into the generator network during training. In the past few years, many GAN models have been adapted for time series applications such as TimeGAN [7], COT-GAN [8], and Time-Variant GAN [9]. These models have been primarily designed for general time series data augmentation rather than forecasting. On the other hand, there are a few models that have been designed explicitly for forecasting purposes and applied in the context of financial time series data (i.e., ForGAN [10] and FinGAN [11]). In this paper, we introduce a crude oil-driven Conditional Generative Adversarial Network (CO-CGAN) to generate probabilistic forecasting of crude oil prices. Our model leverages the ForGAN architecture [10], which combines GANs with recurrent neural networks. This architecture renders GANs more suitable for forecasting tasks and places them into a supervised learning setting whilst producing conditional probability distributions of oil prices given previous historical observations. Our model (CO-CGAN) allows uncertainty estimates beyond the traditional use of point estimates in the forecasting literature. We address the challenge of insufficient data representing rare events affecting price fluctuations by integrating a Lévy–Merton jump process and a sentiment score feature derived from crude oil market sentiment analysis. These components enrich the CGAN model's context, allowing it to better capture the uncertainty and volatility dynamics of oil prices driven by external factors. Given the critical role of hyperparameter tuning in achieving optimal training, we employed the Grey Wolf Optimizer (GWO) to optimize the model's hyperparameters efficiently.

Extensive experiments were conducted on two real-world datasets, Brent and WTI. To evaluate the proposed CO-CGAN model performance, we compared our model to ForGAN, TimeGAN, and two other benchmark models specifically designed for this purpose based on Conditional GAN (CGAN) and Wasserstein-GAN (WGAN).

Standard point-wise evaluation metrics have been used, including MSE, RMSE, MAE, MAPE, and MSPE, as well as the statistical measure (R^2) and the Kullback–Leibler divergence (KL). The experiments demonstrate that our model achieves high forecasting accuracy and robustness compared to the benchmark models, achieving an MSE of (0.000054).

2. Generative Adversarial Networks (GANs)

GANs have rapidly gained attention since their introduction in 2014 [6], becoming a prominent area of interest within the deep learning research community. Their success can be attributed to their ability to generate and manipulate high-quality data across various domains. While GANs have significantly impacted computer vision, particularly in tasks like image and video generation, style transfer, and image completion, their application has extended to other areas, including natural language processing (NLP) and time series analysis. However, ongoing research explores using GANs for time series generation and forecasting [12]. A GAN consists of two primary components: a Generator G and a Discriminator D, typically implemented as neural networks (Figure 1). These two neural networks are trained through a unique adversarial process. The generator G takes in random noise $z \in \mathbb{R}^r$ and creates synthetic data similar to the distribution of the actual training data. At the same time, the discriminator D evaluates the authenticity of the data, distinguishing between actual and generated samples. During this training process, the generator strives to produce increasingly realistic samples capable of deceiving the discriminator. In contrast, the discriminator simultaneously improves its ability to differentiate between real and fake data. This adversarial process continues until the generator produces data that the discriminator can no longer reliably distinguish from the original. As a result, GANs can generate synthetic data that maintain the original dataset's statistical properties and distributional characteristics, making them highly effective for tasks requiring realistic data synthesis. Mathematically, a discriminator D and generator G play the following two-player minimax game with value function $V(G, D)$ [6]:

$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(\boldsymbol{z})))] \tag{1}$$

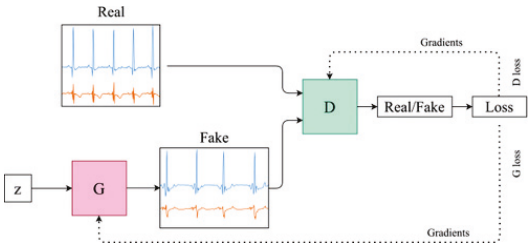


Figure 1. Generative adversarial network [12].

Since their introduction in 2014, numerous GAN variants have been developed to address specific challenges or expand the original framework's capabilities. Notable examples include Conditional GANs (CGANs) [13], which generate data conditioned on auxiliary information such as labels or classes, enabling controlled and targeted data generation. Bidirectional GAN (BiGAN) [14] further extends GANs by simultaneously learning generative and inference models, facilitating the mapping of data points back into the latent space. Other popular variants include Wasserstein GAN (WGAN) [15], which improves training stability by minimizing the Wasserstein distance.

3. Literature Review

Over recent decades, various methodologies have been developed for forecasting oil prices. Initially, statistical and econometric approaches, such as ARIMA (AutoRegressive Integrated Moving Average) and GARCH (Generalized AutoRegressive Conditional Heteroskedasticity), were widely employed. These models often rely on mathematical assumptions and have been effective in capturing linear patterns in time series data. Recently,

deep learning models have gained popularity in the literature due to their capabilities in capturing more complex and non-linear patterns in datasets. These models also enable a better incorporation of external factors [1]. Ref. [2] comprehensively compares and analyses a wide range of machine learning-based approaches for energy and financial forecasting. In our previous work [3], we empirically compared deep learning models for crude oil price forecasting, focusing on approaches based on LSTM, GRU, and Transformer networks, which are widely applied in the literature. LSTM and GRU models have been commonly applied in the literature [16] due to their attribution in maintaining temporal dependencies, even when working with small datasets. Additionally, researchers have investigated combining LSTM and GRU models with other techniques such as Convolutional Neural Networks (CNNs) [17], time series decomposition [16], and attention layers to enhance their performance. Other researchers have explored Transformer-based models, such as Autoformer [18], for general time series forecasting. While these models achieve promising accuracy with high-frequency time series datasets, their performance with oil price datasets is not as impressive. This can be attributed to the fact that the attention mechanism in Transformers does not maintain order, which is critical for this task. Additionally, Transformer training typically requires a larger dataset [3,5]. Generative Adversarial Network (GAN) models have emerged as robust tools for generating synthetic data distributed similarly to the original data. This advancement opens new doors for investigating the use of these models in time series forecasting. In the past few years, several GAN-based models have been introduced to generate synthetic time series data, for instance: TimGAN [7], COT-GAN [8] and Time-Variant GAN [9]. While these models are primarily designed for time series data augmentation, a few other GAN-based models have been proposed for time series forecasting tasks, such as TSF-CGANs [19], ForGAN [10], and Fin-GAN [11]. TSF-CGANs [19] is a time series forecasting model based on CGAN combined with convolutional neural networks (CNNs) and bi-directional long short-term memory (Bi-LSTM). ForGAN is a conditional GAN-based model that was presented in 2019 [10] for probabilistic time series forecasting. This model combines a gated recurrent unit (GRU) or a long short-term memory (LSTM) cell, depending on the data. Both the generator and the discriminator first pass their inputs through a recurrent neural network layer aiming at exploiting the time series structure. The Fin-GAN [11] model employs the ForGAN architecture by introducing an economic-driven loss function for the generator. Examining these models with crude oil price datasets has not yet been investigated well.

4. Proposed Model

The proposed model, built on the ForGAN architecture [10], is a one-step-ahead probabilistic forecasting model applying a conditional generative adversarial network to model the probability distribution of forthcoming crude oil prices based on past observations. We further integrate a Lévy–Merton jump process into the generator and crude oil market sentiment analysis as an additional, conditional input to improve the modelling of price shocks and volatility.

The leading architecture of the model is illustrated in Figure 2. In other words, the objective is to model the probability distribution of one-step-ahead crude oil price x_{t+1} given the historical data $c = x_0, \ldots, x_t$, i.e., $\rho_{(xt+1|c)}$. The historical data include crude oil prices and sentimental score observations.

Generator G and discriminator D are trained simultaneously in an adversarial network. The generator takes a noise vector Z, which is sampled from a Gaussian distribution (ρ_Z), and generates the value of the next step (x_{t+1}) with the condition data ρ_data. Meanwhile,

the discriminator receives (x_{t+1}) and determines if (x_{t+1}) is real or generated by the generator. Hence, the model function is expressed as follows:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x_{t+1}\sim\rho_{\text{data}}(x_{t+1})}[\log D(x_{t+1} \mid c)] + \mathbf{E}_{z\sim\rho_x(z)}[\log(1 - D(G(z \mid c)))] \tag{2}$$

In other words, the generator G learns to transform a known probability distribution ρ_z into the generator distribution ρ_G, which resembles ρ_{data}.

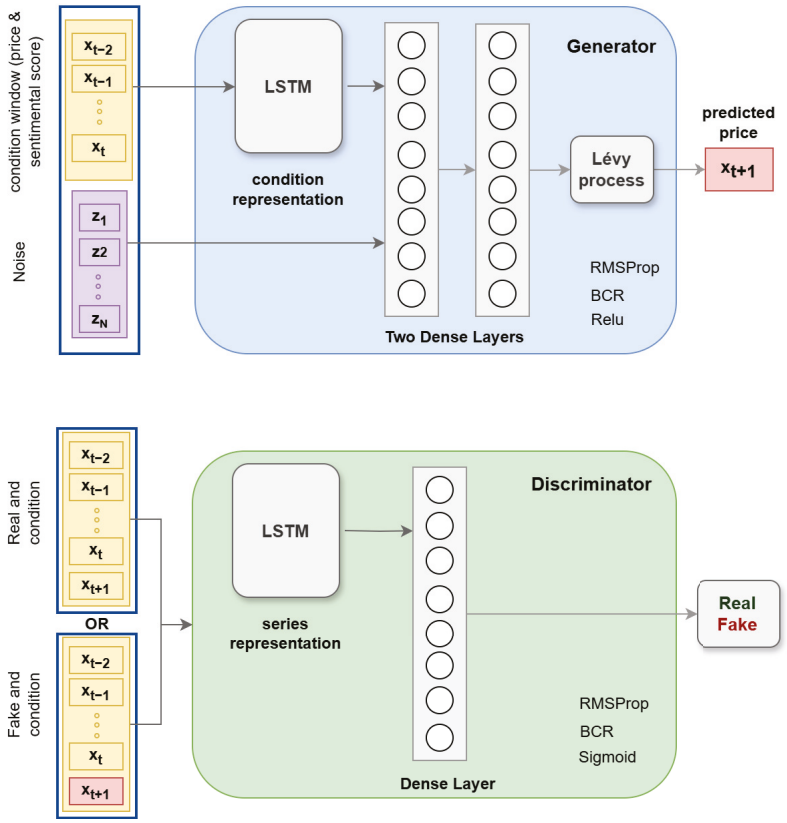


Figure 2. Overview of the proposed model.

4.1. Conditional GAN (CGAN)

Conditional GANs (CGANs) [13], an extension of traditional GANs, provide additional control over the generation process by conditioning the model on supplementary information, denoted as y. This information can be any data, such as labels or data from different sources. By incorporating this conditioning information, CGANs can generate samples that adhere to predefined characteristics, enhancing their ability to generate data from conditional distributions. This results in more targeted and context-specific data generation, which means placing the GAN into a supervised forecasting setting. In other words, rather than having all data samples originate from the same underlying distribution, we assume that the data $\mathbf{x}$ are drawn from a conditional distribution $p_{\text{data}}(\mathbf{x} \mid \text{condition(y)})$, where the distribution depends on a specific deterministic condition—see Figure 2. Both the generator G and the discriminator D are provided with additional deterministic information (condition) alongside the standard inputs. In our case, the sub-sequences (time lags) of oil prices and the sentiment scores represent the labelled data condition. The model

will generate (predict) future oil prices based on past observations (time lag). The modified value function $V(G, D)$ for this configuration is the following:

$$\begin{aligned} &\min_G \max_D V(D,G) = \mathbb{E}_{x\sim\rho_{\text{data}}(x)}[\log D(x \mid y)] \\ &+ \mathbb{E}_{z\sim\rho_x(z)}[\log(1 - D(G(z \mid y)))] \end{aligned} \tag{3}$$

4.2. Lévy–Merton Jump-Diffusion Model

To capture the abrupt price changes and high volatility often observed in crude oil prices, we incorporate the Lévy–Merton jump-diffusion model [20] into our model. Hence, extending the classical Geometric Brownian Motion (GBM) by adding a jump component to capture price volatility, thereby enhancing the accuracy of our forecasts. In particular, let S_t represent the crude oil price at time t, then its evolution under the Merton jump-diffusion model is governed by the following SDE:

$$dS_t = S_t((\mu - \lambda k)dt + \sigma dW_t + dQ_t) \tag{4}$$

where we can see the action of two primary components (see Figure 3):

- Diffusion component: σdW_t, where W_t represents a standard Wiener process, with μ as the drift (expected return) and σ as the volatility. This part captures the *continuous* price changes following a standard GBM.
- Jump component: dQ_t, modelled as a compound Poisson process to account for discontinuous jumps:

$$dQ_t = \sum_{i=1}^{N_t}(Y_i - 1) \tag{5}$$

where N_t is a Poisson process with intensity λ, representing the number of jumps in the interval, and Y_i are i.i.d. random variables representing the jump sizes, which follow a log-normal distribution: $\ln(Y_i) \sim \mathcal{N}(m, \delta^2)$.
Accordingly, the term $k = \mathbb{E}(Y_i - 1) = e^{m+\frac{1}{2}\delta^2} - 1$ represents the expected relative jump size, and the parameter λ is the frequency of jumps. Let us underline that since the expectation of Q_t is

$$\mathbb{E}(Q_t) = \lambda k t, \tag{6}$$

$Q_t - \lambda k t$ is a martingale, preserving the no-arbitrage condition of the model.

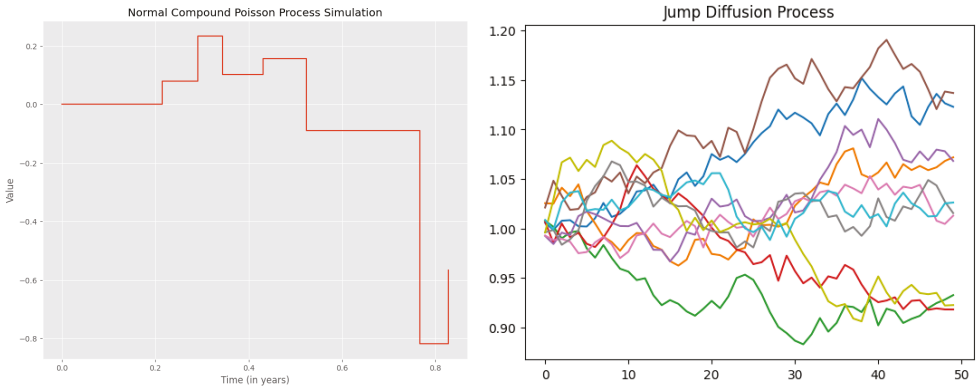


Figure 3. (**Left**): Normal compound Poisson process simulation. λ = 10; T = 1y; jumps $\sim$ N(0, 0.22). Ten jumps per year on average, and magnitude of jumps follows normal distribution N(0, 0.22). (**Right**): Jump paths generated using Merton model (10 paths, 50 steps).

The Lévy–Merton jump-diffusion model is embedded within the CGAN generator to enhance the forecast distribution by incorporating the jump dynamics of crude oil prices. In particular, during training, the jump-diffusion parameters $(\mu, \sigma, \lambda, m, \delta)$ are optimized through the CGAN's neural network gradients, allowing the model to adjust to the complex volatility structure in real time, which implies more accurate probabilistic forecasts and better identification of the risk of extreme price movements.

Essentially, our objective is to generate one-step-ahead forecasts, x_{t+1}, conditioned on historical observations $c = (x_0, x_1, \ldots, x_t)$. Accordingly, we integrate the Lévy–Merton model within the CGAN framework to handle both continuous market fluctuations and discrete jumps, Hence, this ensures a more realistic representation of the crude oil price dynamics.

In particular, let $G(z|c)$ be the generator of the CGAN, where $z \sim \mathcal{N}(0,1)$ represents Gaussian noise that is injected into the network. We modify the generator's output to incorporate the dynamics of the Lévy–Merton model. Indeed, the generator outputs the predicted value $\hat{x}_{t+1}$, where

$$\hat{x}_{t+1} = G(z|c) + J_L(\lambda, m, \delta) \tag{7}$$

with $J_L(\lambda, m, \delta)$ being the Lévy jump term modelling the sudden price shifts, defined as follows:

$$J_L(\lambda, m, \delta) = \sum_{i=1}^{N_t} (Y_i - 1) \tag{8}$$

where $N_t \sim \text{Poisson}(\lambda)$ is the number of jumps in the time interval, and $Y_i \sim \log \mathcal{N}(m, \delta^2)$ are the jump magnitudes. Then, such jumps are added to the generated value to account for extreme events in crude oil prices, while the generator learns both the continuous and jump dynamics during training by optimizing the parameters of the Lévy–Merton model, including λ (jump intensity), m (mean jump size), and δ (jump volatility).

More specifically, since the core of the CGAN training process is the adversarial min–max game between the generator G and the discriminator D with the standard CGAN objective defined as

$$\min_G \max_D V(D,G) = \mathbb{E}_{x_{t+1} \sim \rho_{\text{data}}(x_{t+1})}[\log D(x_{t+1}|c)] + \mathbb{E}_{z \sim \rho_z(z)}[\log(1 - D(G(z|c)))], \tag{9}$$

to integrate the Lévy–Merton jump-diffusion model into this objective, we update the generator to output $\hat{x}_{t+1}$ as follows:

$$\hat{x}_{t+1} = G(z|c) + J_L(\lambda, m, \delta). \tag{10}$$

Thus, the value function becomes

$$\min_G \max_D V(D,G) = \mathbb{E}_{x_{t+1} \sim \rho_{\text{data}}(x_{t+1})}[\log D(x_{t+1}|c)] + \mathbb{E}_{z \sim \rho_z(z)}[\log(1 - D(G(z|c) + J_L(\lambda, m, \delta)))] \tag{11}$$

The discriminator $D(x_{t+1}|c)$ evaluates the authenticity of the predicted value x_{t+1}, determining whether it comes from the true distribution ρ_{data} or is generated by the model $G(z|c) + J_L(\lambda, m, \delta)$. Let us underline that the latter ensures that the generated values include both the continuous price changes (modelled by the CGAN) and the sudden jumps (modelled by the Lévy process).

Moreover, during training, the CGAN learns the optimal parameters of the generator and discriminator. Simultaneously, the parameters of the Lévy–Merton jump-diffusion model—namely, λ, m, and δ—are optimized as part of the generator's output to ensure

that the jump component aligns with the observed data distribution. Specifically, the loss function for the generator is updated to include the jump component:

$$\mathcal{L}_G = -\mathbb{E}_{z\sim\rho_z(z)}[D(G(z|c) + J_L(\lambda, m, \delta))] \tag{12}$$

Furthermore, the Lévy process parameters are updated via backpropagation RMSProp, with the gradients derived from the adversarial loss:

$$\frac{\partial \mathcal{L}_G}{\partial \lambda}, \frac{\partial \mathcal{L}_G}{\partial m}, \frac{\partial \mathcal{L}_G}{\partial \delta} \tag{13}$$

with the latter ensuring that the jump magnitudes and frequencies are tuned dynamically during the training process. Hence, the generator can capture typical market behaviour and rare, large-scale events.

The last step consists of considering that the conditional probability distribution generated by the model, i.e., $\rho_G(x_{t+1}|c)$, is a convolution of the generator's output distribution and the Lévy process distribution. Let $\rho_L(x_{t+1})$ represent the distribution induced by the jump process; then, the overall conditional distribution of x_{t+1}, generated by our model, can be expressed as follows:

$$\rho_G(x_{t+1}|c) = \int \rho_G(\hat{x}_{t+1}|c) \cdot \rho_L(x_{t+1} - \hat{x}_{t+1}) d\hat{x}_{t+1}. \tag{14}$$

The latter convolution ensures that the model captures both the smooth price changes modelled by $\rho_G(\hat{x}_{t+1}|c)$ as well as the discontinuous price jumps modelled by $\rho_L(x_{t+1} - \hat{x}_{t+1})$.

4.3. Accumulative Sentimental Score

Accumulative Sentimental Score (SENT) is a cumulative sentimental score derived from the CrudeBERT_Plus model, as introduced in [21]. This model is a specialized variant of FinBERT, fine-tuned to evaluate the influence of market events on crude oil prices. Its development is grounded in the principles of Adam Smith's theory of supply and demand, emphasizing the interplay between market sentiment and pricing dynamics. Market sentiment plays a pivotal role in shaping oil price movements, as it reflects collective expectations and reactions to geopolitical events, economic fluctuations, and market trends. Sentiments expressed in news headlines, reports, and market commentaries can amplify or mitigate future supply and demand perceptions, influencing trading behaviours and price volatility. The dataset used for training CrudeBERT_Plus consists of headlines from 1034 unique news providers. The largest portion of this dataset originates from Dow Jones newswires (approximately 21,200 headlines), followed by Reuters (around 3000 headlines), Bloomberg (approximately 1100 headlines), and Platts (approximately 870 headlines). Details about generating this sentiment score are thoroughly discussed in [21]; the data are publicly available and can be reached through the following link: https://github.com/fhgr/crudeoil-sentiment2022-dataset (accessed on 10 October 2024).

5. Benchmark Models

To assess the predictive performance of our model, we compare it against two established models: ForGAN and TimGAN.

Furthermore, we developed benchmark models based on common GAN architectures adapted for time series forecasting tasks: Conditional GAN (CGAN) and Conditional Wasserstein GAN with Gradient Penalty (CWGAN-GP). The following list provides a comprehensive overview of the benchmark models employed in this evaluation.

1. Conditional GAN (CGAN): This model is built based on the vanilla conditional GAN model [13], with both the generator and discriminator incorporating a Gated Recurrent Unit (GRU). However, this model does not incorporate a Lévy stochastic process.
2. CWGAN-GP Model: We built this model upon the foundational CWGAN-GP (Conditional Wasserstein GAN with Gradient Penalty) architecture, incorporating enhancements to improve its performance and robustness. Specifically, we integrated a Lévy process, as described in Section 4.2, and LipSwish activation. Wasserstein GAN (WGAN) is an improvement over the traditional GAN framework, addressing issues of training stability and mode collapse. By using the Wasserstein distance (also known as Earth Mover's distance) as a measure of distribution similarity, WGAN provides smoother gradients for the generator, leading to more reliable convergence during training. The model comprises the following key components:
   LSTM units: The CWGAN-GP model integrates LSTM units in its generator and critic components. This design leverages the ability of LSTMs to capture long-term dependencies and complex temporal patterns in sequential data.
   Wasserstein Loss: The critic (discriminator) loss uses the Wasserstein loss, which is simply the difference between the real and fake sample scores:

$$d_loss = d_real_loss + d_fake_loss + \lambda_{gp} \cdot gradient_penalty \tag{15}$$

   The generator loss is the negative of the discriminator's prediction of the generated data:

$$g_loss = -torch.mean(d_g_decision). \tag{16}$$

   Gradient Penalty: The model includes a gradient penalty term to enforce the Lipschitz constraint, which is crucial for WGAN-GP.
   No Critic Sigmoid Activation: The critic does not use a sigmoid activation in the output layer. Instead, it outputs raw scores, which are then used to compute the Wasserstein distance.
   LipSwish Activation: LipSwish activations are used to constrain the Lipschitz constant of the discriminator [22]. Adding this layer to the generator and the discriminator improves the training stability.
3. ForGAN Model: ForGAN is a conditional GAN designed for probabilistic time series forecasting, introduced by [10]. It leverages a conditional GAN to forecast x_{t+1} given the previous L values of the series $x_t, x_{t-1}, \ldots, x_{t-(L-1)}$. This approach utilizes more information than standard point estimates, facilitating uncertainty estimates. The generator and discriminator use a recurrent neural network layer to exploit the time series structure. ForGAN can include either a GRU or an LSTM cell.
4. TimeGAN: TimeGAN is a generative adversarial network specifically designed for time series data. The model introduced by [7] has garnered significant attention. It utilizes an embedding network to map time series data into a high-dimensional latent space and a recovery network to map it back. This model incorporates a recurrent architecture to capture temporal dependencies, and a joint optimization approach ensures that both the adversarial and reconstruction losses are minimized, leading to more realistic and temporally consistent synthetic time series data.

6. Experiments

To evaluate the performance of our model, we tested it using two real-world datasets that exhibit high fluctuation dynamics. We compared our results with those of state-of-the-

art models in the literature. We will introduce the datasets, outline the evaluation methods, describe the experimental setup, and discuss the hyperparameter fine-tuning process.

6.1. Dataset

We used two real-world datasets for the experiments: crude WTI and crude Brent prices. These datasets encompass 2386 observations spanning from 3 January 2012 to 1 April 2021, including the period during which the COVID-19 pandemic impacted the oil and stock markets. To enhance our model for estimating the volatility, we incorporated a daily cumulative sentimental score (SENT) with the two datasets as external features. The SENT feature is obtained from the CrudeBERT model described in [21]. The period from 2012 to 2021 was chosen for analysis due to the availability of sentiment scores during this time frame. Figures 4 and 5 visualize the dataset and the correlation analysis of the three time series (Brent, WTI, and the sentimental index).

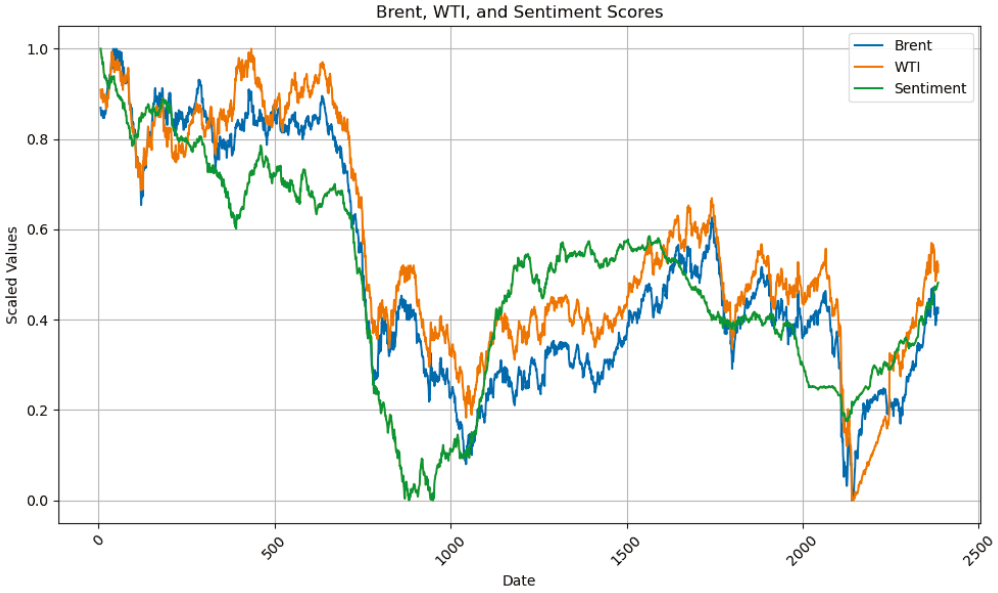


Figure 4. Datasets: WTI, Brent, and the sentimental index.

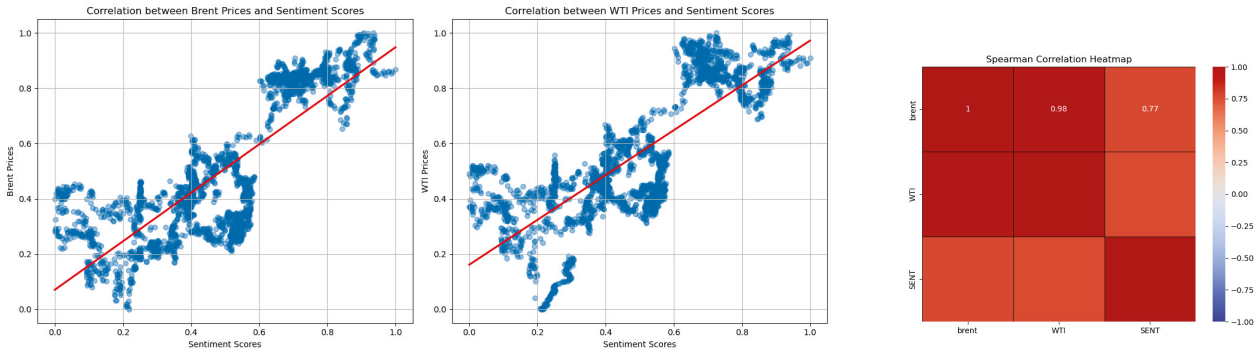


Figure 5. Correlation analysis of the dataset features.

6.2. Evaluation Metrics

Point-wise error metrics are commonly used in forecasting tasks. We report MAE, MSE, RMSE, MSPE, and MAPE to compare our model with state-of-the-art models, in addition to R^2 and KL divergence measures. Below, we define each measure along with its formula, assuming that Y represents the actual values and $\hat{Y}$ represents the predicted values.

- Mean absolute error (MAE): Represents the average of the absolute difference between the actual and predicted values. This measure is less sensitive to outliers.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right| \tag{17}$$

- Mean Squared Error (MSE): Measures the average squared difference between the predicted and actual values. This measure emphasizes larger errors.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{18}$$

- Root Mean Squared Error (RMSE): Square root of MSE.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2} \tag{19}$$

- Mean Squared Prediction Error (MSPE): Quantifies the average squared difference between predicted and actual values.

$$MSPE(Y, \hat{Y}) = \frac{1}{n}\sum_{i}^{n}(Y_i - \hat{Y}_i)^2 \tag{20}$$

- MAPE: Presents the percentage error between actual and predicted values. It is useful for forecasting and percentage-based error analysis.

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{Y_i - \hat{Y}_i}{Y_i}\right| \tag{21}$$

- R^2 (Coefficient of Determination): Measures the proportion of variance between the actual and predicted values. A high R^2 (close to 1) means the model very closely predicts the actual values.

$$R^2 = 1 - \frac{SSR}{SST} \tag{22}$$

  where SSR stands for the sum of squared residuals, which measures the sum of the squared differences between actual and predicted values. SST represents the Sum of the Squared Total, representing the total variance in the data. It is the sum of the squared differences between each actual value and the mean of the actual values.
- Kullback–Leibler Divergence (KL Divergence): Measures the distribution similarity between the actual and generated data. The KL divergence between two distributions Q and P is formulated using the following notation: $KL(P||Q)$, where the $||$ operator indicates "divergence" or Ps divergence from Q.

6.3. Hyperparameters and Model Training

We conducted extensive experiments to determine the best hyperparameters for the model. To avoid excessive complexity, we trained the generator independently as a standalone model, applying the Grey Wolf Optimizer (GWO) to find near-optimal hyperparameters. These parameters include window size, optimizer type, learning rate, dropout rate, and the number of hidden units and layers. However, the Lévy parameters, σ and λ, were optimized through neural network gradients, with (0.02) as an initial value for both.

The table below shows the hyperparameters and their selected values for the generator model. We used the same parameters in the ForGan model for the discriminator with the GRU cell to avoid further hyperparameter fine-tuning. Our strategy was not to use an early stopping mechanism. Instead, we fixed the number of epochs at 10,000 and saved the best-performing model at the training stage. We also tested generating the noise vector for the generator using different methods: Geometric Brownian Motion (GBM), uniform noise, Gaussian noise, and Poisson noise. All experiments were repeated three times and implemented in PyTorch v.2.4.

7. Results and Discussion

Table 1 presents the optimal configuration we found during the training or by the GWO. In particular, the Merton parameters (σ represents the volatility of the diffusion, and λ represents the jump frequency) were calibrated using a simple neural network during the training process. The absolute values of σ and λ were used to generate the Merton path. Meanwhile, the other parameters were determined through optimization with the GWO. The numerical results achieved are summarized in Tables 2 and 3. Figures 6 and 7 plot the actual and forecasted prices by the proposed model. The first chart from the left (in Figures 6 and 7) presents a line plot comparing the predicted and actual values, while the second chart visualizes the same comparison using a scatter plot. The third chart highlights a distributional comparison between the actual and generated values. These visualizations demonstrate the predicted values' closeness to the actual data, showcasing the model's high accuracy. Tables 2 and 3 illustrate the proposed model's evaluation compared to several benchmark models (CGAN, CWGAN-GP, ForGAN, and TimeGAN) on two datasets: Brent and WTI. The tables present model performance metrics (MAE, MSE, RMSE, MSPE, MAPE, R^2, and KL) for Brent and WTI datasets, respectively. The proposed model outperforms the others across most metrics, indicating superior predictive performance and lower error rates. Table 4 presents a comparative analysis of the models' average runtime (in seconds). Among the models, CGAN has the shortest runtime, followed by the proposed model, while TimeGAN demonstrates a significantly higher runtime. It is worth mentioning that this represents the training runtime of the models using the optimal configuration assigned manually. However, it does not include the runtime spent for the GWO optimization process. The experiments were conducted on a Windows 11 environment with a Core i5 processor featuring eight cores and 16 GB of RAM; the implementation used the PyTorch framework.

Table 1. Generator hyperparameters and their assigned values.

Hyperparameter	**Value**
Optimiser	RMSProp
Loss Function	Binary Cross-Entropy Loss (BCE)
Activation Function	Relu
Learning Rate	0.0034
Batch Size	32/16
Window Size	10/15
Dropout	0.33
Torch Fixed Seed	4
Noise Dimension	32
Noise Type	Gaussian
LSTM Hidden Dimension	8
Lévy σ and λ (Brent)	0.0891 and 0.0302
Lévy σ and λ (WTI)	0.3181 and -0.1201

Table 2. Model evaluation metrics—Brent dataset.

Model	MAE	MSE	RMSE	MSPE	MAPE	R^2	KL
Our Model	0.004800	0.000054	0.007348	0.000455	0.013696	0.989210	0.000139
CGAN	0.00700	0.000103	0.010198	0.000991	0.021051	0.979220	0.000325
CWGAN-GP	0.008050	0.000118	0.010886	0.001990	0.031507	0.981623	0.000691
ForGAN	0.011500	0.000207	0.014387	0.002198	0.035774	0.958641	0.000460
TimeGAN	0.273225	0.114837	0.338876	21.167172	1.142248	−0.909833	0.296663

Table 3. Model evaluation metrics—WTI dataset.

Model	MAE	MSE	RMSE	MSPE	MAPE	R^2	KL
Our Model	0.005600	0.000062	0.007874	0.000293	0.012174	0.987919	0.000069
CGAN	0.007300	0.000109	0.010440	0.000597	0.016878	0.980206	0.000279
CWGAN-GP	0.009700	0.000176	0.013266	0.002869	0.037412	0.976150	0.001153
ForGAN	0.010600	0.000188	0.013711	0.001140	0.025341	0.963368	0.000305

Table 4. Run time analysis.

Model	Run Time Average
Our Model	196.152
CGAN	152.752
CWGAN-GP	1088.501
ForGAN	301.539
TimeGAN	4076.762

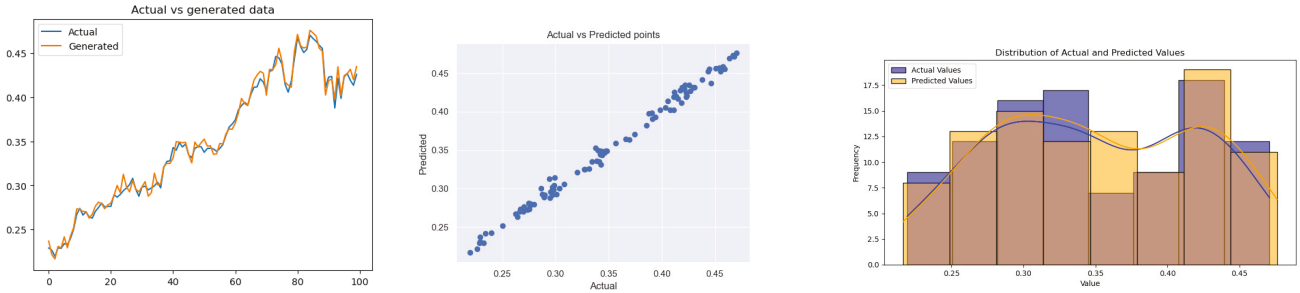


Figure 6. Evaluations of proposed model: actual vs. generated values for Brent dataset.

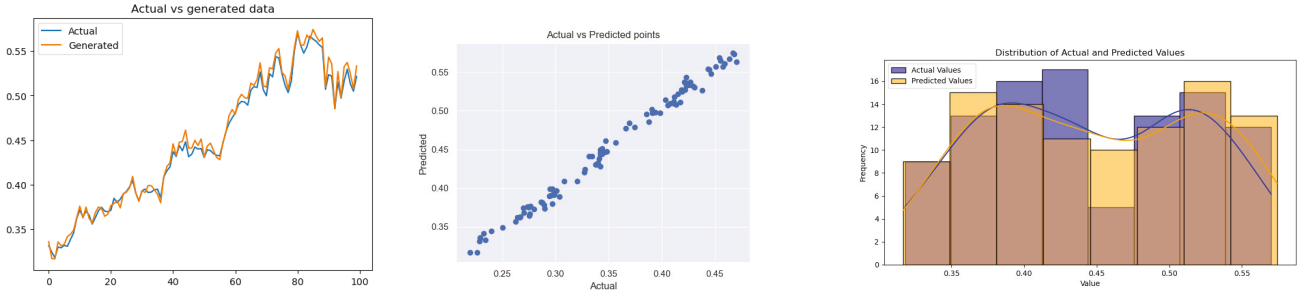


Figure 7. Evaluations of proposed model: actual vs. generated values for WTI dataset.

Figures 8 and 9 provide a comparative analysis of actual vs. forecasted values for the Brent and WTI datasets, respectively. Each figure displays the performance of the proposed model alongside baseline models (CGAN, CWGAN-GP, and ForGAN). The plots illustrate how closely the generated values from each model track the actual values over time. In both datasets, the proposed model demonstrates a high level of accuracy, with generated

values closely following the actual data trends, showcasing its effectiveness in time series forecasting compared to the baseline models. Figure 10 comprehensively analyses the distribution and correlation of generated values vs. actual values for the Brent dataset. On the left, the scatter plot matrix shows the pairwise relationships between the actual and generated values across different models (proposed model, CGAN, CWGAN-GP, ForGAN). Each plot highlights the distribution alignment, indicating how closely the generated values follow the data distribution. On the right, the heatmap displays the correlation matrix, with the proposed model showing the highest correlation (0.997) with the actual values, demonstrating its effectiveness in accurately replicating the data distribution compared to other models. Additionally, incorporating the Lévy process enhanced forecasting accuracy, as indicated by a lower MSE than the base CGAN model. Our empirical findings indicate that dedicating time to tuning model parameters yields better results than focusing on constructing more complex model architectures. Therefore, we applied the Grey Wolf Optimizer (GWO) to optimize our model parameters effectively.



Figure 8. Comparison of actual values with predicted values—proposed model and baseline models—Brent dataset.



Figure 9. Comparison of actual values with forecasts from proposed model and baseline models—WTI dataset.

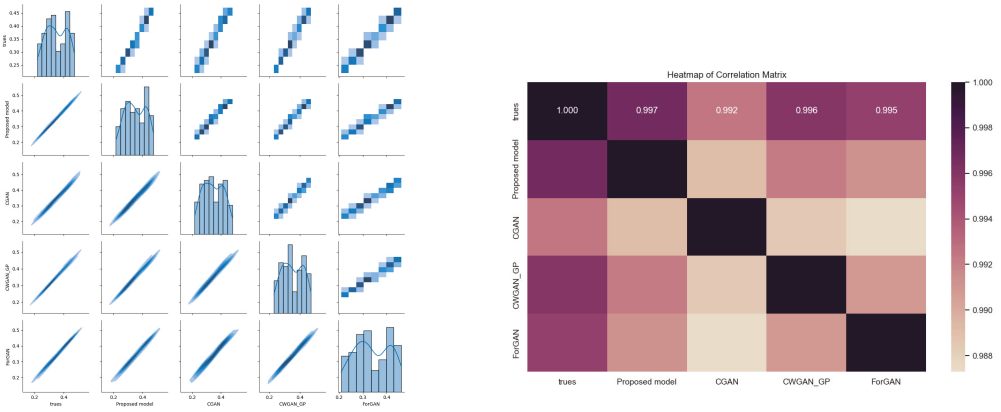


Figure 10. Brent dataset—distribution and correlation analysis of generated values.

8. Conclusions and Future Works

This work introduced a crude oil-driven conditional GAN model for oil price forecasting. Our primary contribution lies in enhancing oil price prediction performance by comprehensively addressing multiple facets of the forecasting task, leveraging diverse applications and cutting-edge techniques. The proposed model uses historical prices and sentimental scores as conditions. It integrates a Lévy–Merton jump-diffusion process with parameters learned through neural network gradients. We optimized the model parameters using the GWO. The model was evaluated on two real-world datasets (crude Brent and crude WTI), demonstrating accurate forecasts and outperforming established models across different test periods that exhibited varying movements. Additionally, incorporating the Lévy process improved forecasting accuracy, as evidenced by lower MSE than the base CGAN model. However, the lack of data related to sentiment market analysis limited our experiments to a specific period (from 2012 to 2021). We employed a simple neural network to calibrate the Merton model variables to reduce computational costs. More advanced differentiable SDE solvers, see, e.g., [22], could also be utilized for this purpose. Similarly to how GANs have been adapted for time series forecasting via the conditional GAN variant, LLMs could be explored for similar applications by reprogramming the input dataset and employing a robust prompt engineering methodology. Additionally, these models can provide a broad context through embedded knowledge and web crawling for relevant events, which can significantly enhance their generalization and predictive performance, particularly within the context of data scarcity.

Author Contributions: Conceptualization, M.A. and L.D.P.; Methodology, M.A. and L.D.P.; Software, M.A.; Investigation, M.A.; Writing—original draft, M.A.; Writing—review & editing, L.D.P.; Supervision, L.D.P.; Project administration, L.D.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in the study are openly available at https://github.com/Med-Rokaimi/GAN_synthetic_data_time_series (accessed on 10 October 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Foroutan, P.; Lahmiri, S. Deep learning systems for forecasting the prices of crude oil and precious metals. *Financ. Innov.* **2024**, *10*, 111. [CrossRef]
2. Ghoddusi, H.; Creamer, G.G.; Rafizadeh, N. Machine learning in energy economics and finance: A review. *Energy Econ.* **2019**, *81*, 709–727. [CrossRef]
3. Alruqimi, M.; Di Persio, L. Enhancing Multi-step Brent Oil Price Forecasting with Ensemble Multi-scenario Bi-GRU Networks. *Int. J. Comput. Intell. Syst.* **2024**, *17*, 225. [CrossRef]
4. Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; Sun, L. Transformers in Time Series: A Survey. *arXiv* **2023**, arXiv:2202.07125.
5. Zeng, A.; Chen, M.H.; Zhang, L.; Xu, Q. Are Transformers Effective for Time Series Forecasting? In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 22 February–1 March 2022.
6. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*; Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K., Eds.; Curran Associates, Inc.: Newry, UK, 2014; Volume 27.
7. Yoon, J.; Jarrett, D.; van der Schaar, M. Time-series Generative Adversarial Networks. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Newry, UK, 2019; Volume 32.
8. Xu, T.; Wenliang, L.K.; Munn, M.; Acciaio, B. COT-GAN: Generating Sequential Data via Causal Optimal Transport. *arXiv* **2020**, arXiv:2006.08571.
9. Asre, S.; Anwar, A. Synthetic Energy Data Generation Using Time Variant Generative Adversarial Network. *Electronics* **2022**, *11*, 355. [CrossRef]
10. Koochali, A.; Schichtel, P.; Dengel, A.; Ahmed, S. Probabilistic Forecasting of Sensory Data with Generative Adversarial Networks—ForGAN. *IEEE Access* **2019**, *7*, 63868–63880. [CrossRef]
11. Milena Vuletić, F.P.; Cucuringu, M. Fin-GAN: Forecasting and classifying financial time series via generative adversarial networks. *Quant. Financ.* **2024**, *24*, 175–199. [CrossRef]
12. Brophy, E.; Wang, Z.; She, Q.; Ward, T. Generative Adversarial Networks in Time Series: A Systematic Literature Review. *ACM Comput. Surv.* **2023**, *55*, 199. [CrossRef]
13. Mirza, M.; Osindero, S. Conditional Generative Adversarial Nets. *arXiv* **2014**, arXiv:1411.1784.
14. Jaiswal, A.; AbdAlmageed, W.; Wu, Y.; Natarajan, P. Bidirectional Conditional Generative Adversarial Networks. *arXiv* **2018**, arXiv:1711.07461.
15. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein GAN. *arXiv* **2017**, arXiv:1701.07875.
16. Busari, G.A.; Lim, D.H. Crude oil price prediction: A comparison between AdaBoost-LSTM and AdaBoost-GRU for improving forecasting performance. *Comput. Chem. Eng.* **2021**, *155*, 107513. [CrossRef]
17. Kim, G.I.; Jang, B. Petroleum Price Prediction with CNN-LSTM and CNN-GRU Using Skip-Connection. *Mathematics* **2023**, *11*, 547. [CrossRef]
18. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. *arXiv* **2021**, arXiv:2106.13008.
19. Huang, X.; Li, Q.; Tai, Y.; Chen, Z.; Liu, J.; Shi, J.; Liu, W. Time series forecasting for hourly photovoltaic power using conditional generative adversarial network and Bi-LSTM. *Energy* **2022**, *246*, 123403. [CrossRef]
20. Merton, R.C. Option pricing when underlying stock returns are discontinuous. *J. Financ. Econ.* **1976**, *3*, 125–144. [CrossRef]
21. Kaplan, H.; Weichselbraun, A.; Braşoveanu, A.M.P. Integrating Economic Theory, Domain Knowledge, and Social Knowledge into Hybrid Sentiment Models for Predicting Crude Oil Markets. *Cogn. Comput.* **2023**, *15*, 1355–1371. [CrossRef] [PubMed]
22. Kidger, P.; Foster, J.; Li, X.; Oberhauser, H.; Lyons, T. Neural SDEs as Infinite-Dimensional GANs. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Prediction of Lithium Carbonate Prices in China Applying a VMD–SSA–LSTM Combined Model

Wenyi Wang [1], Haifei Liu [1], Lin Jiang [2] and Lei Wang [2,*]

[1] School of Management and Engineering, Nanjing University, Nanjing 210093, China
[2] School of Economics and Management, Nanjing Tech University, Nanjing 211816, China
* Correspondence: leiwang77777@163.com

Abstract: Given the highly nonlinear and unstable characteristics of lithium carbonate prices in China's lithium battery industry chain, the accuracy of a single prediction model is limited. This study introduces the prices of related materials in the lithium battery industry and macro-environmental indicators as key influencing factors. This study utilizes Variational Mode Decomposition (VMD) and the Sparrow Search Algorithm (SSA) to further develop the Long Short-Term Memory (LSTM) network, resulting in a VMD–SSA–LSTM combination model for predicting lithium carbonate pricing. The research results indicate that (1) using the VMD to decompose the time series of the original lithium carbonate prices can accurately extract the core features of the prices and significantly weaken the instability of the data; (2) by leveraging SSA to perform global optimization on the three parameters of the LSTM model and fitting the optimal parameters into the LSTM network, the generalization ability and robustness of the model are enhanced; (3) on the lithium carbonate dataset, the VMD–SSA–LSTM model outperforms the typical LSTM and VMD–LSTM models, achieving the lowest prediction error and a goodness-of-fit (R^2) of 0.9880, demonstrating a higher prediction accuracy for lithium carbonate prices. This study presents more precise benchmarks for resource optimization and price decisions in the lithium carbonate industry.

Keywords: prediction of lithium carbonate prices; variational mode decomposition; sparrow search algorithm; long short-term memory network

MSC: 35Q86

1. Introduction

In recent years, with the growing global focus on green energy, especially the rapid development of the new energy vehicle industry, the demand for lithium batteries has significantly increased [1]. Meanwhile, given that lithium carbonate is one of the key raw materials for lithium battery manufacturing, this has led to a sharp increase in market demand for lithium carbonate materials. China has emerged as the greatest market for new energy vehicles worldwide, accounting for nearly three-quarters of the global market share. With such a large volume of new energy vehicles, China not only has a strong demand for lithium carbonate but is also highly sensitive to its price fluctuations. Specifically, significant price changes in lithium carbonate can directly impact the costs of China's battery manufacturers as well as the entire cost structure and market competitiveness of China's new energy automotive industry.

At present, the price of lithium carbonate in China has been on a continuous downward trend after experiencing a round of high prices. This reflects a fundamental shift that has

begun to occur in the supply and demand structure of the lithium carbonate industry. Especially in the context of strong demand for new energy batteries, a large number of Chinese lithium carbonate enterprise executives violate the law of market competition in order to pursue short-term high earnings. These executives' blind and overconfident decision-making results in a large number of short-term influxes in production capacity in the lithium carbonate industry [2]. Coupled with this, the disorderly competition has led to the release of production capacity at a rate far greater than the normal demand of the industry, which in turn has triggered huge fluctuations in the price of lithium carbonate in China [3]. Such dramatic price turbulence is detrimental to the development of the lithium carbonate industry. This further reflects the importance of accurately capturing the trends in the price of lithium carbonate in a timely manner. Therefore, the ability to accurately predict changes in China's lithium carbonate prices will help provide battery manufacturers and end-users with more precise market predictions, thereby optimizing procurement and production plans and reducing supply chain risks.

Research on lithium carbonate price prediction models can be mainly categorized into three types. The first is supply and demand prediction models [4–6]. Although such model are able to predict lithium carbonate prices from the supply and demand dimensions, their accuracy is limited because they do not account for nonlinear market factors or external disturbances. The second is statistical and time series analysis models [7,8]. These models are capable of predicting the trends in lithium carbonate price fluctuations based on historical price data. However, they struggle to handle the instability of lithium carbonate prices when facing sudden events or changes in market structure. The third are machine learning methods [9,10]. Among these, LSTM (Long Short-Term Memory) networks [11,12] have been widely used in time-series forecasting due to their ability to capture long-term dependencies in sequential data. However, applying LSTM alone to predict lithium carbonate prices has certain limitations, such as difficulties handling high volatility and external disturbances. To address these issues, researchers have explored optimized LSTM-based models. These approaches integrate techniques such as VMD (Variational Mode Decomposition) [13] for feature extraction and SSA (Sparrow Search Algorithm) [14] for parameter optimization. While VMD enhances the model's ability to process nonlinear and non-stationary data, SSA helps in fine-tuning hyper-parameters to improve predictive accuracy. In order to further improve the prediction accuracy, scholars have constructed VMD–SSA–LSTM hybrid models [15–17]. A representative study is [16], in which the authors constructed a VMD–SSA–LSTM combined model and accurately predicted the price of silicon in China based on the non-stationarity, nonlinearity, and expanded complexity characteristics of silicon prices across the chain in the photovoltaic industry. However, the existing research primarily focuses on specific commodity markets, with explorations of lithium carbonate pricing mechanisms in the context of the battery industry supply chain being limited. Therefore, inspired by this research, this study aims to accurately predict the price of lithium carbonate in China by constructing a VMD–SSA–LSTM combined model, and taking the highly nonlinear and unstable characteristics of the lithium carbonate price in China's battery industry chain into full consideration.

In summary, according to the characteristics of China's lithium carbonate price volatility, sensitivity to macroeconomic factors, and susceptibility to speculative trading activities, this paper comprehensively considers key influencing factors such as the prices of materials related to the lithium battery industry and the macro-environment. At the same time, on the basis of the VMD algorithm, SSA is introduced to the LSTM model to develop a VMD–SSA–LSTM combination model with a higher prediction accuracy for Chinese lithium carbonate prices. The main innovations of this paper are as follows: (1) The prediction model presented in this

paper takes into account a combination of the relevant pricing of materials in the industry chain, macroeconomic indicators, and the time series characteristics of lithium carbonate prices. By introducing multidimensional influencing factors, the paper reveals the underlying logic behind lithium carbonate price fluctuations, and thus achieves a more accurate prediction analysis. (2) In this paper, a combined model of a Long Short-Term Memory (LSTM) network optimized using Variational Modal Decomposition (VMD) and the Sparrow Search Algorithm (SSA) is applied to the prediction of lithium carbonate prices. VMD effectively decomposes nonlinear and non-stationary data, while the SSA optimizes the hyperparameters, enhancing the overall predictive performance of the model.

The structure of this paper is as follows: In the second part, three distinct models are assembled. Subsequently, the third part focuses on the construction of the combined VMD–SSA–LSTM model. The fourth part centers around predicting the price of lithium carbonate using the VMD–SSA–LSTM combination model. Finally, the fifth part summarizes the main conclusions drawn from this article.

2. Single Model Construction

2.1. VMD Algorithm

Variational mode decomposition (VMD) is a non-recursive algorithm for signal processing, which can decompose complex signals into simple signals with different center frequencies. These signals have a limited bandwidth and sparse characteristics in the frequency band. By optimizing the limited bandwidth constraint, the VMD method ensures that the decomposed modal components and residuals can accurately reconstruct the original signal [18–20]. However, VMD lacks an automated mechanism for determining the optimal K, requiring empirical selection or additional optimization methods. To achieve this, the VMD method consists of the following specific steps:

(1) Analyze the analytical signal corresponding to each mode through the Hilbert transform [18], and subsequently assemble the spectrum to obtain the analytical signal at a certain time t:

$$s_k(t) = \left(\delta(t) + \frac{j}{\pi t}\right) u_k(t) \tag{1}$$

where $u_k(t)$ represents the k-th intrinsic mode function (IMF) component; $\delta(t)$ is a pulse signal; j is the imaginary unit; t is time.

(2) To move the spectrum of each mode to the appropriate baseband, correct the estimated center frequency of the analytical signal for each mode.

(3) Using the squared norm, estimate the bandwidth of the demodulated signal while keeping the total bandwidths to a minimum. The following is the constraint:

$$\begin{cases} \min\limits_{\{u_k\}\{\omega_k\}} \left\{ \sum\limits_K \left\| \mu_t \left[\left(\delta(t) + \frac{j}{\pi t}\right) u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\} \\ s.t. \sum\limits_K u_k(t) = y(t) \end{cases} \tag{2}$$

in which μ_t is a time function; ω_k is the center frequency of the k-th mode component; $y(t)$ is the original signal. In order to solve the above constrained variational problem, the penalty coefficient μ and Lagrange multiplier λ are introduced. $\langle , \rangle$ denotes the dot product. The constraint condition is transformed into an equality reduction:

$$L(\{u_k(t)\}, \{\omega_k\}, \lambda) = \mu \sum_K \left\| \mu_t \left[\left(\delta(t) + \frac{j}{\pi t}\right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 + \left\| y(t) - \sum_K u_k(t) \right\|_2^2 + \left\langle \lambda(t), y(t) - \sum_K u_k(t) \right\rangle \tag{3}$$

2.2. Sparrow Search Algorithm

The Sparrow Search Algorithm (SSA) is a powerful metaheuristic algorithm for optimizing model hyperparameters, improving forecasting accuracy. However, it does not possess an intrinsic forecasting capability and must be combined with a predictive model to yield meaningful results [21,22]. Three roles are involved in this process: discoverers, joiners, and alarmists. Discoverers are responsible for searching within the population, which requires great fitness and a vast search area. Joiners are comparatively less fit and primarily follow discoverers. When alarmists identify predators in a population, they send out warning signals, alerting discoverers to swiftly relocate the population to a safe region. Consequently, the sparrow population's matrix representation is as follows:

$$X = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^d \\ x_2^1 & x_2^2 & \cdots & x_2^d \\ \vdots & \vdots & & \vdots \\ x_n^1 & x_n^2 & \cdots & x_n^d \end{bmatrix} \tag{4}$$

in which n is the population's number of sparrows; d indicates the dimensionality of the variables in the optimization problem.

The SSA population's fitness function can be defined as the following:

$$F_x = \begin{bmatrix} f\left(\begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^d \end{bmatrix}\right) \\ f\left(\begin{bmatrix} x_2^1 & x_2^2 & \cdots & x_2^d \end{bmatrix}\right) \\ \vdots \\ f\left(\begin{bmatrix} x_n^1 & x_n^2 & \cdots & x_n^d \end{bmatrix}\right) \end{bmatrix} \tag{5}$$

in which f represents each sparrow's fitness level.

For discoverers, the precise position update is provided by the following:

$$X_{z,j}^{l+1} = \begin{cases} X_{z,j}^l \times \exp\left(\frac{-z}{\alpha \times i_{\max}}\right), R_2 < T_s \\ X_{z,j}^l + Q \times L, R_2 \geq T_s \end{cases} \tag{6}$$

in which $X_{z,j}^l$ indicates the position of the z-th sparrow in the j-th dimension at the l-th iteration; $i_{\max}$ represents the final iteration number; $\alpha \in [0,1]$; Q is a random value; L is the unit row vector. $R_2 \in [0,1]$ and $T_s \in [0.5,1]$ stand for the safety value and the alarm value, respectively. When $R_2 < T_s$, this signifies that no predators are present in the forage surroundings, enabling the discoverers to conduct extensive searches. When $R_2 \geq T_s$, this indicates that there is dangerous information, and sparrows should be transferred to a safe area to search. Therefore, the following represents the joiners' foraging process after the action of the discoverers:

$$X_{z,j}^{l+1} = \begin{cases} Q \times \exp\left(\frac{X_{worst}^l - X_{z,j}^l}{z^2}\right), i > \frac{n}{2} \\ X_q^{l+1} + \left|X_{z,j}^l - X_q^{l+1}\right| \times D^T \left(DD^T\right)^{-1} \times L, else \end{cases} \tag{7}$$

in which X_q is the greatest position for the discoverers; X_{worst} is the poorest position of the population at that point; D is a s a matrix containing randomly allocated elements of 1 or -1. When $i > \frac{n}{2}$, this indicates that the i-th participant must fly across distinct areas to scavenge for extra power.

When alerted to the danger, the position of the alerter is upgraded:

$$X_{z,j}^{l+1} = \begin{cases} X_{z,j}^{l} + K \times \left(\frac{\left| X_{z,j}^{l} - X_{worst}^{l} \right|}{(g_z - g_\omega) + \varepsilon} \right), g_z = g_b \\ X_{best}^{l} + \beta \times \left| X_{z,j}^{l} - X_{best}^{l} \right|, g_z \neq g_b \end{cases} \tag{8}$$

in which g_z represents the sparrow's current level of fitness; g_b and g_ω are the global optimal fitness and the worst fitness; X_{best} is the global optimal position at the current time; β is the control step parameter and obeys the standard normal distribution; K is the flight trajectory of the sparrow, which is a random number of $[-1, 1]$; and ε is a constant used for avoiding division by zero. When $g_z \neq g_b$, this shows that sparrows are vulnerable to danger. When $g_z = g_b$, this shows that sparrows are aware of the danger and tend to be close to other individuals at this time, reducing the possibility of them being attacked.

At each iteration, the fitness values of all individuals are calculated and updated according to the size of the fitness values to obtain the position of the current global optimal fitness until the end of the cycle. At the end of the cycle, the best fitness value is taken as the global optimal solution.

2.3. Long Short-Term Memory Network

Long Short-Term Memory (LSTM), a prominent architecture for neural systems, is designed to capture long-term dependencies in time series data. It achieves this by integrating sophisticated gating mechanisms, namely the forget gate, input gate, and output gate, which collaboratively regulate the flow of information. Through the utilization of these gates, LSTM has a remarkable ability to retain long-term memory within time series data. Moreover, it effectively mitigates the critical issues of gradient disappearance and gradient explosion that commonly occur during the back-propagation process, thereby enhancing the stability and performance of the model in handling sequential data [23].

While LSTM demonstrates strong capabilities in time-series forecasting, its performance can be influenced by certain factors. For instance, due to its reliance on complex gating mechanisms, LSTM models may require considerable computational resources, especially when handling large-scale datasets. Furthermore, the model's predictive accuracy is sensitive to hyperparameter tuning, necessitating careful optimization to achieve the best results.

LSTM also describes a cell state for storing information. Figure 1 shows the LSTM's hidden layer cell structure. Equations (9)–(13) provide an illustration of its forward determining technique:

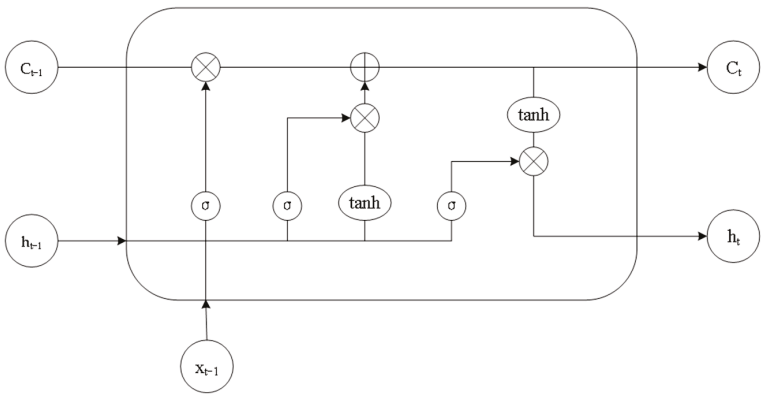


Figure 1. LSTM unit cell structure.

(1) The forget gate functions to determine the information that it is necessary to forget [24,25].

$$f_t = \sigma\Big(W_f \cdot [h_{t-1}, x_t] + b_f\Big) \tag{9}$$

where f_t is the forget gate, h_{t-1} is the previous output, x_t is the input of the current cell, W_f is the weight matrix of the forget gate, σ is sigmoid function, and b_f is the deviation term.

(2) The input gate is in charge of choosing which fresh data will be kept in the cell state.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{10}$$

$$C_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{11}$$

where i_t is the input gate, σ is sigmoid function, C_t is cell state, tanh is hyperbolic tangent activation function, W_i and W_c are the weight matrix, and b_i and b_c are the deviation term.

(3) The output gate regulates the amount that data is released by the LSTM unit:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{12}$$

$$h_t = o_t * \tanh(C_t) \tag{13}$$

where o_t is the output gate, σ is sigmoid function, h_t is the output value of the current unit, W_o is the weight matrix, b_o is the deviation term, and $*$ is the point multiplication of a vector.

Each of the above techniques has its own advantages and limitations when applied individually. To address these limitations, we propose an integrated VMD–SSA–LSTM model, which combines the decomposition capability of VMD, hyperparameter optimization of SSA, and deep learning-based forecasting strength of LSTM. This hybrid approach enhances feature extraction, model robustness, and predictive accuracy, making it suited for handling the high volatility in lithium carbonate prices.

3. VMD–SSA–LSTM Combination Model

3.1. Data Sources and Indicators

This study used applicable information from the Chinese lithium battery industrial chain, which ran from 2 January 2020 to 29 November 2024, as the research sample to guarantee the accuracy and timeliness of the data. The price of lithium carbonate, an important raw material in the lithium battery industry, is impacted by a number of factors. To precisely predict the price, this study chose the prices of lithium hydroxide, lithium iron phosphate, lithium manganese oxide, lithium hexafluorophosphate, the CSI 300 Index, and the Shanghai Composite Index as the factors influencing lithium carbonate prices, as presented in Table 1. After excluding samples that had missing values for the primary variables, the final datasets contained 1182 samples.

Table 1. Data source.

Indicator	Variables	Data Sources
Related Product Prices	Lithium Hydroxide Price	Wind
	Lithium Iron Phosphate Price	Wind
	Lithium Manganese Oxide Price	Wind
	Lithium Hexafluorophosphate Price	Wind
Macroeconomic Indicators	CSI 300 Index	Shanghai Stock Exchange
	Shanghai Composite Index	Shanghai Stock Exchange

In this paper, the lithium battery industry chain, related material prices, and macro-environmental indicators are included in the range of factors to be considered, providing strong support for the comprehensive consideration of the changes in the price of lithium carbonate. These factors interact with each other and jointly determine the market price of lithium carbonate, as explained below:

(1) Lithium hydroxide and lithium carbonate have certain correlations in terms of preparation processes and uses. The price connection mechanism between the two indicates that fluctuations in the price of lithium hydroxide may have an impact on the supply and demand relationship in the lithium carbonate market. In addition, as a part of the lithium battery materials, the price trend of lithium hydroxide reflects the overall cost changes in the entire lithium battery industry chain.

(2) Lithium iron phosphate is a key cathode material for lithium batteries, and its demand has a direct impact on lithium carbonate prices. With the rapid growth of the new energy vehicle sector, demand for lithium iron phosphate has skyrocketed, driving up the price of lithium carbonate. However, fluctuations in the price of lithium iron phosphate may have an indirect impact on the lithium carbonate market. For example, if the price of lithium iron phosphate falls, this may lead to a decrease in demand for lithium carbonate, which affects its price.

(3) As another cathode material for lithium batteries, changes in demand for lithium manganate will have an impact on market demand for lithium carbonate. The supply and demand situation, as well as the price trend of lithium manganate, can be used as an important reference indicator of the lithium carbonate market, particularly in the context of the diversified development of lithium battery materials. Due to its low cost, lithium manganate is primarily used in low-end markets such as power tools and electric bicycles; if demand in these areas surges, the price of lithium carbonate will be lowered.

(4) Lithium hexafluorophosphate is an important component of lithium battery electrolytes, and supply and demand fluctuations have a direct impact on lithium battery production costs. The rising price of lithium hexafluorophosphate will put more pricing pressure on lithium battery producers, perhaps increasing demand for lithium carbonate. In contrast, a drop in the price of lithium hexafluorophosphate may relieve cost pressure on lithium battery makers, hence inhibiting the lithium carbonate market.

(5) The stock market's performance regularly reflects investors' forecasts for future economy and industry. The movements of the CSI 300 Index and the Shanghai Composite Index can be utilized to assess macroeconomic conditions and market confidence, which can have an indirect impact on lithium carbonate prices. When the macro-environment is performing well, investors may be more willing to invest in lithium battery-related industries, thus pushing up the demand for and price of lithium carbonate.

3.2. The Process of Combination Model

To systematically predict lithium carbonate prices, this study constructed a VMD–SSA–LSTM hybrid model, which integrates data decomposition, feature extraction, and parameter optimization to enhance prediction accuracy. The process of the combined model proposed in this paper is shown in Figure 2 and the framework consists of five key steps:

(1) The model considers seven influencing factors, including lithium hydroxide price, lithium iron phosphate price, lithium manganese oxide price, lithium hexafluorophosphate price, CSI 300 Index, Shanghai Composite Index, and historical lithium carbonate prices. The output variable is the predicted lithium carbonate price. Before modeling, the dataset is divided into training (70%) and validation (30%) sets, and all data are normalized to ensure consistency.

(2) Since lithium carbonate prices exhibit high volatility and non-stationarity, VMD (Variational Mode Decomposition) is applied only to lithium carbonate price data to decompose them into multiple Intrinsic Mode Functions (IMFs), which helps to extract key frequency components and reduce noise.

(3) The input value of LSTM stores the data of the first m lithium carbonate prices and the first $(m + 1)$ influencing factors, while the output value outputs the predicted value of the lithium carbonate price at the $(m - 1)$-th moment.

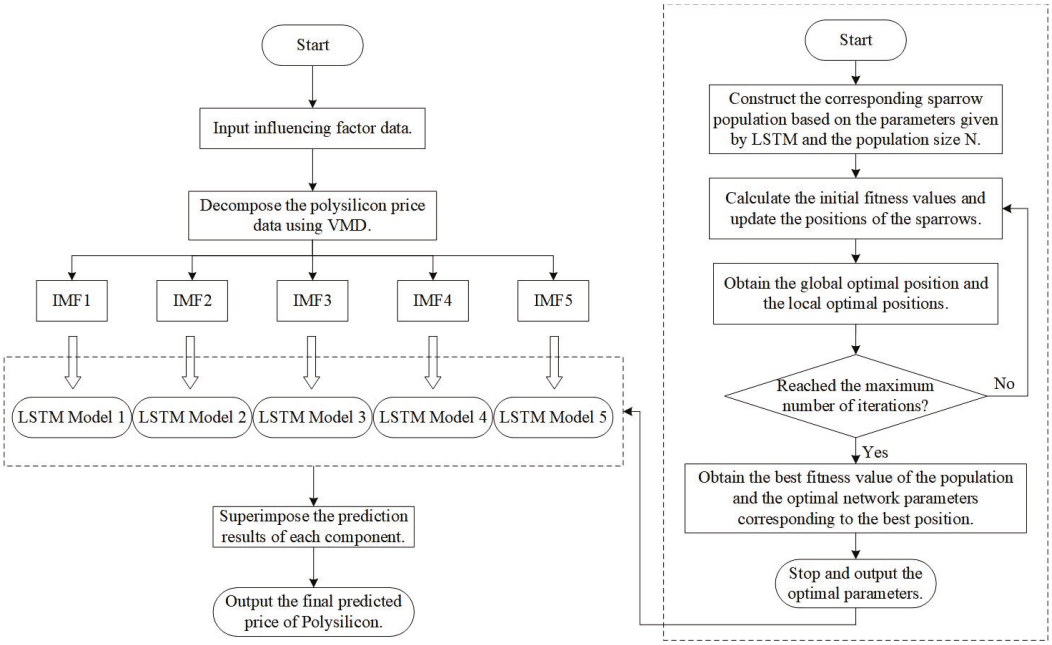


Figure 2. The operation flow chart of VMD–SSA–LSTM.

(4) Optimization parameters, such as the number of sparrows in the population N and the boundary constraints for the circumstances to be improved, are set in the LSTM. Using the information given from the LSTM, the SSA constructs a matching sparrow population and calculates its ideal fitness value and position.

(5) The parameter data from the SSA are used to determine the three optimal parameters for the LSTM model. The LSTM prediction model is reconstructed according to these optimal parameters. Next, the projected lithium carbonate prices are renormalized to generate the prediction curve.

3.3. Evaluation Index of Combination Model

This study evaluates prediction model accuracy using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2) [26].

Mean Absolute Error (MAE) measures the average magnitude of errors in predictions, providing a clear indication of the model's predictive accuracy. Smaller MAE values indicate fewer deviations between the predicted and actual values, thus reflecting a more accurate model. Root Mean Squared Error (RMSE) penalizes larger errors more heavily by squaring the residuals before averaging. It is sensitive to outliers, making it useful for highlighting large prediction errors. Lower RMSE values indicate better overall model performance. Mean Absolute Percentage Error (MAPE) expresses the prediction error as a percentage, offering an easily interpretable metric, which is particularly useful when

comparing models across different scales. Smaller MAPE values indicate higher relative accuracy in predictions. R-squared (R^2) indicates the proportion of variance in the target variable explained by the model. An R^2 value closer to 1 denotes a higher degree of fit, suggesting the model explains most of the variability in the data.

While an R^2 value nearer 1 denotes a higher degree of fit, smaller values of MAE, RMSE, and MAPE imply more accurate model prediction.

$$MAE = \frac{\sum_{t=1}^{N} |f_t - y_t|}{N} \tag{14}$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^{N} (f_t - y_t)^2}{N}} \tag{15}$$

$$MAPE = \frac{\sum_{t=1}^{N} \left| \frac{f_t - y_t}{y_t} \right|}{N} \times 100\% \tag{16}$$

$$R^2 = 1 - \frac{\sum_{t=1}^{N} (y_t - f_t)^2}{\sum_{t=1}^{N} (y_t - \overline{y})^2} \tag{17}$$

For these equations, y_t is the genuine value of lithium carbonate prices; f_t indicates the price predictions for lithium carbonate materials; $\overline{y}$ stands for the average value of lithium carbonate prices; N shows the quantity of samples in the test datasets.

4. Analysis of Lithium Carbonate Price Prediction in China

4.1. Descriptive Statistical Analysis

Table 2 presents the descriptive statistics of the relevant variables. The mean and standard deviation of lithium carbonate and lithium hydroxide are large, indicating high price volatility, and the minimum and maximum prices of the two are very close to each other, so there may be some market linkage. The skewness of most of the variables is close to positive (>0), indicating that these data have a slight rightward bias, i.e., longer tails. The skewness of the SSE Composite Index is −0.06, which is close to a symmetric distribution. The kurtosis of all variables is close to 3, indicating that they are close to the thickness of the tails of the normal distribution, with lithium hexafluorophosphate prices and lithium manganate prices having relatively high kurtosis.

Table 2. Descriptive statistics.

VarName	Obs	Mean	SD	Min	Max	Skewness	Kurtosis
Lithium Hydroxide Price	1182	19.90	16.97	4.80	56.02	0.92	2.27
Lithium Iron Phosphate Price	1182	7.82	4.84	3.25	17.70	0.86	2.15
Lithium Manganese Oxide Price	1182	3.63	2.19	1.18	8.58	0.99	2.93
Lithium Hexafluorophosphate Price	1182	19.97	15.77	5.42	59.00	1.15	3.18
CSI 300 Index	1182	4215.84	605.93	3159.25	5807.72	0.38	2.07
Shanghai Composite Index	1182	3224.03	236.23	2660.17	3715.37	−0.06	2.14
Lithium Carbonate Price	1182	20.23	17.10	4.00	56.76	0.88	2.24

Subsequently, the Phillips–Perron test was employed to examine the stability of lithium carbonate prices. The calculated p-value was 0.8580, substantially greater than the signif-

icance level of 0.05. Thus, the null hypothesis could not be rejected. The results suggest potential non-stationarity in the lithium carbonate price data, indicating that statistical properties like the mean and variance may vary over time. This has crucial implications for the further analysis and modeling of lithium carbonate price trends.

4.2. Variational Mode Decomposition

The VMD approach was used to break down the data because of the volatility and unpredictability of lithium carbonate price data. The selection of decomposition modes has a considerable impact on the VMD outcomes. In light of this, the ideal number of decomposition modes was obtained using the central frequency approach, yielding $K = 5$ [27]. The choice of $K = 5$ in VMD was based on the volatility and non-stationary nature of lithium carbonate price data. To determine the optimal number of decomposition modes, the central frequency approach was used. This method involves gradually increasing K and analyzing whether the extracted modes have stable and meaningful frequency distributions. If K is too small, different frequency components may mix, leading to incomplete decomposition. Conversely, if K is too large, redundant modes may appear, introducing noise rather than useful information. Through experimental validation, $K = 5$ was found to achieve a balance between preserving key information and avoiding excessive decomposition. This selection ensures that VMD effectively captures both short-term fluctuations and long-term trends, ultimately improving the predictive performance of the model. Other parameters have the following numerical settings: penalty factor $\alpha = 2500$; noise tolerance $tau = 0$; $DC = 0$; and convergence precision $tol = 10^{-7}$.

In this research, a total of 1182 pieces of price data, with empty data removed, were decomposed into five Intrinsic Mode Function (IMF) components. The IMF components resulting from the decomposition of the price data from the 1st to the 1000th trading days are depicted in Figure 3. Among them, the first row represents the original sequence signal and rows 2–6 represent the IMF components (IMF1 to IMF5) decomposed by VMD, arranged from low frequency to high frequency. Each IMF component retains the characteristics of the original lithium carbonate price signal while avoiding modal aliasing effects.

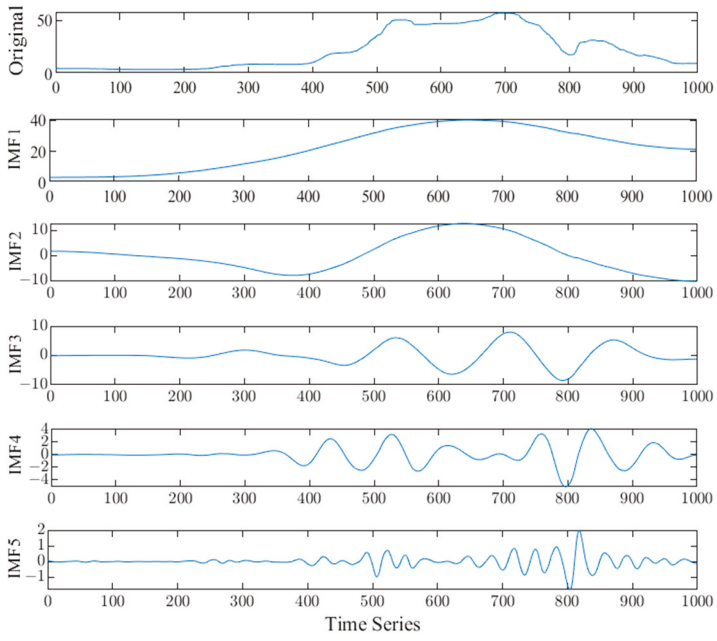


Figure 3. VMD of lithium carbonate price.

The sample datasets were processed to improve convergence speed and prediction accuracy, minimize the impact of anomalous data on predictions, and prevent notable numerical disparities among features. This also expedites the LSTM prediction model's training process. Thus, normalization was used to scale the data to a range of (0, 1) for both the lithium carbonate price datasets and the impacting factors. The following equation provides the calculation used for the normalization approach.

$$y_t = \frac{x_t - \min(x_t)}{\max(x_t) - \min(x_t)} \tag{18}$$

In the equation, y_t is the normalized result of a given dataset, x_t indicates the given initial data, and $\min(x)$ and $\max(x)$ stand for the minimum and maximum values in the sample, respectively.

4.3. Parameter Optimization of LSTM

To enhance the model's expressive capability, activation functions were used to introduce nonlinear factors. The combination model consists of an input layer, LSTM layer, ReLU activation layer, and a formula output layer. To reduce the influence of subjective experience on the combination model, the SSA was employed to optimize three parameters: the number of neurons in the hidden layer, the number of iterations, and the learning rate. The LSTM network step size was set to 5, and upper and lower boundaries were established.

The hyperparameters of the LSTM model were carefully tuned to balance underfitting and overfitting. The number of hidden layer neurons was set to between 50 and 300, as values below 50 led to underfitting, while values exceeding 300 caused overfitting. The iteration count ranged from 50 to 300, considering that excessive iterations increased weight updates, leading to overfitting, whereas too few iterations reduced prediction accuracy. The learning rate was selected to range from 0.001 to 0.01, as a higher rate risked gradient explosion, while a lower rate slowed convergence.

Activation functions were utilized to inject nonlinear elements into the model in order to improve its expressiveness. A suitable optimization algorithm must be used in order to increase the prediction accuracy and create a more accurate prediction model. The three network parameters were optimized through the application of the SSA and the PSO algorithm. The comparison of the fitness and the number of iterations of the two algorithms is shown in Figure 4. As depicted in Figure 4a, the fitness value of the PSO–LSTM model was 0.026, whereas that of the SSA–LSTM model was 0.010. Compared with the PSO optimization algorithm, the SSA optimization showed a stronger optimization ability in the iteration, which shows that the SSA can play a better role in parameter optimization. Therefore, this paper adopted the SSA optimization, which is more suitable for LSTM parameter optimization. The process by which SSA optimizes the LSTM is shown in Figure 4b, and the optimal parameters of the LSTM optimized by the SSA are 103 hidden-layer neurons, 300 iterations, and a learning rate of 0.001.

Subsequently, the LSTM network was be fitted with the optimal hyperparameters, and each component underwent modeling. To verify the reliability and prediction accuracy of the VMD–SSA–LSTM combination model constructed in this paper for predicting lithium carbonate prices, the dataset obtained after performing the Variational Mode Decomposition (VMD) on the lithium carbonate price data was used as the sample dataset. The training sample set was the first 70% of this dataset, which spans roughly 827 trade days between 2 January 2020, and 1 June 2023. The remaining 30% served as the testing sample set, covering roughly 355 trading days between 2 June 2023 to 29 November 2024. This

experiment examined the prediction of the sample dataset using the LSTM, VMD–LSTM, and VMD–SSA–LSTM models. Figure 5 shows a comparison of the variations between the actual and anticipated values for each of the three models. It is evident that the single LSTM network fluctuates between [−2.5, 0.6] for the complex lithium carbonate prices when forecasting the 355 lithium carbonate price data in the testing set. The VMD–SSA–LSTM model, on the other hand, shows prediction errors that vary between [−0.5, 2], indicating reduced prediction errors and better stability.

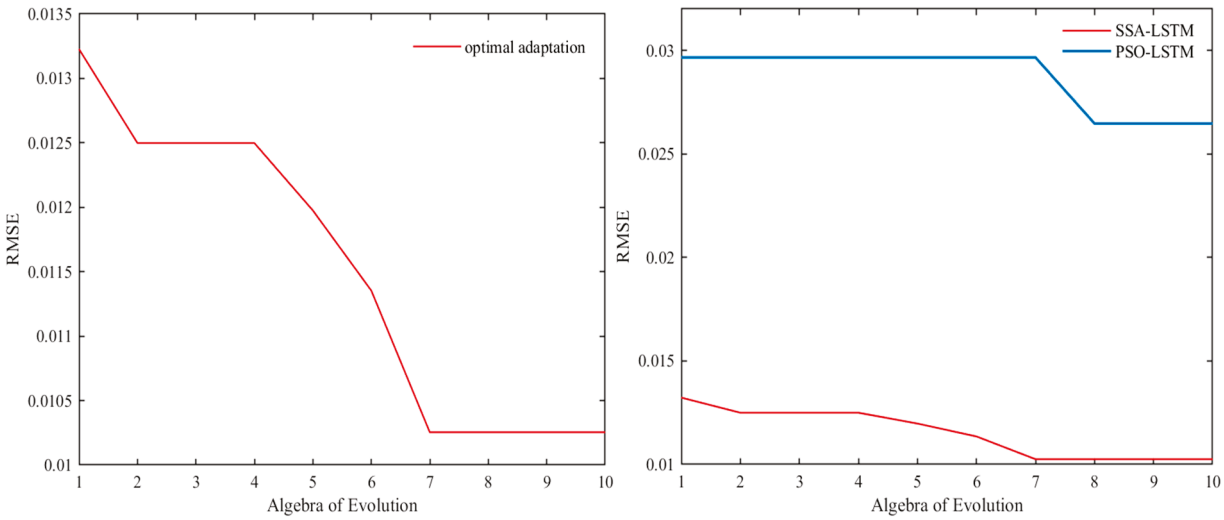


Figure 4. Optimization iterative process of SSA and PSO. (**a**) The convergence curves of LSTM optimized by SSA and PSO; (**b**) the convergence curve of the SSA–LSTM.

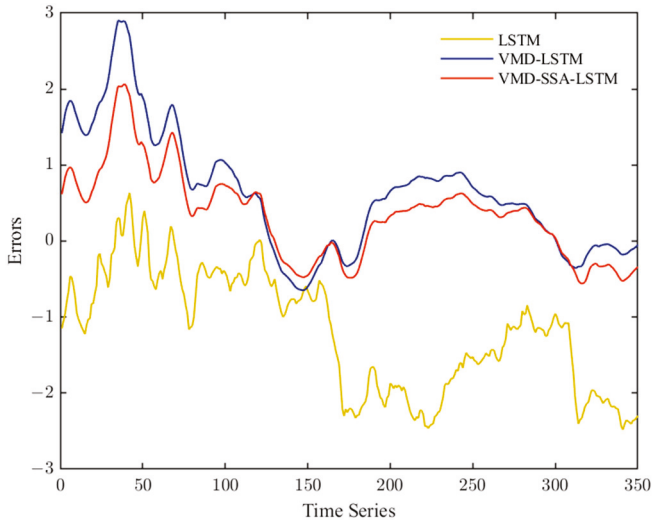


Figure 5. Comparison of errors regarding the lithium carbonate prices predicted by three models.

Figure 6 depicts the results of lithium carbonate material price predictions, where Figure 6a shows the results of the models in the training set, and the results of the testing set are shown in Figure 6b. It can be observed that the overall price of lithium carbonate fluctuates greatly during the trading days from 2020 to 2024, and the price shows an obvious downward trend during 2023, from 509,700 CNY/tonne at the beginning of the year to 96,900 CNY/tonne at the end of the year. The price of lithium carbonate fluctuated slightly in 2024 and finally fell to 78,200 CNY/tonne. The single LSTM network fits the

target curve the worst overall, especially when the price of lithium carbonate fluctuated greatly. However, the model in this paper fitted the actual prices more accurately.

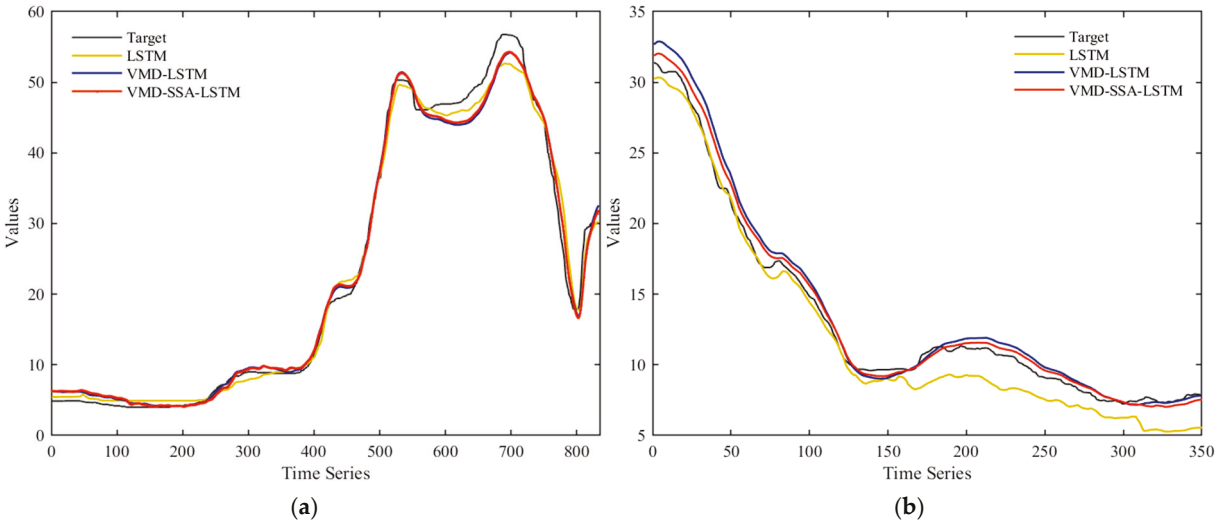


Figure 6. Comparison of the lithium carbonate prices predicted by three models. (**a**) Training set results; (**b**) testing set results.

In order to further verify the effectiveness and stability of the model in this paper, four evaluation indicators were introduced to assess the performance of the model. Table 3 shows the performance index of the models.

Table 3. Prediction performance indicators of four models for the training set and testing set.

	Training Set				Testing Set			
Model	**MAE**	**RMSE**	**MAPE%**	$\mathbf{R^2}$	**MAE**	**RMSE**	**MAPE%**	$\mathbf{R^2}$
LSTM	1.3467	1.8501	4.6514	0.9906	1.1976	1.4556	5.7504	0.9532
VMD–LSTM	1.3219	1.8305	4.9193	0.9908	0.6014	0.7657	4.5882	0.9870
VMD–SSA–LSTM	1.1527	1.5252	4.1266	0.9936	0.5614	0.7359	4.3568	0.9880

The VMD–SSA–LSTM combination model exhibited the best performance in the training set, as indicated by Table 3. In particular, the performance measures demonstrate that, in comparison to the single LSTM model, this approach increased the R^2 by 0.0030 and decreased the MAE by 0.1940, the RMSE by 0.3249, and the MAPE by 0.5248 percentage points. This suggests that, in comparison to predictions produced without data decomposition, the model's prediction accuracy is higher with VMD. The VMD technique helps to decompose the original complex time series data into several relatively simple and more regular sub-components, which might help the model to better understand and model the data structure.

In addition, the MAE decreased by 0.1692, the RMSE by 0.3053, and the MAPE by 0.7927 percentage points, and the R^2 increased by 0.0028, in comparison to the VMD–LSTM prediction technique. This further illustrates that the VMD–SSA–LSTM combination model shows great promise in the field of lithium carbonate price prediction, leveraging the strengths of data decomposition, parameter optimization, and neural network modeling. It effectively reduces prediction errors and enhances the model's ability to capture the underlying patterns in the data, indicating its potential for practical applications in the complex and dynamic market environment of lithium carbonate pricing.

4.4. Further Analysis

According to the results obtained from the above research, and combined with the current reality of the development of China's lithium carbonate industry, this paper further analyzes how to predict the price of lithium carbonate in 2025:

(1) Use the supply of lithium resources to further predict the price of lithium carbonate in 2025. The above analysis verified the advantages of this paper's model in predicting the price of lithium carbonate. This paper predicts the trends in lithium carbonate price changes based on the above model and, combined with the current supply of lithium resources in the world, the situation can be further analyzed and the following information can be obtained: First, the world's lithium resource reserves and supply is sufficient. In 2024, the world's total supply of lithium resources is expected to be 1.33 million tonnes, the world's total demand is expected to be 1.15 million tonnes; the world's supply of lithium resources exceeds the demand for 180,000 tonnes. This will inevitably lead to a further decline in the price of lithium carbonate raw materials in 2025, which, in turn, will trigger a sustained decline in the price of lithium carbonate. Secondly, China's lithium resources supply chain is complete. China is currently the world's largest new energy vehicle market, and also has built a complete lithium resources supply chain system. This will effectively guarantee the effective supply of lithium resources and protect it against the impact of the external uncertainty environment. This further supports the idea that the price of lithium carbonate will generally be lower in 2025. These analyses also further illustrate the robustness of the forecast model presented above.

(2) The price of lithium carbonate in 2025 can be further predicted based on cognitive biases under the effect of information asymmetry. In the process of constructing the above prediction model, it was found that the fluctuations in lithium carbonate prices are not only the result of the internal and external environment, but also a comprehensive reflection of the cognitive biases of different market participants. This indicates that the accurate prediction of lithium carbonate prices needs to be further analyzed in combination with the cognitive biases of different market participants. A specific analysis can be made of the information asymmetry caused by different market participants' cognitive biases. Due to the development trend of the lithium carbonate industry and the fluctuations in the trading price, there is a large amount of information asymmetry. On the one hand, the industry's market participants that make investment decisions, being misled by the past benefits of the rich historical market information that is available, may choose to continue to increase their investment, leading to the release of excessive production capacity. On the other hand, when the lithium carbonate industry suffers short-term losses, market participants' decisions will also be based on their inherent empirical judgment; they may insist that the losses are short-lived adjustments, and that there will still be better investment returns later on. This would lead to a high lithium carbonate production capacity, resulting in the entire lithium carbonate industry being caught in a capacity trap from which it cannot extricate itself. This further indicates that the price of lithium carbonate in 2025 will not only show an overall low trend, but also may face more frequent fluctuations in the adjustment period.

(3) The theory of herd behavior can be used to further predict the price of lithium carbonate in 2025. At present, although the lithium carbonate industry has been in a booming stage of development, investors outside the industry are still affected by herd behavior and blindly enter the lithium carbonate industry. Specific analysis shows that first, investors outside the lithium carbonate industry look at the development

of new energy industry and believe it will bring huge dividends, choosing a swarm into this industry. This behavior is typical herd behavior. This suggests that investors outside the lithium carbonate industry not only lack the ability to think independently but also rely excessively on the decisions of others. Secondly, under the influence of information asymmetry, these investors ignore in-depth studies of the market and investment targets and, furthermore, ignore their own core business and risk control measures. This leads to a lack of scientific and rational investment decisions by investors outside the lithium carbonate industry, who instead blindly enter the lithium carbonate industry. As a result, due to the effects of herd behavior, the price of lithium carbonate in 2025 will be excessively volatile, further deviating from its true value and increasing the risk and uncertainty of the market. More seriously, under the continuous aggregation of herd behavior, there will be over-investment and investment mistakes in lithium carbonate enterprises, which will bring severe investment risks and financial difficulties.

5. Conclusions

To improve the stability and accuracy of lithium carbonate price prediction, this study examined the features of Chinese lithium carbonate prices. Through the incorporation of the Variational Mode Decomposition (VMD) and the Sparrow Search Algorithm (SSA) method into the Long Short-Term Memory (LSTM) model, we constructed the VMD–SSA–LSTM combination model, which enables a higher accuracy to be obtained when forecasting lithium carbonate prices in China. The primary findings from the research presented in this paper are outlined below:

(1) The VMD–SSA–LSTM combination model not only comprehensively considers various factors affecting lithium carbonate prices but also fully accounts for their intrinsic association with time series data. (2) By using the VMD algorithm to decompose the time series and effectively extract components at different frequencies, combined with the SSA for global parameter optimization of the LSTM model, the VMD–SSA–LSTM combination model effectively integrates the frequency and temporal characteristics of time series data, enabling a holistic consideration of multidimensional data features. (3) On the lithium carbonate pricing datasets, the VMD–SSA–LSTM model performs better in terms of prediction than both the VMD–LSTM model and the traditional LSTM model. The model significantly enhances the convergence accuracy. This enables the model to adapt more quickly and effectively to new data and market dynamics, with a stronger generalization ability and adaptability.

This study provides valuable insights for price decision-making in the lithium carbonate industry. While the research primarily concentrates on the data properties of the Chinese lithium carbonate market, significant heterogeneity exists among lithium carbonate markets in different countries and regions due to differences in policy environments, resource endowments, and industrial structures. Moreover, the global lithium carbonate market has formed close interconnections through trade and industrial chain links. In this context, price fluctuations in any region not only reflect local market characteristics but may also impact other regions through supply chains or demand-side channels. The next phase of this research will focus on the global lithium carbonate market. Through the extensive collection of data from major markets and an analysis of the price transmission mechanisms and the linkages between different regional markets, we can construct a more globally oriented prediction model. This will provide industry participants with decision-making references that are more aligned with actual market dynamics, thereby promoting the sustainable development of the lithium carbonate market. At the same

time, in-depth research will be conducted on the types of cognitive biases of Chinese entrepreneurs and their impact on investment decisions regarding lithium carbonate price forecasts. This will provide a decision-making reference for preventing excessive price fluctuations in the lithium carbonate market and ensuring the normal operation of lithium carbonate enterprises.

Author Contributions: H.L.: conceptualization and methodology. W.W. and L.J.: writing—original draft preparation and analysis. L.W.: calculation and supervision. W.W., H.L., L.J., and L.W.: contributed equally to this work. They are co-first authors. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by Key Program of International Cooperation and Exchange of National Natural Science Foundation of China (W2411065), National Natural Science Foundation of China (No. 71971111) and Jiangsu Province's Youth Science and Technology Talent Support Project (JSTJ-2024-438).

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: Thank you to the reviewers for providing insightful feedback, which greatly benefited us.

Conflicts of Interest: The authors declare that they have no competing interests.

References

1. Peng, Z.; Yin, T.; Zhu, X.; Lu, X.; Li, X. Hybrid price prediction method combining TCN-BiGRU and attention mechanism for battery-grade lithium carbonate. *Kybernetes* **2024**. [CrossRef]
2. Pan, Z.; Guo, J. External Uncertainty and Raw Material Price in Energy Transition: Implications for Green. *Renew. Energy* **2025**, *241*, 122354. [CrossRef]
3. Zhao, Z.; Zhang, W.; Cui, K.; Jin, S.; Liu, K.; Yang, S.; Lai, Y.; Chang, X. Multistage gradient crystallization study towards lithium carbonate crystal growth. *Sep. Purif. Technol.* **2025**, *360*, 130955. [CrossRef]
4. Ebensperger, A.; Maxwell, P.; Moscoso, C. The lithium industry: Its recent evolution and future prospects. *Resour. Policy* **2005**, *30*, 218–231. [CrossRef]
5. Ambrose, H.; Kendall, A. Understanding the future of lithium: Part 1, resource model. *J. Ind. Ecol.* **2020**, *24*, 80–89. [CrossRef]
6. Bajolle, H.; Lagadic, M.; Louvet, N. The future of lithium-ion batteries: Exploring expert conceptions, market trends, and price scenarios. *Energy Res. Soc. Sci.* **2022**, *93*, 102850. [CrossRef]
7. Zhang, X.; Chang, H.L.; Su, C.W.; Qin, M.; Umar, M. Exploring the dynamic interaction between geopolitical risks and lithium prices: A time-varying analysis. *Resour. Policy* **2024**, *90*, 104840. [CrossRef]
8. Su, X.; Zhao, Y. Risk spillovers between Chinese new energy futures and carbon-intensive assets: Asymmetric effect, time–frequency dynamics, and portfolio strategies. *North Am. J. Econ. Financ.* **2025**, *75*, 102275. [CrossRef]
9. Li, X.; Sengupta, T.; Mohammed, K.S.; Jamaani, F. Forecasting the lithium mineral resources prices in China: Evidence with Facebook Prophet (Fb-P) and Artificial Neural Networks (ANN) methods. *Resour. Policy* **2023**, *82*, 103580. [CrossRef]
10. Sai, W.; Pan, Z.; Liu, S.; Jiao, Z.; Zhong, Z.; Miao, B.; Chan, S.H. Event-driven forecasting of wholesale electricity price and frequency regulation price using machine learning algorithms. *Appl. Energy* **2023**, *352*, 121989. [CrossRef]
11. Ding, G.; Qin, L. Study on the prediction of stock price based on the associated network model of LSTM. *Int. J. Mach. Learn. Cybern.* **2020**, *11*, 1307–1317. [CrossRef]
12. Jailani, N.L.M.; Dhanasegaran, J.K.; Alkawsi, G.; Alkahtani, A.A.; Phing, C.C.; Baashar, Y.; Capretz, L.F.; Al-Shetwi, A.Q.; Tiong, S.K. Investigating the power of LSTM-based models in solar energy forecasting. *Processes* **2023**, *11*, 1382. [CrossRef]
13. Guo, F.; Deng, S.; Zheng, W.; Wen, A.; Du, J.; Huang, G.; Wang, R. Short-Term Electricity Price Forecasting Based on the Two-Layer VMD Decomposition Technique and SSA-LSTM. *Energies* **2022**, *15*, 8445. [CrossRef]
14. Han, M.; Zhong, J.; Sang, P.; Liao, H.; Tan, A. A combined model incorporating improved SSA and LSTM algorithms for short-term load forecasting. *Electronics* **2022**, *11*, 1835. [CrossRef]
15. Wang, K.; Fan, X.; Yang, X.; Zhou, Z. An AQI decomposition ensemble model based on SSA-LSTM using improved AMSSA-VMD decomposition reconstruction technique. *Environ. Res.* **2023**, *232*, 116365. [CrossRef]

16. Wang, J.; Jiang, L.; Wang, L. Prediction of China's Polysilicon Prices: A Combination Model Based on Variational Mode Decomposition, Sparrow Search Algorithm and Long Short-Term Memory. *Mathematics* **2024**, *12*, 3690. [CrossRef]
17. Wang, J.; Zhang, D.; Huang, Q.; Cui, Z. Multiple-step accurate prediction of wave energy: A hybrid model based on quadratic decomposition, SSA and LSTM. *Int. J. Green Energy* **2025**, *22*, 100–123. [CrossRef]
18. Dragomiretskiy, K.; Zosso, D. Variational mode decomposition. *IEEE Trans. Signal Process.* **2013**, *62*, 531–544. [CrossRef]
19. Wu, J.; Hu, Y.; Wu, D.; Yang, Z. An aquatic product price forecast model using VMD-IBES-LSTM hybrid approach. *Agriculture* **2022**, *12*, 1185. [CrossRef]
20. Liu, T.; Ma, X.; Li, S.; Zhang, C. A stock price prediction method based on meta-learning and variational mode decomposition. *Knowl. Based Syst.* **2022**, *252*, 109324. [CrossRef]
21. Xue, J.; Shen, B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control Eng.* **2020**, *8*, 22–34. [CrossRef]
22. Awadallah, M.A.; Al-Betar, M.A.; Doush, I.A.; Makhadmeh, S.N.; Al-Naymat, G. Recent versions and applications of sparrow search algorithm. *Arch. Comput. Methods Eng.* **2023**, *30*, 2831–2858. [CrossRef] [PubMed]
23. Hochreiter, S. Long Short-term Memory. In *Neural Computation*; MIT-Press: Cambridge, MA, USA, 1997.
24. Bhandari, H.N.; Rimal, B.; Pokhrel, N.R.; Rimal, R.; Dahal, K.R.; Khatri, R.K. Predicting stock market index using LSTM. *Mach. Learn. Appl.* **2022**, *9*, 100320. [CrossRef]
25. Wu, H.; Chen, S.; Ding, Y. Comparison of ARIMA and LSTM for Stock Price Prediction. *Financ. Eng. Risk Manag.* **2023**, *6*, 1–7.
26. Mu, G.; Gao, N.; Wang, Y.; Dai, L. A stock price prediction model based on investor sentiment and optimized deep learning. *IEEE Access* **2023**, *11*, 51353–51367. [CrossRef]
27. Liu, H.; Li, D.; Yuan, Y.; Zhang, S.; Zhao, H.; Deng, W. Fault diagnosis for a bearing rolling element using improved VMD and HT. *Appl. Sci.* **2019**, *9*, 1439. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Transformer-Based Models for Probabilistic Time Series Forecasting with Explanatory Variables

Ricardo Caetano [1,†], José Manuel Oliveira [2,3,*,†] and Patrícia Ramos [2,4,†]

[1] ISCAP, Polytechnic of Porto, Rua Jaime Lopes Amorim s/n, 4465-004 São Mamede de Infesta, Portugal; 2090310@iscap.ipp.pt
[2] Institute for Systems and Computer Engineering, Technology and Science, Campus da FEUP, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal; patricia@iscap.ipp.pt
[3] Faculty of Economics, University of Porto, Rua Dr. Roberto Frias, 4200-464 Porto, Portugal
[4] CEOS.PP, ISCAP, Polytechnic of Porto, Rua Jaime Lopes Amorim s/n, 4465-004 São Mamede de Infesta, Portugal
* Correspondence: jmo@fep.up.pt
† These authors contributed equally to this work.

Abstract: Accurate demand forecasting is essential for retail operations as it directly impacts supply chain efficiency, inventory management, and financial performance. However, forecasting retail time series presents significant challenges due to their irregular patterns, hierarchical structures, and strong dependence on external factors such as promotions, pricing strategies, and socio-economic conditions. This study evaluates the effectiveness of Transformer-based architectures, specifically Vanilla Transformer, Informer, Autoformer, ETSformer, NSTransformer, and Reformer, for probabilistic time series forecasting in retail. A key focus is the integration of explanatory variables, such as calendar-related indicators, selling prices, and socio-economic factors, which play a crucial role in capturing demand fluctuations. This study assesses how incorporating these variables enhances forecast accuracy, addressing a research gap in the comprehensive evaluation of explanatory variables within multiple Transformer-based models. Empirical results, based on the M5 dataset, show that incorporating explanatory variables generally improves forecasting performance. Models leveraging these variables achieve up to 12.4% reduction in Normalized Root Mean Squared Error (NRMSE) and 2.9% improvement in Mean Absolute Scaled Error (MASE) compared to models that rely solely on past sales. Furthermore, probabilistic forecasting enhances decision making by quantifying uncertainty, providing more reliable demand predictions for risk management. These findings underscore the effectiveness of Transformer-based models in retail forecasting and emphasize the importance of integrating domain-specific explanatory variables to achieve more accurate, context-aware predictions in dynamic retail environments.

Keywords: transformers; time series; probabilistic forecasting; retail; covariates; deep learning; data-driven decision making

MSC: 68T07

1. Introduction

Accurate forecasting models are fundamental to the retail industry as they play a pivotal role in optimizing operations, enhancing customer satisfaction, and improving financial performance [1]. Retail businesses operate in a complex environment that is influenced by

dynamic consumer behavior, seasonal trends, promotional activities, and external factors, such as economic conditions and weather. As a result, the ability to anticipate demand accurately is essential for effective decision making at strategic, tactical, and operational levels [2].

At the strategic level, forecasting models inform long-term decisions such as market entry strategies, channel development, and store location planning. These decisions require robust aggregate sales forecasts to understand market trends and the potential impacts of technological advancements or competitive shifts. For example, accurate forecasts enable retailers to decide whether to expand into online channels or develop smaller, local stores in response to evolving consumer preferences. Tactically, forecasts guide mid-term planning, such as promotional strategies, category management, and inventory allocation. Retailers use these models to determine optimal pricing, promotional frequencies, and assortments that maximize profitability while minimizing waste. Accurate forecasts also ensure product availability during peak demand periods, maintaining high service levels and strengthening customer loyalty. Operationally, accurate forecasts address immediate needs such as store-level inventory management, workforce scheduling, and replenishment planning. These tasks require high-granularity data, often at the Stock-Keeping Unit (SKU) level, to minimize stockouts and overstocking [3]. For instance, ensuring sufficient inventory levels during a promotional campaign avoids missed sales opportunities while preventing excess stock that can lead to markdowns or spoilage. Moreover, the financial implications of inaccurate forecasting are significant. Retail operates on thin margins, where misaligned inventory levels can lead to substantial losses. Overestimations result in higher storage costs and markdowns, while underestimations lead to lost sales and customer dissatisfaction. Accurate forecasting models mitigate these risks, providing a balance between demand and supply, which is crucial for cash flow optimization and profitability.

Deep learning models have emerged as a superior approach to time series forecasting in retail, surpassing traditional statistical methods in handling the complexities and dynamic demands of this domain [4]. Statistical models such as ARIMA or exponential smoothing excel in forecasting tasks with straightforward trends and seasonality, but they often struggle when dealing with high-dimensional, hierarchical data structures, irregular sales patterns, and the integration of external influencing factors [5]. In contrast, deep learning models are capable of capturing intricate temporal patterns and dependencies across multiple time series [6,7]. Empirical evidence from the M4 competition and subsequent Kaggle competitions underscores the performance superiority of deep learning models in diverse scenarios [8]. For example, the Wikipedia Web Traffic competition demonstrated the ability of recurrent neural networks (RNNs) to outperform statistical benchmarks by effectively modeling long-term dependencies and incorporating contextual data. Similarly, the Corporación Favorita Grocery Sales competition showcased how ensembles of neural networks and gradient boosting methods excelled in scenarios involving hierarchical and disaggregated sales data. Another critical advantage of deep learning is its capacity for cross-learning, where patterns are learned across multiple time series [9]. This contrasts with traditional models that often require separate parameter estimation for each time series. Cross-learning enables deep learning models to generalize better and produce more robust forecasts, particularly in cases of sparse or noisy data.

Additionally, deep learning methods provide probabilistic forecasts, allowing for the estimation of uncertainty and prediction intervals, a critical aspect in retail decision making for inventory management and promotional planning. These capabilities enable retailers to align supply with demand more effectively, reduce costs from overstocking, and mitigate risks of stockouts [10].

The Transformer architecture has revolutionized deep learning, particularly in applications requiring efficient handling of sequential data [11]. While traditional neural networks and RNNs were pivotal in the earlier stages of sequence modeling, they face specific limitations that restrict their effectiveness in capturing complex dependencies within sequences [12].

The integration of external variables is essential for improving demand forecasting accuracy in retail settings. Retail sales are influenced by a wide range of factors beyond historical sales patterns, including promotional activities, pricing strategies, socio-economic conditions, calendar events, and even weather fluctuations. According to [2], product-level sales data exhibit significant volatility and seasonality, with multiple demand drivers affecting purchasing behavior, such as marketing campaigns, stockouts, and changes in local demographics. The inclusion of such external variables enables forecasting models to not only enhance predictive accuracy, but to also support better inventory management, promotional planning, and overall retail decision making.

This paper introduces a comprehensive approach to probabilistic time series forecasting in retail using Transformer-based deep learning models. This study highlights the integration of explanatory variables, such as promotions, pricing, and socio-economic indicators, demonstrating their impact on improving forecast accuracy. The key contributions of this paper include the following:

- Development of Transformer-Based Forecasting Models: This study explores various Transformer-based architectures tailored for retail demand forecasting, including Vanilla Transformer, Informer, Autoformer, ETSformer, NSTransformer, and Reformer. These models are evaluated on their ability to capture the long-term dependencies, seasonality, and external factors affecting sales patterns.
- Incorporation of Explanatory Variables: This research emphasizes the importance of integrating explanatory variables, such as calendar events, promotional activities, pricing, and socio-economic factors, in improving forecast accuracy. The models effectively leverage these covariates to address the complexities of retail data. To the best of our knowledge, this is the first study to comprehensively evaluate the impact of such explanatory variables within multiple Transformer-based forecasting architectures for retail demand prediction.
- Probabilistic Forecasting: The models provide probabilistic forecasts, capturing the uncertainty associated with demand predictions. This feature is crucial for risk management and decision-making processes in retail operations, ensuring a more resilient inventory management strategy.
- Empirical Evaluation Using Real-World Data: This paper includes a thorough empirical evaluation using the M5 dataset, a comprehensive retail dataset provided by Walmart. The results demonstrate the robustness and effectiveness of the proposed models in improving forecast accuracy across various retail scenarios.

The remainder of this paper is structured as follows. Section 2 provides a comprehensive review of the recent advancements in retail time series forecasting, highlighting the evolution of deep learning models and the integration of explanatory variables. Section 3 describes the Transformer architectures used in this study and their application to probabilistic time series forecasting. Section 4 presents the dataset used, the experimental setup, and the results of the model evaluations, emphasizing the performance improvements achieved by the proposed approaches. Section 5 summarizes the key findings of this research, discusses the implications for retail operations, and suggests directions for future work.

2. Related Work

Recent advancements in retail time series forecasting have been driven by deep learning models, explanatory variables, and probabilistic forecasting methods to account for uncertainty [13]. Given the complexity of forecasting across various domains, the literature review is structured into three key subsections. The first examines how neural networks and Transformer-based architectures address challenges, such as hierarchical structures and irregular demand patterns. The second explores the influence of external factors—such as pricing, promotions, and macroeconomic indicators—on predictive accuracy. The third discusses methods for quantifying uncertainty, which is essential for managing risks in inventory, pricing, and supply chain operations.

2.1. Retail Time Series Forecasting with Deep Learning

The state of the art in deep learning for time series forecasting in retail involves a range of innovative models and hybrid techniques to address the complexities of retail sales data [14]. Recent research has introduced diverse deep learning architectures designed to enhance the accuracy of sales forecasting in different retail contexts.

Bandara et al. [15] presented a demand forecasting framework for e-commerce using LSTM networks. By leveraging cross-series information from related products in a product hierarchy, their model provides accurate forecasts while addressing the challenges of non-stationary, sparse, and highly intermittent sales data. The proposed LSTM-based method significantly outperforms state-of-the-art univariate techniques, demonstrating its effectiveness for large-scale retail forecasting.

Joseph et al. [16] proposed a hybrid deep learning framework combining Convolutional Neural Networks (CNN) with Bi-directional Long Short-Term Memory (BiLSTM) for store item demand forecasting. By utilizing CNN for feature extraction and BiLSTM for modeling temporal dependencies, the framework aims to enhance accuracy in predicting retail demand. Their approach, which employs Lazy Adam optimization, significantly outperforms traditional machine learning models, achieving lower forecasting errors and improving inventory decisions in the retail context.

Giri and Chen [17] presented a deep learning framework for demand forecasting in the fashion and apparel retail industry. The proposed model combines image features of clothing items with sales data to predict weekly demand for new fashion products. The approach uses machine learning clustering to categorize products based on sales profiles and image similarity, resulting in accurate predictions, even for newly launched items without extensive historical data. Their study demonstrated the potential of integrating visual attributes and sales data to enhance forecast accuracy in fashion retail.

Mogarala Guruvaya et al. [18] proposed a Bi-GRU-APSO model, which combines Bi-Directional Gated Recurrent Units (Bi-GRU) with Adaptive Particle Swarm Optimization (APSO) for retail sales forecasting. This hybrid approach uses feature selection techniques, including APSO, Recursive Feature Elimination (RFE), and Minimum Redundancy Maximum Relevance (MRMR), to enhance the accuracy and computational efficiency of forecasts. Their model demonstrated superior performance on benchmark datasets, achieving higher accuracy metrics compared to conventional models, making it suitable for multi-channel retail sales forecasting.

de Castro Moraes et al. [19] presented a comparative analysis of deep learning models for optimizing single-period inventory decisions, focusing on the Newsvendor Problem. Their study evaluated different deep learning architectures, including MLP, CNN, RNN, and LSTM, to determine their impact on inventory optimization by providing accurate demand forecasts. The results indicate that recurrent models, especially RNNs and LSTMs,

outperform others in minimizing inventory mismatch costs. The research also shows that data-driven approaches that leverage empirical error distributions significantly outperform traditional model-based inventory methods.

de Castro Moraes et al. [20] proposed hybrid deep learning models combining Convolutional Neural Networks with Long Short-Term Memory for retail sales forecasting. Their study introduced stacked (S-CNN-LSTM) and parallel (P-CNN-LSTM) hybrid architectures to capture both temporal dependencies and external features in retail data. The models were evaluated using real-world retail datasets, outperforming simpler neural network architectures and standard autoregressive methods, while reducing computational complexity and improving both short-term and long-term forecasting accuracy.

Additionally, Wu et al. [21] proposed a two-stage deep learning model called OCCPH-MHA for enhancing sales forecasting in multi-channel retail. The first stage uses a heterogeneous graph neural network to identify consumer group preferences based on purchase history, while the second stage integrates these preferences with time-series demand data using multi-head attention mechanisms. The model significantly improves sales forecast accuracy for multi-channel retail environments by leveraging consumer behavior insights and product preferences, showcasing its robustness in predicting demand across both online and offline channels.

Finally, Sousa et al. [22] developed a two-stage model for predicting demand for new products in fashion retail using censored data. The first stage involved transforming historical sales data into demand using multiple heuristics and an Expectation–Maximization (EM) algorithm to estimate demand during stockout events. The second stage used machine learning models—Random Forest, Deep Neural Networks, and Support Vector Regression—to predict demand for new products based on the features of similar past items. The EM algorithm and Random Forest provided the most accurate predictions, demonstrating the model's effectiveness in improving production management decisions for new product launches.

2.2. Explanatory Variables in Retail Demand Forecasting

The use of explanatory variables in retail time series forecasting has gained significant traction as researchers have recognized the importance of incorporating external and contextual data to improve the accuracy of sales predictions. Various studies have highlighted how the integration of different external variables can enhance the performance of deep learning models in predicting retail sales.

Huang et al. [23] explored the impact of competitive information, such as competitor prices and promotions, on forecasting the sales of Fast-Moving Consumer Goods (FMCGs) at the SKU level. The authors proposed a two-stage approach, involving variable selection and factor analysis, to effectively refine competitive explanatory variables, integrating them into an Autoregressive Distributed Lag (ADL) model. The study demonstrated that incorporating competitive information significantly improved forecasting accuracy compared to traditional methods, highlighting the importance of competitive dynamics in retail sales predictions.

Loureiro et al. [24] explored the application of Deep Neural Networks (DNNs) for sales forecasting in the fashion retail industry. Their study incorporated a wide set of explanatory variables, including physical product characteristics and domain expert opinions, to predict the sales of new fashion products. The results showed that, while the DNN performed well, its improvements over simpler methods, like Random Forest, were not always significant. The findings emphasize the importance of using both advanced modeling techniques and domain expertise to enhance sales predictions in fashion retail.

Punia et al. [25] proposed a hybrid forecasting method combining Long Short-Term Memory networks and Random Forest (RF) for demand forecasting in multi-channel retail. The model leverages LSTM for temporal relationships and RF for handling explanatory variables, improving the accuracy across both online and offline sales channels. Empirical evaluations show that the hybrid method outperforms other benchmark methods, demonstrating robustness in managing complex demand patterns across multiple channels in retail.

Lim et al. [26] introduced the Temporal Fusion Transformer (TFT), an attention-based architecture for multi-horizon time series forecasting. TFT combines recurrent layers for local processing with self-attention layers to model long-term dependencies, enabling both high performance and interpretability. The model's specialized components, such as gating mechanisms and variable selection networks, facilitate feature selection and enhance the relevance of temporal information. TFT demonstrated significant improvements in forecasting accuracy over benchmark models, making it suitable for retail and other applications that require reliable and interpretable multi-step predictions.

Wang [27] proposed a novel framework that incorporates economic indicators and dynamic interactions to improve sales forecasting for different retail sectors, such as hypermarkets, supermarkets, and convenience stores. By identifying influential economic predictors like the Consumer Price Index (CPI), Retail Employment Population (REP), and real wage, as well as by considering the competitive interactions between retail channels, the model enhances forecasting accuracy and provides managerial insights into sector-specific trends. Their study demonstrates the potential of integrating macroeconomic indicators and inter-sector dynamics for optimized retail inventory and sales management.

Kao and Chueh [28] presented a deep learning-based model for purchase forecasting aimed at reducing waste in food products with short shelf lives. The model uses Artificial Neural Networks (ANNs) to predict purchase quantities by incorporating factors such as store environment, weather, and consumer behavior. The proposed approach, tested on a cream puff product, effectively reduces forecasting errors with a Mean-Square Percentage Error (MSPE) of less than 6%. Their study demonstrates the potential of integrating ANN-based forecasting into merchandising to enhance inventory efficiency and sustainability in retail operations.

Ramos et al. [29] examined the use of shrinkage and dimensionality reduction techniques, specifically ridge regression and principal component analysis (PCA), for forecasting seasonal sales in retail. Their study focused on integrating multiple demand drivers, such as promotions and pricing, into statistical models like ARIMA and ETS. Empirical results using supermarket sales data showed that PCA-based models performed better during promotional periods, while shrinkage estimators outperformed alternatives during non-promotional periods, resulting in approximately 10% accuracy improvement over benchmark models.

Punia and Shankar [30] proposed a deep learning-based decision support system for demand forecasting in retail, integrating sequence modeling with machine learning methods. Their model effectively captures both temporal and covariate-based variations in demand data using structured and unstructured data sources, including promotions, weather, and economic indicators. The results demonstrated that the proposed ensemble model outperformed traditional statistical benchmarks, enhancing forecast accuracy and enabling more informed inventory and promotion planning for retailers.

Nasseri et al. [31] conducted a comparative study on the application of tree-based ensemble models, specifically Extra Tree Regressors (ETRs), and Long Short-Term Memory networks for retail demand prediction. Utilizing a dataset of over 5.2 million records,

including external factors like weather and COVID-19 data, their study found that ETR outperformed LSTM across multiple evaluation metrics, particularly in perishable product categories. This demonstrates the robustness of tree-based ensemble methods for capturing complex patterns in retail demand forecasting.

Ramos and Oliveira [7] investigated the impact of incorporating static and dynamic covariates into deep learning models for sales forecasting. Using the DeepAR model, their study tested various combinations of time-, event-, price-, and ID-related features using the M5 competition dataset. Results indicated that incorporating time, event, and ID features significantly improved forecast accuracy, while price features offered minimal benefits. The optimal model achieved a 1.8% RMSSE (Root Mean Scaled Squared Error) and 6.5% MASE (Mean Absolute Scaled Error) improvement, emphasizing the value of feature integration for enhancing prediction reliability in retail forecasting.

Wellens et al. [32] presented a simplified decision tree framework for retail sales forecasting that effectively integrates explanatory variables. Their study demonstrated that a streamlined implementation of tree-based machine learning methods, using variables such as promotions and national events, significantly outperforms traditional statistical models while maintaining computational efficiency. The framework's success is largely attributed to the inclusion of feature engineering and explanatory variables, which improve forecast accuracy and reduce inventory costs, thereby making it more accessible for practical adoption by traditional retailers.

Praveena and Prasanna Devi [33] proposed a hybrid deep learning model called Deep Prophet Memory Neural Network (DPMNN) for seasonal item demand forecasting in retail. By integrating temporal, historical, trend, and seasonal data, DPMNN outperformed state-of-the-art models such as LSTM and Prophet in reducing forecasting errors like RMSE and MAPE. Their study demonstrates the efficacy of combining feature selection techniques with deep learning to optimize retail inventory management, effectively reducing overstock and stockouts.

2.3. Probabilistic Forecasting of Time Series Using Deep Learning

Probabilistic time series forecasting has gained prominence as an effective method for capturing uncertainty in predictions, providing valuable insights for decision making across domains such as retail, finance, and supply chain management.

Wen et al. [34] introduced the Multi-Horizon Quantile Recurrent Forecaster (MQ-RNN), a probabilistic forecasting framework that combines recurrent and convolutional neural networks with quantile regression for multi-step time series prediction. The model leverages both temporal and static covariates, effectively handling challenges like shifting seasonality, cold starts, and planned event spikes. By adopting a direct multi-horizon strategy, MQ-RNN mitigates the error accumulation commonly found in recursive forecasting methods, providing stable and efficient performance, as demonstrated in applications for retail demand and energy forecasting.

DeepAR, proposed by Salinas et al. [35], is another deep learning model that uses an autoregressive recurrent neural network to learn from related time series for probabilistic forecasting. By training on a large number of similar time series, DeepAR produces more accurate forecasts compared to traditional methods while effectively capturing the distribution of future values. This model uses an autoregressive framework that can integrate diverse data, providing flexibility for large-scale applications like retail demand prediction where individual time series are related through shared features such as product categories.

Rasul et al. [36] introduced TimeGrad, an autoregressive denoising diffusion model for multivariate probabilistic time series forecasting. The model employs diffusion proba-

bilistic methods, leveraging gradient estimation to generate accurate probabilistic forecasts for complex time series data with thousands of correlated dimensions. TimeGrad utilizes Langevin sampling to convert noise into samples of the distribution of interest. Experimental results demonstrated that TimeGrad sets a new state-of-the-art performance in multivariate probabilistic forecasting, outperforming existing methods across a range of real-world datasets.

Rasul et al. [37] proposed a model for multivariate probabilistic time series forecasting using conditioned normalizing flows. Their approach combines autoregressive deep learning techniques with normalizing flows to capture complex dependencies across time series, enabling accurate probabilistic predictions. The model achieves scalability while retaining high-dimensional dependency representation, making it suitable for scenarios involving thousands of interacting time series. Empirical evaluations on various real-world datasets demonstrated that this method outperformed existing baseline models in terms of accuracy and computational efficiency.

Hasson et al. [38] introduced the Level Set Forecaster (LSF), a novel algorithm designed to transform any point estimator into a probabilistic forecaster. By leveraging the grouping of similar predictions into partitions, LSF creates consistent probabilistic forecasts, particularly when used with tree-based models like XGBoost. Empirical evaluations demonstrated that LSF rivals state-of-the-art deep learning models in forecasting accuracy, providing a significant advancement in turning point predictions into probabilistic forecasts effectively.

Rangapuram et al. [39] proposed an end-to-end approach for generating coherent probabilistic forecasts for hierarchical time series. Unlike traditional two-step methods that require separate reconciliation processes, this model incorporates reconciliation as part of a single trainable framework, ensuring coherent predictions across all levels of a hierarchy. By leveraging the reparameterization trick and a differentiable convex optimization layer, the model is capable of simultaneously learning from all time series in a hierarchy while maintaining coherence without post-processing. Empirical results demonstrated significant improvements in forecast accuracy, making this approach highly effective for large-scale applications like retail and energy demand forecasting.

Kan et al. [40] proposed the Multivariate Quantile Function Forecaster (MQF2), a probabilistic forecasting method designed to improve multi-horizon predictions using a multivariate quantile function. MQF2 combines elements of autoregressive and sequence-to-sequence models to capture the dependency structure across time, thereby avoiding error accumulation and quantile crossing. The model is particularly effective in inventory management scenarios, enhancing forecasting accuracy for supply chain decisions by integrating dependencies like product cannibalization and substitutability.

Shchur et al. [41] introduced AutoGluon–TimeSeries, an open-source AutoML library designed for probabilistic time series forecasting. The framework enables users to generate accurate point and quantile forecasts with minimal coding effort by leveraging ensembles of diverse forecasting models. AutoGluon–TimeSeries demonstrated strong empirical performance on 29 benchmark datasets, outperforming existing methods in terms of both point and probabilistic forecast accuracy, making it a robust solution for practitioners with varying levels of expertise.

Tong et al. [42] introduced a hierarchical Transformer model with probabilistic decomposition, called Probabilistic Decomposition Transformer (PDTrans), which is designed to mitigate the cumulative errors common in autoregressive forecasting. By combining a Transformer for primary autoregressive forecasting with a conditional generative model, PDTrans enables hierarchical, probabilistic, and interpretable forecasts. The model effec-

tively separates seasonal and trend components, providing accurate forecasts for complex temporal patterns, as demonstrated across multiple time series datasets.

Sprangers et al. [43] introduced a Bidirectional Temporal Convolutional Network (BiTCN) for probabilistic time series forecasting, focusing on reducing the parameter count required by traditional Transformer-based methods. The model uses two temporal convolutional networks to, respectively, encode future covariates and past observations, enabling efficient and accurate forecasting. Their study demonstrated that BiTCN performs on par with state-of-the-art methods while requiring fewer parameters, significantly reducing both memory usage and training costs, thus making it a more accessible option for large-scale forecasting tasks.

Lastly, Olivares et al. [44] introduced the Deep Poisson Mixture Network (DPMN) for probabilistic hierarchical forecasting. The model combines neural networks with a mixture of Poisson distributions to produce coherent forecasts at different aggregation levels without requiring explicit reconciliation steps. DPMN ensures hierarchical coherence, making it particularly effective for large-scale forecasting tasks. Their empirical evaluations demonstrated significant improvements over existing methods, achieving an 11.8% better CRPS score on Australian tourism data and an 8.1% improvement on grocery sales data.

3. Probabilistic Forecasting with Transformer-Based Models

The Transformer architecture's reliance on attention mechanisms, rather than recurrence, allows for significant parallelization, which reduces training time while maintaining high performance. The use of self-attention throughout the encoder and decoder stacks enables the model to effectively capture long-range dependencies in the data, making it especially powerful for tasks that involve complex sequential relationships. In time series forecasting, sequences of numerical observations are treated similarly to sequences of words or tokens in language models as both require understanding and capturing dependencies across ordered elements. This analogy is reflected in the application of Transformer-based architectures, as introduced by Vaswani et al. [45], which were originally developed for natural language processing but have proven highly effective for time series tasks [26], where learning complex temporal patterns is akin to learning the relationships between words in a sentence.

In the following section, the Vanilla Transformer [45], Informer [46], Autoformer [47], ETSformer [48], NSTransformer [49], and Reformer [50] architectures are presented. These models were chosen for their availability [51], widespread use [52], and demonstrated effectiveness in performance assessment [13], providing a balanced comparison between well-established approaches and recent advancements tailored specifically for time series forecasting. These architectures were also employed in the empirical study conducted for this paper.

3.1. Deep Learning Transformers for Time Series Forecasting

Vaswani et al. [45] introduced the Transformer architecture, which revolutionized the field of deep learning by relying entirely on attention mechanisms rather than traditional recurrent or convolutional layers for sequence transduction tasks. The architecture is composed of an encoder–decoder structure, where both the encoder and decoder are built using multiple identical layers stacked on top of each other. The encoder comprises n_e identical layers, each of which includes a multi-head self-attention mechanism and a position-wise feed-forward network. Each layer uses residual connections followed by layer normalization, allowing the model to retain information and stabilize training. The attention mechanism enables the encoder to capture the dependencies between all elements in the input sequence, regardless of their relative positions. The decoder also consists of n_d identical layers, but with an additional sub-layer compared to the encoder. In each decoder layer, multi-head self-attention is combined with encoder–decoder attention, allowing the decoder to attend to the output

of the encoder stack. Additionally, a masking mechanism is applied to prevent positions from attending to subsequent positions, ensuring that the model maintains its autoregressive properties. Attention mechanisms are the core of the Transformer architecture, enabling it to effectively weigh the relevance of different parts of the input sequence.

The computation of attention relies on three main components: queries (**Q**), keys (**K**), and values (**V**). To derive these components, the input matrix $\mathbf{Y} \in \mathbb{R}^{L \times D}$ is multiplied by learnable weight matrices for queries, keys, and values, yielding $\mathbf{Q} \in \mathbb{R}^{L \times D_k}$, $\mathbf{K} \in \mathbb{R}^{L \times D_k}$, and $\mathbf{V} \in \mathbb{R}^{L \times D_v}$:

$$\mathbf{Q} = \mathbf{Y}\mathbf{W}^Q, \quad \mathbf{K} = \mathbf{Y}\mathbf{W}^K, \quad \mathbf{V} = \mathbf{Y}\mathbf{W}^V. \tag{1}$$

Using these matrices, the attention mechanism computes query–key interactions by multiplying **Q** with the transpose of **K**, then applying a scaling factor, followed by a softmax activation, and, finally, by multiplying with **V**. This results in a matrix of size $L \times D$. To address numerical instability and prevent the vanishing gradient problem during training, the dot product is scaled by dividing by the square root of the key dimension D_k. The final output of self-attention, where each row corresponds to the output vector for a given query, is computed as follows:

$$\text{SelfAttention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{D_k}}\right)\mathbf{V}. \tag{2}$$

Multi-head attention uses multiple sets of learned projections to perform attention in parallel, allowing the model to attend to different subspaces of the input information simultaneously. To handle the sequential nature of data, the model also incorporates positional encodings that provide information about the relative positions of elements in the sequence. This is crucial as the architecture lacks any recurrence or convolution, making it necessary to explicitly add position information to allow the model to understand the order of the sequence. Figure 1 provides a detailed depiction of this architecture, along with the others discussed in this section, highlighting the specific components and attention mechanisms that characterize each Transformer variant.

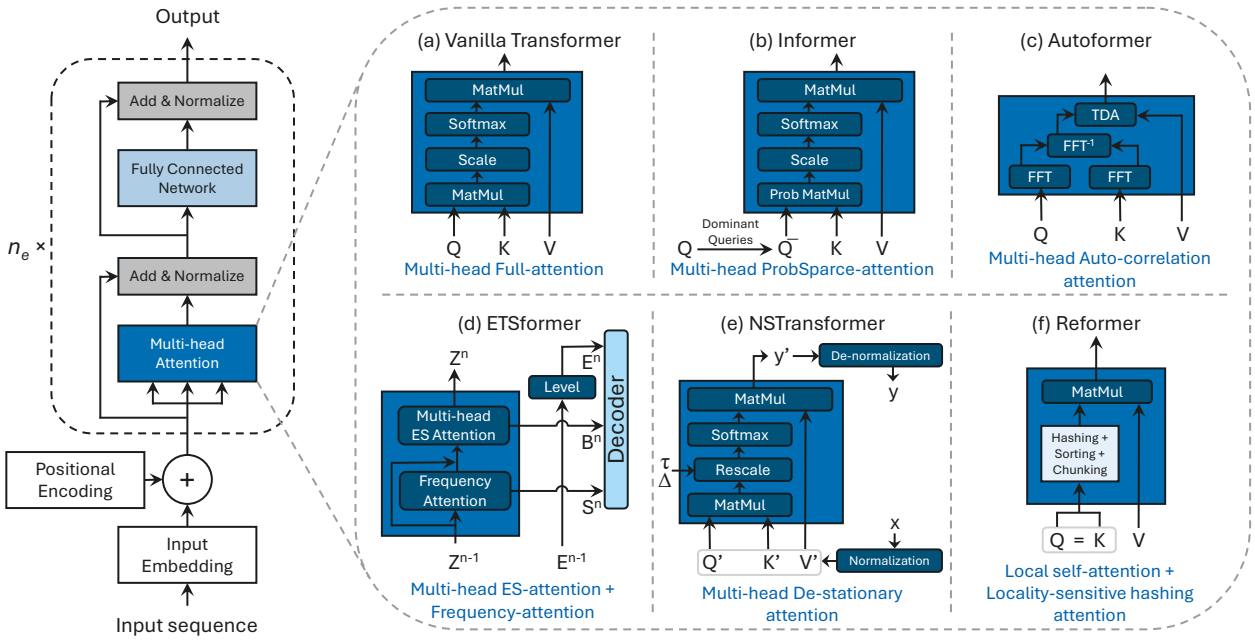


Figure 1. Attention mechanisms of the Transformer architectures used in this study: (**a**) Vanilla Transformer; (**b**) Informer; (**c**) Autoformer; (**d**) ETSformer; (**e**) NSTransformer; and (**f**) Reformer.

In their work, Zhou et al. [46] introduce a new Transformer-based architecture designed specifically for the challenges of Long Sequence Time-Series Forecasting (LSTF). The architecture, named Informer, focuses on improving computational efficiency and scalability for long input sequences, addressing the limitations of traditional Transformer models like high computational complexity and memory usage. The Informer architecture follows an encoder–decoder framework but with several key innovations. The encoder uses the ProbSparse self-attention mechanism, which replaces the canonical dot-product self-attention with a probabilistic sampling approach. This allows Informer to achieve a time complexity and memory usage of $\mathcal{O}(L \log(L))$, significantly reducing the quadratic complexity typically seen in standard Transformer architectures, making it suitable for long sequence data.

Additionally, self-attention distilling is applied within the encoder to highlight dominant attention scores and reduce redundant input combinations. This operation significantly compresses the attention map, reducing the space complexity while still preserving important information. The encoder outputs a refined representation that maintains robust long-range dependencies. The decoder employs a generative style that predicts the entire output sequence in a single forward pass rather than the traditional step-by-step dynamic decoding. This drastically improves the inference speed, particularly for long sequences, and it prevents the accumulation of errors that is common in autoregressive decoding. The combination of ProbSparse attention, self-attention distilling, and the generative decoder makes Informer an efficient and scalable solution for long-term forecasting. The Informer model has demonstrated superior performance in capturing long-range dependencies while being computationally feasible for very large datasets, making it highly suitable for empirical studies involving long sequence time-series forecasting in various domains, such as finance and energy.

Wu et al. [47] proposed a novel architecture specifically designed to improve long-term time series forecasting called Autoformer. This model innovates by moving beyond the limitations of the canonical Transformer, particularly addressing the inefficiencies associated with traditional self-attention mechanisms in long-term forecasting contexts. The Autoformer architecture follows an encoder–decoder framework but diverges from the typical self-attention approach by incorporating series decomposition blocks and an Auto-Correlation mechanism. The encoder is designed to eliminate the trend-cyclical components using series decomposition blocks, which allows it to focus on modeling seasonal patterns effectively. Each encoder layer includes a series decomposition operation, which progressively separates the seasonal and trend components, making the hidden representations more suitable for accurate long-term forecasting. In the decoder, Autoformer includes an accumulation mechanism for trend components and stacked Auto-Correlation blocks for refining seasonal components.

The unique Auto-Correlation mechanism replaces self-attention to discover dependencies based on series periodicity and to aggregate similar sub-series, thus enhancing both computational efficiency and the utilization of information from the entire sequence. This mechanism reduces the computational complexity from the quadratic order (as seen in Vanilla Transformers) to $\mathcal{O}(L \log(L))$, making it feasible for long-term sequences. The combination of progressive decomposition and Auto-Correlation mechanisms allows Autoformer to handle intricate temporal patterns more effectively while maintaining computational efficiency. Empirical evaluations have shown that Autoformer achieves state-of-the-art results across multiple benchmarks in applications such as energy, traffic, economics, weather, and disease forecasting.

Woo et al. [48] presented a novel Transformer architecture tailored for time-series forecasting by combining traditional exponential smoothing concepts with the Transformer

framework. The model, named ETSformer, was specifically designed to enhance long-term time series prediction while maintaining interpretability and computational efficiency. The ETSformer architecture builds on an encoder–decoder design that incorporates Exponential Smoothing Attention (ESA) and Frequency Attention (FA) mechanisms to address the limitations of the Vanilla self-attention used in standard Transformers. The architecture consists of modular decomposition blocks that extract time-series components like level, growth, and seasonality at each layer, effectively breaking down complex time series into interpretable sub-components. The encoder is responsible for decomposing the time series data into latent seasonal and growth representations, while the decoder combines these components to produce the final forecast.

Exponential Smoothing Attention replaces the traditional dot-product attention mechanism with an attention function that emphasizes recent observations, which is similar to the exponential smoothing method commonly used in traditional forecasting models. This approach enhances the model's ability to predict trends over time. Frequency Attention, on the other hand, uses Fourier transformation to identify and extract dominating seasonal patterns, which allows the model to effectively capture recurring behaviors. The combination of Exponential Smoothing and Frequency Attention ensures that the model not only achieves a state-of-the-art performance in terms of forecasting accuracy, but also maintains interpretability by explicitly modeling level, growth, and seasonal components. The ETSformer was empirically evaluated on multiple benchmark datasets and showed significant improvements over existing Transformer-based approaches for time series forecasting.

NSTransformer, which was introduced by Liu et al. [49], was designed to overcome the limitations of traditional Transformers in handling non-stationary time series data. This model combines two core components (which together enable an effective modeling of non-stationary real-world data): Series Stationarization and De-stationary Attention. The NSTransformer architecture follows a standard Transformer encoder–decoder structure but introduces innovations specifically for handling non-stationary data. The Series Stationarization module applies a normalization technique that unifies key statistics (mean and variance) of each input time series, thereby stabilizing the input distribution for better generalization. This module acts as a preprocessing step that makes non-stationary inputs more tractable for the Transformer.

However, to address the problem of over-stationarization (where stationarization causes loss of valuable temporal characteristics of the original data), NSTransformer also includes a De-stationary Attention mechanism. This attention mechanism restores the original non-stationary properties that were lost during stationarization. Through using a learned de-stationary factor, this mechanism approximates the attention that would have been obtained from raw non-stationary data, ensuring that the model retains the distinct temporal dependencies necessary for accurate forecasting. The NSTransformer also incorporates a two-stage transformation process (which transforms predictions back to the original scale): normalization before feeding data to the model and de-normalization after generating outputs. These features make the NSTransformer suitable for effectively leveraging non-stationary information while maintaining the computational efficiency and long-term dependency capabilities of standard Transformer-based models.

Kitaev et al. [50] introduced the Reformer, an efficient Transformer architecture specifically designed to handle long sequences with reduced computational and memory requirements. The Reformer makes significant architectural changes to the original Transformer by incorporating Locality-Sensitive Hashing (LSH) Attention and Reversible Residual Layers. The LSH Attention mechanism is a key innovation of the Reformer. Traditional Transformers use scaled dot-product attention, which has a computational complexity

of $\mathcal{O}(L^2)$, where L is the sequence length, making it infeasible for long input sequences. Reformer replaces this with Locality-Sensitive Hashing to approximate attention, reducing the complexity to $\mathcal{O}(L\log(L))$. In this approach, the keys and queries are hashed into buckets such that only similar elements are grouped together for attention calculations. This significantly reduces the number of dot products computed while still capturing the most important relationships between elements, making it possible to efficiently handle long sequences. The second major modification is the use of Reversible Residual Layers. In standard Transformer architectures, each layer requires storing intermediate activations for backpropagation, which scales linearly with the number of layers, creating a large memory burden. Reformer addresses this by employing reversible residual connections, inspired by RevNets, which allow activations from previous layers to be reconstructed during backpropagation rather than stored. This approach effectively eliminates the need for storing layer-wise activations, significantly reducing memory usage during training.

Additionally, chunking is applied to feed-forward layers to further manage memory usage. The feed-forward layers, typically responsible for large intermediate activations, are processed in smaller chunks, reducing the peak memory requirement without affecting the model's performance. This enables Reformer to efficiently handle feed-forward computations for long sequences. The combination of LSH Attention, Reversible Residual Layers, and chunked feed-forward processing allows the Reformer to maintain the expressive power of the original Transformer architecture while being significantly more efficient in both memory and computation. The Reformer is particularly suitable for tasks involving long sequences, such as language modeling and time-series forecasting, where traditional Transformers face scalability issues.

3.2. Probabilistic Forecasting of Time Series Data

Let $\mathcal{D} = \{\boldsymbol{y}^i_{1:T^i}\}_{i=1}^N$ represent a dataset consisting of N univariate time series, where each uniformly spaced time series $\boldsymbol{y}^i_{1:T^i} = \left(y^i_1, \ldots, y^i_{T^i}\right)$ contains T^i observations and y^i_t denotes the value of the i-th time series at time t [41]. For example, y^i_t might indicate the number of units sold of product i on day t. To simplify the notation, it will be assumed that all time series have the same length T, even though the models can handle time series of varying lengths. The goal of time series forecasting is to predict the next H values for each time series in $\mathcal{D}$, where H is referred to as the prediction length or forecast horizon. Additionally, each time series $\boldsymbol{y}^i_{1:T}$ may have associated covariates $\boldsymbol{X}^i_{1:T+H}$, which can include both static and time-varying features. Static covariates are attributes that remain constant across time, such as store location or product ID. Time-varying covariates change over time and could include factors like the day of the month or planned promotions.

The problem in probabilistic time series forecasting can be formally described as modeling the joint conditional distribution of the future time series values $\boldsymbol{y}^i_{T+1:T+H}$, given its historical observations $\boldsymbol{y}^i_{1:T}$ and any associated covariates $\boldsymbol{X}^i_{1:T+H}$. This is represented as follows [34]:

$$p_\phi\left(\boldsymbol{y}^i_{T+1:T+H} \mid \boldsymbol{y}^i_{1:T}, \boldsymbol{X}^i_{1:T+H}\right), \tag{3}$$

where ϕ denotes the parameters of the parametric distribution being modeled. Thus, the objective of probabilistic time series forecasting is not just to provide a single point prediction but to estimate the full conditional distribution, capturing the inherent uncertainty in the future values [53]. This allows for more robust decision making in applications where the range of possible outcomes and their probabilities are as critical as the predictions themselves.

In practice, instead of using the entire history of each time series i, which can vary significantly, a focus is placed on extracting fixed context windows of size $C \geq 1$ [54].

This approach involves sampling subsequences from the full time series, allowing us to estimate the conditional distribution of the next H future values based on the selected context window and the corresponding covariates. This conditional distribution can be expressed as follows:

$$p_\phi\left(\boldsymbol{y}^i_{C+1:C+H} \mid \boldsymbol{y}^i_{1:C}, \boldsymbol{X}^i_{1:C+H}\right). \tag{4}$$

It is worth noting that the initial time step of the context window does not necessarily align with the beginning of the time series. When a neural network with weights θ is used to model this distribution, predictions are conditioned on these learned parameters. To estimate the conditional distribution described above, inspired by Rasul et al. [36,37], an autoregressive approach can be applied, leveraging the chain rule of probability as follows:

$$p_\phi\left(\boldsymbol{y}^i_{C+1:C+H} \mid \boldsymbol{y}^i_{1:C}, \boldsymbol{X}^i_{1:C+H}; \theta\right) = \prod_{t=C+1}^{C+H} p_\phi\left(\boldsymbol{y}^i_t \mid \boldsymbol{y}^i_{1:t-1}, \boldsymbol{X}^i_{1:t}; \theta\right). \tag{5}$$

The tokenization process employed in this study involves creating lagged features based on past values of the time series, which are tailored to align with the data's frequency [54]. Following the recommendations of Alexandrov et al. [55], appropriate lag values were selected for various frequencies, including quarterly, monthly, weekly, daily, and hourly. For a given frequency, a sorted set of positive lag indices, $\mathbb{L} = \{1, \ldots, \mathcal{L}\}$, is defined, where $\mathcal{L}$ represents the largest lag index in the set. These lag indices are generally not evenly spaced in time. Lag features are then generated for each context window $\boldsymbol{y}^i_{1:C}$. This process involves sampling from an extended window containing $\mathcal{L}$ additional historical points, which are denoted as $\boldsymbol{y}^i_{-\mathcal{L}:C}$ [54]. If a total of K static and dynamic covariates are added to these lagged features, the resulting token for each time series value will have a size of $|\mathbb{L}| + K$. Figure 2 illustrates this tokenization process.

As shown in Figure 2, the architecture of the probabilistic Transformer-based models employed in this study consists of two main components: an encoder and a decoder. For the encoder, a sequence of C tokens is generated by tokenizing the data through the concatenation of covariates $\boldsymbol{X}^i_{1:C}$ with lagged features sampled from the extended window $\boldsymbol{y}^i_{-\mathcal{L}:C}$. Similarly, for the decoder, the sequence of H tokens is created by concatenating covariates $\boldsymbol{X}^i_{C+1:C+H}$ with lagged features sampled from the extended window $\boldsymbol{y}^i_{-\mathcal{L}+C:C+H}$. Both encoder and decoder tokens are used during training. These tokens are then passed through a shared linear projection layer, which maps the features into the hidden dimension of the attention mechanism. To encode the position of each token in the sequence, positional encoding, as outlined by Vaswani et al. [45], is applied. This encoding uses a combination of sine and cosine functions at different frequencies, which are added to the token embeddings. By incorporating information about both relative and absolute positions, positional encoding allows the attention mechanism to effectively capture the sequential order of tokens, a critical aspect for time series modeling.

After processing data through the masked decoder layers, the model predicts the parameters ϕ for the forecast distribution of the next time step. These parameters are computed by a parametric distribution head, which serves as the model's final layer. The distribution head projects the features learned by the model to the parameters of the selected probability distribution [54]. Various parametric distributions can be employed; in this study, the Student's t-distribution was utilized, which outputs three parameters: mean μ, scale σ, and degrees of freedom ν. Training is performed by minimizing the negative log-likelihood of the forecasted distribution across all predicted time steps.

During inference, for a time series containing at least $\mathcal{L}$ observations, a feature vector is tokenized and passed into the model to estimate the distribution of the subsequent time step.

Using greedy autoregressive decoding, the model can simulate multiple future trajectories up to the defined forecast horizon $H \geq 1$. These simulations enable the computation of uncertainty intervals, which are critical for decision making and for assessing the model's accuracy on unseen data.

This methodology to probabilistic time series forecasting has been applied to the previously described Transformer-based architectures, including the Vanilla Transformer [45], Informer [46], Autoformer [47], ETSformer [48], NSTransformer [49], and Reformer [50]. The implementation builds upon the tools and frameworks introduced by Kashif Rasul [51,56].

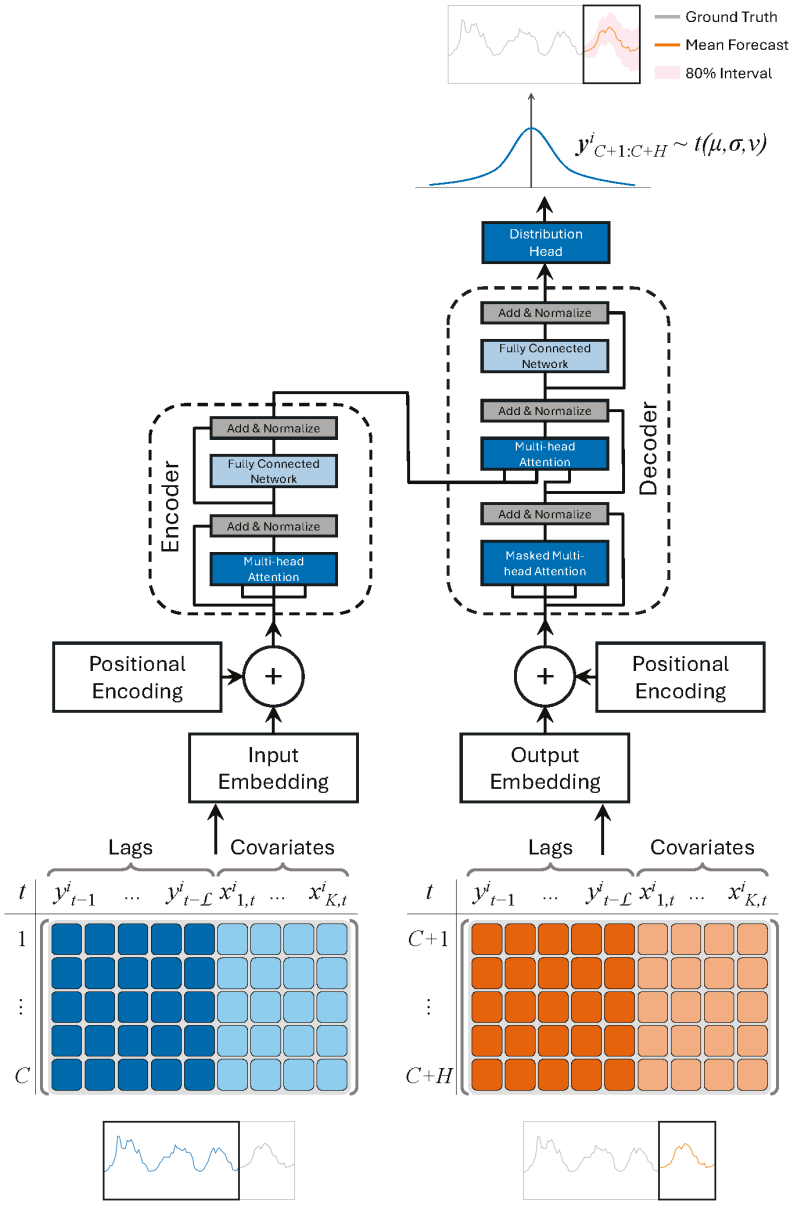


Figure 2. Architecture of the Transformer-based models for probabilistic time series forecasting and their tokenization process.

4. Empirical Evaluation

4.1. Dataset

The M5 dataset represents a significant advancement in the realm of retail forecasting by leveraging a publicly available dataset to facilitate transparent, reproducible, and rig-

orous evaluation of forecasting methodologies [57,58]. Publicly accessible datasets like the M5 are crucial for advancing the field as they enable researchers and practitioners to benchmark methods, validate results, and push the boundaries of innovation. This open access fosters collaboration, promotes replication of results, and provides a shared foundation for addressing complex forecasting challenges.

The M5 dataset, generously provided by Walmart, consists of 3049 individual product time series of daily unit sales data, spanning approximately 5.4 years from 29 January 2011 through to 19 June 2016, resulting in a total of 1969 daily data points. The dataset includes products from three categories—hobbies, foods, and household—sold across 10 stores located in three U.S. states: California, Texas, and Wisconsin. This hierarchical structure enables evaluations at multiple aggregation levels, ranging from total sales across all stores to individual product sales at specific locations, thereby reflecting the intricate, hierarchical, and multivariate nature of retail forecasting. The dataset's comprehensive design ensures representation of diverse shopping behaviors, regional market dynamics, and product-specific trends, making it a robust resource for developing, benchmarking, and testing advanced forecasting models.

In this work, data from three stores—one from each state—were analyzed, resulting in a total of 9147 distinct time series. This selection was made to accommodate limited computational resources for training the models while enabling a focused analysis that still represents the diversity and complexity of the dataset, capturing variations across regions, product categories, and store-level dynamics.

The framework of the M5 competition was adopted, with the final 28 days of each time series (from 23 May 2016 to 19 June 2016) reserved as the test set for out-of-sample evaluation. The earlier data, covering up from 29 January 2011 through to 22 May 2016, was used to train the models.

4.2. Explanatory Variables

The M5 dataset includes several explanatory variables that enhance its utility for improving the accuracy of forecasting models in retail settings. These variables supplement the core sales data and enable the modeling of external factors influencing demand. The key exogenous variables in the M5 dataset are as follows:

- Calendar-Related Information: This includes a wide range of time-related variables such as the date, week day, week number, month, and year. Additionally, it includes indicators for special days and holidays (e.g., the Super Bowl, Valentine's Day, and Orthodox Easter), which are categorized into four classes: sporting, cultural, national, and religious. Special days account for about 8% of the dataset, with their distribution across the classes being 11% sporting, 23% cultural, 32% national, and 34% religious.
- Selling Prices: Prices are provided at a weekly level for each store. The weekly average prices reflect consistent pricing across the seven days of a week. If a price is unavailable for a given week, it indicates that the product was not sold during that period. Over time, the selling prices may vary, offering critical information for understanding price elasticity and its impact on sales.
- SNAP Activities: The dataset includes a binary indicator for Supplemental Nutrition Assistance Program (SNAP) activities. These activities denote whether a store allowed purchases using SNAP benefits on a particular date. This variable accounts for about 33% of the days in the dataset and reflects the socio-economic factors affecting consumer purchasing behavior.

These variables are instrumental in enriching the dataset's predictive power by providing critical contextual information. Calendar-related variables capture temporal effects such

as seasonality and special events, helping models identify recurring patterns in consumer behavior. Price and promotional data offer valuable insights into how market conditions influence purchasing decisions, improving the model's ability to forecast demand fluctuations. Additionally, socio-economic factors are well represented through the inclusion of SNAP activities. The SNAP indicators reflect variations in demand driven by government assistance programs, which can significantly influence consumer spending behavior and sales dynamics. This is particularly relevant in economically vulnerable regions, where such programs play a key role in shaping purchasing patterns. By incorporating these diverse exogenous variables, the M5 dataset provides a robust foundation for developing sophisticated forecasting models that can effectively address the complexities of retail sales.

Table 1 presents a comprehensive summary of the input features (lags and covariates) used in the time series forecasting models. The table highlights the diversity of features extracted from the dataset to improve the models' predictive accuracy.

Table 1. The lags and explanatory variables used in the forecasting models, including time-related features, price data, SNAP activities, event indicators, and hierarchical identifiers.

Data	No. of Variables	Feature	Type	No. of Categories	Encoding
Sales	30	Lags: {1, 2, 3, 4, 5, 6, 7, 8, 13, 14, 15, 20, 21, 22, 27, 28, 29, 30, 31, 56, 84, 363, 364, 365, 727, 728, 729, 1091, 1092, 1093}	Continuous	—	—

Time	9	`Day of week`
		`Day of month`
		`Day of year`
		`Month of year`
		`Week of year`
		`Week of month`
		`Year`
			Categorical	7
				31
				366
				12
				53
				6
				6
					Encoded as zero-based index and normalized to $[-0.5, 0.5]$

		`Is weekend`		2	Boolean
		`Age`	Continuous	—	$\log_{10}(2 + \texttt{n_sale_days})$
Price	3	Item's daily price normalized by mean/std Item's daily price normalized by department's daily mean/std price Item's daily price normalized by store's daily mean/std price	Continuous	—	—
Snap	3	Supplemental nutrition assistance program days in CA, TX, WI	Categorical	3	Boolean
Events	2	Event name: {`nan`,`ChanukahEnd`,`Christmas`, `CincoDeMayo`,`ColumbusDay`,`Easter`,`EidAl-Fitr`, `EidAlAdha`,`Father'sDay`,`Halloween`,`IndependenceDay`, `LaborDay`,`LentStart`,`LentWeek2`,`MartinLutherKingDay`, `MemorialDay`,`Mother'sDay`,`NBAFinalsEnd`,`NBAFinalsStart`, `NewYear`,`OrthodoxChristmas`,`OrthodoxEaster`,`PesachEnd`, `PresidentsDay`,`PurimEnd`,`RamadanStart`,`StPatricksDay`, `SuperBowl`,`Thanksgiving`,`ValentinesDay`,`VeteransDay`}	Categorical	31	Encoded as zero-based index and normalized to $[-0.5, 0.5]$
	2	Event type: {`nan`,`Cultural`,`National`,`Religious`,`Sporting`}	Categorical	5	

ID	60	`item_id`
		`dept_id`
		`cat_id`
		`store_id`
		`state_id`
			Categorical	3049
				7
				3
				3
				3
					Encoded as zero-based index and embedded using a learnable embedding layer with an embedding dimension of min(50,(`n_categ`+1)//2)

The Sales feature includes 30 specific lag values that represent the historical sales observations used as inputs for forecasting. These lag values cover different time intervals to capture both short-term and long-term patterns, including daily, weekly, and annual cycles,

ensuring the models have a broad temporal context. Several time-related features were included as categorical variables, such as `Day of week`, `Day of month`, `Day of year`, `Month of year`, `Week of year`, `Week of month`, and `Year`. These categorical time features, encoded as a zero-based index and normalized to a range of $[-0.5, 0.5]$, help the models account for seasonality and calendar effects. The `Is weekend` feature is a binary indicator used to identify weekends, which can impact sales patterns due to changes in consumer behavior. Another continuous feature included is `Age`, which represents the age of the product in the dataset. This is calculated as a logarithmic transformation of the number of sale days and helps capture the effect of product lifecycle on sales. The table also includes three price-related features, representing the daily price of items normalized by different factors: the mean and standard deviation of item prices, department prices, and store prices. These features capture the impact of price changes on sales. SNAP activities are included as a categorical feature indicating whether purchases were allowed using the Supplemental Nutrition Assistance Program (SNAP) benefits in the states of California, Texas, and Wisconsin. This variable captures socio-economic factors that influence consumer demand. The Events feature accounts for 31 distinct special days, such as holidays and other significant events, categorized into four classes: sporting, cultural, national, and religious. Including these variables helps the models account for spikes or drops in sales associated with specific events. Additionally, ID features are included to capture hierarchical information from the dataset. These IDs include item IDs, department IDs, category IDs, store IDs, and state IDs, each of which are encoded as categorical variables. The IDs are embedded using a learnable embedding layer to help the model understand relationships across different levels of the hierarchy, such as items within a department or stores within a state.

4.3. Hyperparameter Tuning

Selecting a model that performs consistently well in out-of-sample predictions is a crucial step in the modeling process. To achieve this, it is common practice to use a validation set for distinguishing between competing models. Considering that deep learning models can be sensitive to hyperparameter settings and initialization, an effective strategy for model selection becomes essential. In this study, the final 28 days of the training period, from 25 April to 22 May 2016, were designated as a validation set to objectively compare and rank different model configurations.

To explore the hyperparameter space systematically and to identify optimal settings, this study employed the Optuna framework [59]: an advanced tool for hyperparameter optimization. Optuna is an open-source Python library that was designed to streamline the process of hyperparameter tuning, particularly for deep learning models, including those based on Transformers. The framework offers dynamic search space construction through its define-by-run API, and it supports efficient search strategies like Tree-structured Parzen Estimator (TPE), Random Search, and Grid Search. Additionally, Optuna includes pruning techniques to optimize computational resources and integrates seamlessly with popular deep learning frameworks such as PyTorch. The optimization process in Optuna involves defining an objective function, conducting a study, running the optimization trials, and analyzing the resulting configurations. This approach simplifies the time-consuming task of tuning hyperparameters, allowing researchers to focus more on refining their models and interpreting results. By employing this robust optimization tool, this study ensured that the selected hyperparameter configurations enhanced the performance and reliability of the forecasting models.

Table 2 outlines the hyperparameter search spaces explored in this study using the Optuna Hyperparameter Optimization (HPO) framework. Optuna was employed to randomly sample values from these predefined spaces, generating a variety of model configurations. The configuration that achieved the highest validation score, based on the Mean Weighted

Quantile Loss (MWQL), was selected as the optimal model. MWQL, a metric specifically designed for evaluating probabilistic forecasts, which approximates (a weighted average of) the Continuous Ranked Probability Score (CRPS) [60], provides a comprehensive assessment of accuracy across multiple quantile levels, making it an effective criterion for ranking model performance, as defined in Equation (8). The table details the ranges of key hyperparameters, including context length, batch size, and the number of encoder and decoder layers used across the evaluated Transformer-based models. The context length, ranging from 28 days to multiples of 28-day periods, defines the historical time window used in training, with different lengths aiming to capture a range of temporal patterns from short-term fluctuations to more extended seasonal trends. The batch size parameter varies between 32 and 256, impacting the number of data samples processed in one iteration, thus affecting training stability and efficiency. Furthermore, the number of encoder and decoder layers represents the model's depth. Deeper models, such as those with up to 16 encoder layers, are capable of learning more complex patterns, but they also require greater computational resources. The exploration of these hyperparameter spaces allowed this study to fine tune each model, ensuring robust performance across different sales data patterns.

Table 2. The model's hyperparameter search spaces used in HPO.

Hyperparameter	**Range**	
	Transformer, Autoformer, Informer NSTransformer, Reformer	**ETSformer**
Context length	$\{28, 28 \times 2, 28 \times 3\}$	
Batch size	$\{32, 64, 128, 256\}$	
Number of encoder layers	$\{2, 4, 8, 16\}$	—
Number of decoder layers	$\{2, 4, 8, 16\}$	—

Table 3 summarizes the settings used in the hyperparameter tuning process with the Optuna framework to determine the optimal configurations for each model. It presents key parameters, including the number of trials, epochs, batches per epoch, samples processed in each optimization trial, and validation function. The number of trials refers to the total number of model configurations evaluated during the optimization process. Each trial represents a unique combination of hyperparameter values that were sampled from the predefined search spaces. In this study, 10 trials were conducted for each model to explore diverse configurations. The number of epochs indicates how many times the entire training dataset was passed through the model during each trial, ensuring sufficient iterations for parameter adjustment. The number of batches per epoch specifies how many batches of data were processed in each epoch. The hyperparameter tuning process involved sampling 20 values per trial, which were subsequently used to calculate the MWQL metric, as described in Section 4.4. These samples were drawn to ensure robust point and probabilistic predictions across different forecast horizons.

Table 3. The hyperparameter settings applied during the tuning process with the Optuna framework.

Parameter	**Value**
Number of trials	10
Number of epochs	10
Number of batches per epoch	50
Number of samples	20
Validation function	Mean Weighted Quantile Loss (MWQL)

Table 4 presents the parameter-specific configurations applied to each of the Transformer-based models considered in this study. This table highlights the variations across critical settings, including prediction length, distribution output, loss function, and dimensionality of key components such as layers and attention mechanisms. The prediction length was consistently set to 28 days across all models, ensuring a standardized forecast horizon. The distribution output employed was Student's t-distribution, which accounts for the potential variability in sales data. The loss function used in all models was the Negative Log-Likelihood, a suitable choice for probabilistic forecasting, focusing on minimizing prediction uncertainty. The learning rate for all Transformer-based models was set to 10^{-3}, ensuring a stable and efficient convergence during the training process. This rate was chosen to balance the need for sufficient parameter updates while avoiding overshooting the optimal solution. The scaling of the input target varied among models. For instance, while most of the models use a standardized approach with mean and standard deviation normalization, some models, such as NSTransformer, do not apply scaling. This difference in scaling techniques reflects the distinct architecture and assumptions underlying each model. The lags sequence parameter, which provides historical context for the models, was consistent across the models with a predefined set of lag values to capture short-term and seasonal trends. In terms of the dimensionality of Transformer layers, the parameter settings varied, with models like ETSformer and Informer employing larger layer sizes (up to 64) to capture more complex patterns in the time series data. The number of attention heads and the feed-forward hidden size varied depending on the model architecture. For example, Informer utilizes specialized attention mechanisms, such as ProbAttention, to enhance computational efficiency, whereas other models rely on standard multi-head attention. Specific models, such as Autoformer and Informer, include unique settings like moving average windows and autocorrelation factors to enhance performance in handling seasonality and periodic patterns. These specialized configurations reflect the targeted design of each model to address distinct challenges in time series forecasting.

Table 5 presents the optimal hyperparameter configurations identified for each of the Transformer-based models used in this study, as determined through the Optuna optimization framework. These configurations were fine tuned to maximize forecasting accuracy based on the Mean Weighted Quantile Loss (MWQL) metric. The context length values varied across models, reflecting the different time horizons used to capture historical patterns in the time series data. For example, some models perform better with shorter context windows (e.g., 28 days), while others benefit from longer windows (e.g., 28×3 days), in accounting for more extended seasonal trends. The batch size also differed among the models, indicating the number of data samples processed in each iteration during training. Smaller batch sizes can improve model stability, while larger batch sizes enhance computational efficiency. The batch size selection balances the trade-off between training speed and prediction accuracy. The number of encoder and decoder layers influences the model's capacity to learn complex temporal patterns. Deeper models, with more layers, tend to capture more intricate dependencies, but at the cost of increased computational requirements. The Best MWQL value represents the minimum Mean Weighted Quantile Loss achieved during the hyperparameter tuning process for each model. This value is crucial as it indicates the model's effectiveness in providing accurate probabilistic forecasts across multiple quantiles. Comparing the Best MWQL values obtained with and without explanatory features highlights the importance of feature integration in improving forecast accuracy. Models that incorporate explanatory features consistently achieve lower MWQL values, demonstrating the impact of additional context in refining prediction intervals and better capturing the inherent uncertainties of retail demand. Lower MWQL values reflect

better model performance, demonstrating the model's ability to produce reliable prediction intervals that capture uncertainty effectively.

Table 4. The parameter-specific configurations applied to each of the Transformer-based models.

Parameter	Transformer	Autoformer	ETSformer	Informer	NSTransformer	Reformer
Prediction length of decoder	28					
Distribution output	Student's t					
Loss function	Negative log likelyhood					
Learning rate	10^{-3}					
Size of target	1					
Scale of the input target	mean	std	std	std	—	std
Lags sequence	[1, 2, 3, 4, 5, 6, 7, 8, 13, 14, 15, 20, 21, 22, 27, 28, 29, 30, 31, 56, 84, 363, 364, 365, 727, 728, 729, 1091, 1092, 1093]					
Dimensionality of Transformer layers	32	—	64	64	—	64
Number of attention heads	2					
Feed-forward hidden size	32	32	—	32	32	—
Activation function	gelu	relu	—	relu	gelu	—
Dropout for fully connected layers	0.1					
Moving average window	—	25	—	—	—	—
Autocorrelation factor	—	1	—	—	—	—
Number of layers	—	—	2	—	—	—
K largest amplitudes	—	—	4	—	—	—
Embedding kernel size	—	—	3	—	—	—
Attention in encoder	—	—	—	ProbAttention	—	—
Use distilling in encoder	—	—	—	True	—	—
ProbSparse sampling factor	—	—	—	5	—	—

Table 5. The optimal hyperparameter configurations obtained through the Optuna framework for each Transformer-based model.

Hyperparameter	Transformer	Autoformer	ETSformer	Informer	NSTransformer	Reformer
			Without features			
Context length	28	28×2	28	28×3	28×3	28×2
Batch size	128	64	128	256	256	64
Number of encoder layers	16	16	—	16	4	4
Number of decoder layers	2	8	—	16	8	4
Best MWQL value	0.6121	0.7403	0.6178	0.7753	0.6081	1.5830
			With features			
Context length	28	28×3	28	28×2	28×3	28
Batch size	256	256	128	128	256	32
Number of encoder layers	8	4	—	2	4	16
Number of decoder layers	4	4	—	2	4	2
Best MWQL value	0.6067	0.7387	0.6312	0.7730	0.6228	1.3990

4.4. Performance Metrics

To evaluate the performance of the forecasting models on the test set, a set of widely recognized accuracy metrics was employed. The test set consisted of the final 28 days of each time series (from 23 May 2016, to 19 June 2016), and it was reserved for out-of-sample evaluation following the framework of the M5 competition. These metrics provided

insights into the accuracy and reliability of the forecasts by comparing the predicted values against the observed data in this holdout period. The model predictions were generated by autoregressively sampling the future time steps from the conditioned context window. For each time step in the prediction horizon, 20 samples were drawn ensuring robust point and probabilistic predictions across the test set.

To evaluate the point forecasts, the Mean Absolute Scaled Error (MASE) and the Normalized Root Mean Squared Error (NRMSE) were used, which were calculated as follows: MASE is a scale-independent metric that evaluates the accuracy of forecasts by comparing them to a naïve baseline model. For a dataset consisting of N time series, it is defined as follows [61]:

$$\text{MASE} = \frac{1}{N}\sum_{i=1}^{N}\frac{\frac{1}{H}\sum_{t=T+1}^{T+H}|\tilde{y}_t^i - y_t^i|}{\frac{1}{T-1}\sum_{t=2}^{T}|y_t^i - y_{t-1}^i|}, \tag{6}$$

where y_t^i denotes the value of the i-th time series at time t, $\tilde{y}_t^i$ denotes the median of the samples, H is the prediction length or forecast horizon, and T is the length of the time series i in the training set. To simplify the notation, it is assumed that all of the time series had the same length T.

A lower MASE value indicates better model performance. Specifically, a value less than 1 indicates that the forecasting model performs better than the naïve baseline, while values greater than 1 indicate worse performance. MASE is particularly effective for comparing models across different time series as it is invariant to scaling.

NRMSE normalizes the RMSE (Root Mean Squared Error) by dividing it by the mean of the observed values in the test set. For a dataset consisting of N time series, it is defined as follows:

$$\text{NRMSE} = \frac{1}{\frac{1}{N}\sum_{i=1}^{N}\frac{1}{H}\sum_{t=T+1}^{T+H}|y_t^i|}\sqrt{\frac{1}{N}\sum_{i=1}^{N}\frac{1}{H}\sum_{t=T+1}^{T+H}\left(\hat{y}_t^i - y_t^i\right)^2}, \tag{7}$$

where y_t^i denotes the value of the i-th time series at time t, $\hat{y}_t^i$ denotes the mean of the samples, H is the prediction length or forecast horizon, and T is the length of the time series i in the training set. Lower NRMSE values indicate better model performance. NRMSE is sensitive to large deviations, meaning it assigns greater penalties to larger errors. It is particularly useful for understanding the spread of the prediction errors in the test set.

In addition to point forecast metrics, this study assessed probabilistic forecasts using the Mean Weighted Quantile Loss (MWQL) and Mean Absolute Error Coverage (MAE Coverage). These metrics evaluate the model's ability to capture uncertainty and provide reliable prediction intervals.

MWQL assesses the quality of probabilistic forecasts by evaluating how well a model predicts various quantiles of the future distribution. For a set of $\mathcal{Q}$ quantiles $\{q_1, ..., q_\mathcal{Q}\}$, it is defined as follows [60,62]:

$$\text{MWQL} = \frac{1}{\mathcal{Q}}\sum_{j=1}^{\mathcal{Q}}\text{WQL}_{q_j}, \tag{8}$$

where $\mathcal{Q}$ is the number of quantiles and WQL_{q_j} is the Weighted Quantile Loss of quantile q_j, which is defined for a dataset of N time series as

$$\text{WQL}_{q_j} = \frac{1}{\sum_{i=1}^{N}\sum_{t=T+1}^{T+H}|y_t^i|}\sum_{i=1}^{N}\sum_{t=T+1}^{T+H}\rho_{i,q_j}\left(y_t^i, f_t^{i,q_j}\right), \tag{9}$$

where T is the length of the time series i in the training set, H is the prediction length or forecast horizon, y_t^i is the value of the i-th time series at time t, f_t^{i,q_j} is the predicted quantile q_j of time series i at time t, and $\rho_{i,q_j}\left(y_t^i, f_t^{i,q_j}\right)$ is the quantile loss at level q_j, which is defined as

$$\rho_{i,q_j}\left(y_t^i, f_t^{i,q_j}\right) = \begin{cases} 2(1-q_j)\left(f_t^{i,q_j} - y_t^i\right) & \text{, if } y_t^i < f_t^{i,q_j} \\ 2q_j\left(y_t^i - f_t^{i,q_j}\right) & \text{, if } y_t^i \geq f_t^{i,q_j} \end{cases}. \tag{10}$$

Lower MWQL values indicate better performance as they reflect the model's ability to provide accurate predictions across different quantiles. This metric is crucial for applications that require understanding uncertainty, such as inventory management. In all of the experiments, quantiles $\{0.1, 0.2, \ldots, 0.9\}$ were used to calculate MWQL, resulting in $\mathcal{Q} = 9$.

MAE Coverage quantifies the proportion of time points where the actual value lies below the predicted quantile. For a set of $\mathcal{Q}$ quantiles $\{q_1, ..., q_\mathcal{Q}\}$, it is defined as follows:

$$\text{MAE Coverage} = \frac{1}{\mathcal{Q}} \sum_{j=1}^{\mathcal{Q}} \left| \text{Coverage}_{q_j} - q_j \right|, \tag{11}$$

where $\mathcal{Q}$ is the number of quantiles and Coverage_{q_j} is the coverage of quantile q_j, which defined for a dataset of N time series as

$$\text{Coverage}_{q_j} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{H} \sum_{t=T+1}^{T+H} \tau_{i,q_j}\left(y_t^i, f_t^{i,q_j}\right),$$

where T is the length of the time series i in the training set, H is the prediction length or forecast horizon, y_t^i is the value of the i-th time series at time t, f_t^{i,q_j} is the predicted quantile q_j of time series i at time t, and $\tau_{i,q_j}\left(y_t^i, f_t^{i,q_j}\right)$ is defined as

$$\tau_{i,q_j}\left(y_t^i, f_t^{i,q_j}\right) = \begin{cases} 1 & \text{, if } y_t^i \leq f_t^{i,q_j} \\ 0 & \text{, if } y_t^i > f_t^{i,q_j} \end{cases}. \tag{12}$$

Higher MAE Coverage values indicate better coverage and reliability of probabilistic forecasts, meaning the prediction intervals are well calibrated to the observed data.

By employing these performance metrics, this study provided a comprehensive evaluation of both point and probabilistic forecasts, ensuring the robustness and reliability of the Transformer-based models for retail demand forecasting. Emphasis was placed on achieving lower values for error-based metrics (MASE, NRMSE, and MWQL) and higher values for coverage-related metrics (MAE Coverage) to indicate better predictive performance and model reliability.

4.5. Results and Discussion

The empirical evaluation results presented in Tables 6–8 highlight the performance differences between the various Transformer-based forecasting models employed in this study. These results encompass both point forecast and probabilistic forecast metrics, providing a comprehensive comparison of the models' accuracy and reliability over the full forecast horizon, as well as at different forecast steps. The use of traditional benchmark models, such as ARIMA and ETS, proved impractical in this study due to their excessive compu-

tational demands when incorporating multiple regressors, rendering them unsuitable for large-scale retail demand forecasting applications.

Table 6 summarizes the overall performance of the models across point forecast metrics, such as Mean Absolute Scaled Error (MASE) and Normalized Root Mean Square Error (NRMSE), and probabilistic forecast metrics, including Weighted Quantile Loss (WQL) at different quantile levels, Mean Weighted Quantile Loss (MWQL), and Mean Absolute Error (MAE) Coverage. The results demonstrate that the inclusion of additional explanatory features consistently improves the performance of most models. For instance, the Transformer model with explanatory features achieved a lower NRMSE of 1.650 compared to 1.748 without features, indicating better accuracy in point forecasts. Similarly, the MAE Coverage for the Transformer model increased from 0.081 to 0.190 with the inclusion of features, suggesting improved reliability in probabilistic forecasts.

Table 6. Overall performance comparison of the Transformer-based models across point and probabilistic forecast metrics.

		Point Forecast Metrics			**Probabilistic Forecast Metrics**			
Model		**MASE**	**NRMSE**	**WQL0.1**	**WQL0.5**	**WQL0.9**	**MWQL**	**MAE Coverage**
Transformer	Without features	**0.902**	1.748	0.254	**0.735**	**0.675**	**0.629**	0.081
	With features	0.906	**1.650**	**0.227**	0.738	0.716	0.634	**0.190**
Autoformer	Without features	1.062	**2.421**	**0.378**	**0.859**	**0.701**	**0.741**	0.070
	With features	**1.054**	2.439	0.395	0.925	0.731	0.796	**0.071**
ETSformer	Without features	**0.985**	1.984	**0.316**	**0.769**	**0.595**	**0.646**	0.049
	With features	1.067	**1.738**	0.328	0.797	0.615	0.674	**0.127**
Informer	Without features	1.026	**2.470**	0.322	**0.921**	**0.935**	**0.801**	**0.109**
	With features	**0.996**	2.522	**0.253**	0.942	1.038	0.830	0.086
NSTransformer	Without features	**0.917**	**1.669**	**0.249**	**0.746**	**0.680**	**0.636**	0.073
	With features	0.979	1.849	0.263	0.785	0.781	0.687	**0.178**
Reformer	Without features	2.463	4.550	0.547	2.217	3.182	2.097	**0.485**
	With features	**1.979**	**3.892**	**0.468**	**1.768**	**2.329**	**1.642**	0.449

For each forecasting accuracy metric and model (both with and without features), the most accurate value is highlighted in bold.

Table 7 provides a detailed breakdown of the point forecasting performance across varying forecast horizons. The accuracy of all the models generally decreased as the forecast horizon extended, which was expected due to the increasing uncertainty over time. As is often the case in forecasting studies, error metrics do not always fully align, and different evaluation criteria may highlight different aspects of model performance. Regarding point forecast metrics, the inclusion of additional explanatory features was clearly beneficial for all the models according to NRMSE, with the exception of the NSTransformer model. Similarly, when considering MASE, the inclusion of explanatory features improves performance for Autoformer, Informer, and Reformer, although this benefit was not observed for Transformer, ETSformer, and NSTransformer.

Table 8 presents the results of probabilistic forecasting over different forecast horizons. The MWQL values, which measure the overall accuracy of probabilistic predictions, generally increase as the forecast horizon lengthens, reflecting the increasing difficulty of maintaining high forecast accuracy over longer periods. The inclusion of additional explanatory features is clearly beneficial for all models except Informer and Reformer

according to MAE Coverage. However, for these two models, the inclusion of explanatory features does provide improvements according to MWQL, indicating that the benefits may be more evident when evaluating predictive uncertainty rather than direct coverage.

The results across Tables 6–8 consistently demonstrate the impact of incorporating explanatory features, such as calendar information, selling prices, and SNAP activity indicators, on model performance. These features enhance both point and probabilistic forecast accuracy by providing additional context that helps the models capture underlying patterns in the data more effectively. For four out of the six models (Transformer, Autoformer, Informer, and Reformer), the inclusion of explanatory features yielded improvements in three of the four error metrics. In the case of ETSformer and NSTransformer, at least one metric reflected a positive impact. While the benefits were not uniformly observed across all the models and evaluation criteria, the overall findings indicated that incorporating additional explanatory variables generally enhances forecasting accuracy and reliability.

Table 7. The point forecasting accuracy of the models across different forecast horizons.

		MASE				**NRMSE**			
Model		**1 Step**	**7 Steps**	**14 Steps**	**21 Steps**	**1 Step**	**7 Steps**	**14 Steps**	**21 Steps**
Transformer	Without features	**0.836**	**0.855**	**0.886**	**0.898**	1.488	1.639	1.659	1.704
	With features	0.837	0.856	0.888	0.903	**1.383**	**1.535**	**1.567**	**1.615**
Autoformer	Without features	1.027	1.043	1.065	1.067	2.408	2.530	2.469	2.452
	With features	**1.001**	**1.024**	**1.054**	**1.059**	**2.112**	**2.442**	**2.418**	**2.431**
ETSformer	Without features	**0.919**	**0.945**	**0.974**	**0.985**	1.859	2.129	1.986	1.987
	With features	0.988	1.021	1.053	1.066	**1.534**	**1.663**	**1.684**	**1.709**
Informer	Without features	0.981	1.004	1.023	1.029	2.227	2.487	**2.448**	**2.446**
	With features	**0.879**	**0.915**	**0.968**	**0.991**	**2.152**	**2.435**	2.473	2.496
NSTransformer	Without features	**0.857**	**0.872**	**0.902**	**0.916**	**1.402**	**1.543**	**1.589**	**1.634**
	With features	0.871	0.904	0.948	0.969	1.565	1.775	1.792	1.829
Reformer	Without features	1.621	1.873	2.151	2.334	2.908	3.521	3.943	4.277
	With features	**1.379**	**1.583**	**1.786**	**1.910**	**2.811**	**3.331**	**3.575**	**3.754**

For each forecasting accuracy metric and model (both with and without features), the most accurate value is highlighted in bold.

Table 8. The probabilistic forecasting accuracy of the models across different forecast horizons.

		MWQL				**MAE Coverage**			
Model		**1 Step**	**7 Steps**	**14 Steps**	**21 Steps**	**1 Step**	**7 Steps**	**14 Steps**	**21 Steps**
Transformer	Without features	0.594	0.588	**0.605**	**0.618**	0.056	0.061	0.070	0.072
	With features	**0.585**	**0.587**	**0.605**	0.622	**0.083**	**0.133**	**0.155**	**0.176**
Autoformer	Without features	**0.761**	**0.744**	**0.739**	**0.740**	0.051	0.060	0.064	0.067
	With features	0.790	0.787	0.789	0.794	**0.069**	**0.067**	**0.069**	**0.070**
ETSformer	Without features	**0.623**	**0.621**	**0.632**	**0.641**	0.046	0.046	0.048	0.049
	With features	0.645	0.646	0.658	0.668	**0.126**	**0.124**	**0.128**	**0.128**
Informer	Without features	0.783	0.796	0.794	**0.798**	**0.095**	**0.097**	**0.105**	**0.108**
	With features	**0.711**	**0.737**	**0.780**	0.814	0.059	0.065	0.076	0.083
NSTransformer	Without features	**0.595**	**0.594**	**0.613**	**0.627**	0.043	0.050	0.061	0.068
	With features	0.608	0.623	0.650	0.671	**0.058**	**0.096**	**0.137**	**0.161**
Reformer	Without features	1.283	1.489	1.753	1.949	**0.423**	**0.455**	**0.473**	**0.481**
	With features	**1.133**	**1.292**	**1.448**	**1.562**	0.339	0.388	0.420	0.438

For each forecasting accuracy metric and model (both with and without features), the most accurate value is highlighted in bold.

The ability to achieve both accurate point forecasts and reliable probabilistic forecasts was crucial for the practical applications in retail demand forecasting. Accurate point forecasts help in optimizing inventory levels and reducing stockouts, while reliable probabilistic forecasts enable better risk management by providing well-calibrated prediction intervals. The study's findings underscore the importance of integrating diverse explanatory variables to improve the robustness and reliability of forecasting models in dynamic retail settings.

Overall, the results highlight that Transformer-based models, when augmented with relevant explanatory features, provide a powerful tool for both short-term and long-term retail demand forecasting. The improvements observed across the different metrics and forecast horizons affirm the potential of these models to enhance decision-making processes in retail operations by providing more accurate and reliable forecasts.

5. Conclusions

This study investigated the role of Transformer-based deep learning models in probabilistic time series forecasting for retail demand prediction. By leveraging explanatory variables such as calendar events, pricing, and socio-economic indicators, this research aimed to assess how incorporating external covariates enhances forecasting accuracy. The results demonstrate that including these variables leads to improvements in predictive performance, enabling a more precise estimation of demand fluctuations.

The empirical findings reveal that models enriched with explanatory variables achieve up to a 12.4% reduction in Normalized Root Mean Squared Error (NRMSE) and a 2.9% improvement in Mean Absolute Scaled Error (MASE) compared to those relying solely on past sales. These enhancements validate the significance of integrating domain-specific information into forecasting models, particularly in environments where demand is influenced by external factors beyond historical trends. Additionally, probabilistic forecasting proves to be a valuable tool in retail operations, providing well-calibrated uncertainty estimates that support risk-aware decision making. This capability allows businesses to optimize inventory levels, reduce stockouts, and minimize excess stock, contributing to improved supply chain efficiency and financial stability.

This study also highlighted the importance of adaptability in forecasting techniques, as demand patterns vary across different product categories, seasons, and promotional cycles. The ability of deep learning models to generalize across multiple time series offers a scalable solution for large-scale retail applications. Furthermore, the findings emphasize the necessity of balancing computational efficiency with forecasting accuracy, ensuring that predictive models remain practical for real-world deployment.

While this research focused on retail forecasting, the methodology is applicable to other domains where demand estimation is influenced by external variables, such as finance, energy, and healthcare. Future research directions could explore the inclusion of additional contextual factors, such as real-time consumer sentiment or weather data, to further refine forecast accuracy. Additionally, the integration of explainability techniques would enhance model interpretability, allowing practitioners to gain deeper insights into the key drivers of demand variability.

Overall, this study underscores the effectiveness of Transformer-based models in forecasting retail demand and demonstrates the critical impact of incorporating explanatory variables. The results reinforce the need for data-driven approaches in predictive modeling, where leveraging external information can improve forecasting reliability and operational decision making in dynamic and uncertain environments.

Author Contributions: Conceptualization, R.C., J.M.O. and P.R.; methodology, R.C., J.M.O. and P.R.; software, R.C., J.M.O. and P.R.; validation, R.C., J.M.O. and P.R.; formal analysis, R.C., J.M.O. and P.R.; investigation, R.C., J.M.O. and P.R.; resources, R.C., J.M.O. and P.R.; data curation, R.C., J.M.O. and P.R.; writing—original draft preparation, R.C., J.M.O. and P.R.; writing—review and editing, R.C., J.M.O. and P.R.; visualization, R.C., J.M.O. and P.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: A publicly available dataset was used in this study. The data can be found here: https://www.kaggle.com/competitions/m5-forecasting-accuracy/data (accessed on 1 April 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Petropoulos, F.; Apiletti, D.; Assimakopoulos, V.; Babai, M.Z.; Barrow, D.K.; Ben Taieb, S.; Bergmeir, C.; Bessa, R.J.; Bijak, J.; Boylan, J.E.; et al. Forecasting: Theory and practice. *Int. J. Forecast.* **2022**, *38*, 705–871. [CrossRef]
2. Fildes, R.; Ma, S.; Kolassa, S. Retail forecasting: Research and practice. *Int. J. Forecast.* **2022**, *38*, 1283–1318. [CrossRef]
3. Oliveira, J.M.; Ramos, P. Assessing the Performance of Hierarchical Forecasting Methods on the Retail Sector. *Entropy* **2019**, *21*, 436. [CrossRef] [PubMed]
4. Theodoridis, G.; Tsadiras, A. Retail Demand Forecasting: A Multivariate Approach and Comparison of Boosting and Deep Learning Methods. *Int. J. Artif. Intell. Tools* **2024**, *33*, 2450001. [CrossRef]
5. Ramos, P.; Oliveira, J.M. A procedure for identification of appropriate state space and ARIMA models based on time-series cross-validation. *Algorithms* **2016**, *9*, 76. [CrossRef]
6. Benidis, K.; Rangapuram, S.S.; Flunkert, V.; Wang, Y.; Maddix, D.; Turkmen, C.; Gasthaus, J.; Bohlke-Schneider, M.; Salinas, D.; Stella, L.; et al. Deep Learning for Time Series Forecasting: Tutorial and Literature Survey. *ACM Comput. Surv.* **2022**, *55*, 1–36. [CrossRef]
7. Ramos, P.; Oliveira, J.M. Robust Sales Forecasting Using Deep Learning with Static and Dynamic Covariates. *Appl. Syst. Innov.* **2023**, *6*, 85. [CrossRef]
8. Bojer, C.S.; Meldgaard, J.P. Kaggle forecasting competitions: An overlooked learning opportunity. *Int. J. Forecast.* **2021**, *37*, 587–603. [CrossRef]
9. Oliveira, J.M.; Ramos, P. Cross-Learning-Based Sales Forecasting Using Deep Learning via Partial Pooling from Multi-level Data. In Proceedings of the Engineering Applications of Neural Networks, León, Spain, 14–17 June 2023; Iliadis, L., Maglogiannis, I., Alonso, S., Jayne, C., Pimenidis, E., Eds.; Springer: Cham, Switzerland, 2023; pp. 279–290. [CrossRef]
10. Teixeira, M.; Oliveira, J.M.; Ramos, P. Enhancing Hierarchical Sales Forecasting with Promotional Data: A Comparative Study Using ARIMA and Deep Neural Networks. *Mach. Learn. Knowl. Extr.* **2024**, *6*, 2659–2687. [CrossRef]
11. Oliveira, J.M.; Ramos, P. Investigating the Accuracy of Autoregressive Recurrent Networks Using Hierarchical Aggregation Structure-Based Data Partitioning. *Big Data Cogn. Comput.* **2023**, *7*, 100. [CrossRef]
12. Islam, S.; Elmekki, H.; Elsebai, A.; Bentahar, J.; Drawel, N.; Rjoub, G.; Pedrycz, W. A comprehensive survey on applications of transformers for deep learning tasks. *Expert Syst. Appl.* **2024**, *241*, 122666. [CrossRef]
13. Oliveira, J.M.; Ramos, P. Evaluating the Effectiveness of Time Series Transformers for Demand Forecasting in Retail. *Mathematics* **2024**, *12*, 2728. [CrossRef]
14. Torres, J.F.; Hadjout, D.; Sebaa, A.; Martínez-Álvarez, F.; Troncoso, A. Deep Learning for Time Series Forecasting: A Survey. *Big Data* **2021**, *9*, 3–21. [CrossRef]
15. Bandara, K.; Shi, P.; Bergmeir, C.; Hewamalage, H.; Tran, Q.; Seaman, B. Sales Demand Forecast in E-commerce Using a Long Short-Term Memory Neural Network Methodology. In Proceedings of the Neural Information Processing, ICONIP 2019, Sydney, NSW, Australia, 12–15 December 2019; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2019; Volume 11955, pp. 462–474. [CrossRef]
16. Joseph, R.V.; Mohanty, A.; Tyagi, S.; Mishra, S.; Satapathy, S.K.; Mohanty, S.N. A hybrid deep learning framework with CNN and Bi-directional LSTM for store item demand forecasting. *Comput. Electr. Eng.* **2022**, *103*, 108358. [CrossRef]
17. Giri, C.; Chen, Y. Deep Learning for Demand Forecasting in the Fashion and Apparel Retail Industry. *Forecasting* **2022**, *4*, 565–581. [CrossRef]

18. Mogarala Guruvaya, A.; Kollu, A.; Divakarachari, P.B.; Falkowski-Gilski, P.; Praveena, H.D. Bi-GRU-APSO: Bi-Directional Gated Recurrent Unit with Adaptive Particle Swarm Optimization Algorithm for Sales Forecasting in Multi-Channel Retail. *Telecom* **2024**, *5*, 537–555. [CrossRef]
19. De Castro Moraes, T.; Yuan, X.M.; Chew, E.P. Deep Learning Models for Inventory Decisions: A Comparative Analysis. In *Proceedings of the Intelligent Systems and Applications*; Arai, K., Ed.; Springer: Cham, Switzerland, 2024; pp. 132–150. [CrossRef]
20. de Castro Moraes, T.; Yuan, X.M.; Chew, E.P. Hybrid convolutional long short-term memory models for sales forecasting in retail. *J. Forecast.* **2024**, *43*, 1278–1293. [CrossRef]
21. Wu, J.; Liu, H.; Yao, X.; Zhang, L. Unveiling consumer preferences: A two-stage deep learning approach to enhance accuracy in multi-channel retail sales forecasting. *Expert Syst. Appl.* **2024**, *257*, 125066. [CrossRef]
22. Sousa, M.; Loureiro, A.; Miguéis, V. Predicting demand for new products in fashion retailing using censored data. *Expert Syst. Appl.* **2025**, *259*, 125313. [CrossRef]
23. Huang, T.; Fildes, R.; Soopramanien, D. The value of competitive information in forecasting FMCG retail product sales and the variable selection problem. *Eur. J. Oper. Res.* **2014**, *237*, 738–748. [CrossRef]
24. Loureiro, A.; Miguéis, V.; da Silva, L.F. Exploring the use of deep neural networks for sales forecasting in fashion retail. *Decis. Support Syst.* **2018**, *114*, 81–93. [CrossRef]
25. Punia, S.; Nikolopoulos, K.; Singh, S.P.; Madaan, J.K.; Litsiou, K. Deep learning with long short-term memory networks and random forests for demand forecasting in multi-channel retail. *Int. J. Prod. Res.* **2020**, *58*, 4964–4979. [CrossRef]
26. Lim, B.; Arık, S.Ö.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [CrossRef]
27. Wang, C.H. Considering economic indicators and dynamic channel interactions to conduct sales forecasting for retail sectors. *Comput. Ind. Eng.* **2022**, *165*, 107965. [CrossRef]
28. Kao, C.Y.; Chueh, H.E. Deep Learning Based Purchase Forecasting for Food Producer-Retailer Team Merchandising. *Sci. Program.* **2022**, *2022*, 2857850. [CrossRef]
29. Ramos, P.; Oliveira, J.M.; Kourentzes, N.; Fildes, R. Forecasting Seasonal Sales with Many Drivers: Shrinkage or Dimensionality Reduction? *Appl. Syst. Innov.* **2023**, *6*, 3. [CrossRef]
30. Punia, S.; Shankar, S. Predictive analytics for demand forecasting: A deep learning-based decision support system. *Knowl.-Based Syst.* **2022**, *258*, 109956. [CrossRef]
31. Nasseri, M.; Falatouri, T.; Brandtner, P.; Darbanian, F. Applying Machine Learning in Retail Demand Prediction—A Comparison of Tree-Based Ensembles and Long Short-Term Memory-Based Deep Learning. *Appl. Sci.* **2023**, *13*, 11112. [CrossRef]
32. Wellens, A.P.; Boute, R.N.; Udenio, M. Simplifying tree-based methods for retail sales forecasting with explanatory variables. *Eur. J. Oper. Res.* **2024**, *314*, 523–539. [CrossRef]
33. Praveena, S.; Prasanna Devi, S. A Hybrid Deep Learning Based Deep Prophet Memory Neural Network Approach for Seasonal Items Demand Forecasting. *J. Adv. Inf. Technol.* **2024**, *15*, 735–747. [CrossRef]
34. Wen, R.; Torkkola, K.; Narayanaswamy, B.; Madeka, D. A Multi-Horizon Quantile Recurrent Forecaster. *arXiv* **2018**, arXiv:1711.11053.
35. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *Int. J. Forecast.* **2020**, *36*, 1181–1191. [CrossRef]
36. Rasul, K.; Seward, C.; Schuster, I.; Vollgraf, R. Autoregressive Denoising Diffusion Models for Multivariate Probabilistic Time Series Forecasting. In Proceedings of the 38th International Conference on Machine Learning, Online, 18–24 July 2021; Volume 139, pp. 8857–8868.
37. Rasul, K.; Sheikh, A.S.; Schuster, I.; Bergmann, U.; Vollgraf, R. Multivariate Probabilistic Time Series Forecasting via Conditioned Normalizing Flows. *arXiv* **2021**, arXiv:2002.06103.
38. Hasson, H.; Wang, B.; Januschowski, T.; Gasthaus, J. Probabilistic Forecasting: A Level-Set Approach. In Proceedings of the Advances in Neural Information Processing Systems, Online, 7 December 2021; Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W., Eds.; Curran Associates, Inc.: San Jose, CA, USA, 2021; Volume 34, pp. 6404–6416.
39. Rangapuram, S.S.; Werner, L.D.; Benidis, K.; Mercado, P.; Gasthaus, J.; Januschowski, T. End-to-End Learning of Coherent Probabilistic Forecasts for Hierarchical Time Series. In Proceedings of the 38th International Conference on Machine Learning, Online, 18–24 July 2021; Meila, M., Zhang, T., Eds.; PMLR; *Proceedings of Machine Learning Research;* Volume 139, pp. 8832–8843.
40. Kan, K.; Aubet, F.X.; Januschowski, T.; Park, Y.; Benidis, K.; Ruthotto, L.; Gasthaus, J. Multivariate Quantile Function Forecaster. In Proceedings of the 25th International Conference on Artificial Intelligence and Statistics, Virtual, 28–30 March 2022; PMLR; *Proceedings of Machine Learning Research*; Volume 151, pp. 10603–10621.

41. Shchur, O.; Turkmen, C.; Erickson, N.; Shen, H.; Shirkov, A.; Hu, T.; Wang, Y. AutoGluon-TimeSeries: AutoML for Probabilistic Time Series Forecasting. In Proceedings of the International Conference on Automated Machine Learning, Potsdam, Germany, 12–15 November 2023; PMLR; pp. 9–1.
42. Tong, J.; Xie, L.; Yang, W.; Zhang, K.; Zhao, J. Enhancing time series forecasting: A hierarchical transformer with probabilistic decomposition representation. *Inf. Sci.* **2023**, *647*, 119410. [CrossRef]
43. Sprangers, O.; Schelter, S.; de Rijke, M. Parameter-efficient deep probabilistic forecasting. *Int. J. Forecast.* **2023**, *39*, 332–345. [CrossRef]
44. Olivares, K.G.; Meetei, O.N.; Ma, R.; Reddy, R.; Cao, M.; Dicker, L. Probabilistic hierarchical forecasting with deep Poisson mixtures. *Int. J. Forecast.* **2024**, *40*, 470–489. [CrossRef]
45. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.u.; Polosukhin, I. Attention is All you Need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30, pp. 5998–6008.
46. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 11106–11115. [CrossRef]
47. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–14 December 2021; Volume 34, pp. 22419–22430.
48. Woo, G.; Liu, C.; Sahoo, D.; Kumar, A.; Hoi, S. ETSformer: Exponential Smoothing Transformers for Time-series Forecasting. *arXiv* **2022**, arXiv:2202.01381.
49. Liu, Y.; Wu, H.; Wang, J. Non-stationary transformers: Exploring the stationarity in time series forecasting. In Proceedings of the 36th Conference on Neural Information Processing Systems (NeurIPS 2022), New Orleans, LA, USA, 28 November–9 December 2022; *Adv. Neural Inf. Process. Syst.*;Volume 35, pp. 9881–9893.
50. Kitaev, N.; Łukasz, K.; Levskaya, A. Reformer: The Efficient Transformer. *arXiv* **2020**, arXiv:2001.04451.
51. Rasul, K. Time Series Transformer. *Hugging Face*. Available online: https://huggingface.co/docs/transformers/en/model_doc/time_series_transformer (accessed on 6 September 2024).
52. Casolaro, A.; Capone, V.; Iannuzzo, G.; Camastra, F. Deep Learning for Time Series Forecasting: Advances and Open Problems. *Information* **2023**, *14*, 598. [CrossRef]
53. Ansari, A.F.; Stella, L.; Turkmen, C.; Zhang, X.; Mercado, P.; Shen, H.; Shchur, O.; Rangapuram, S.S.; Arango, S.P.; Kapoor, S.; et al. Chronos: Learning the Language of Time Series. *arXiv* **2024**, arXiv:2403.07815.
54. Rasul, K.; Ashok, A.; Williams, A.R.; Ghonia, H.; Bhagwatkar, R.; Khorasani, A.; Bayazi, M.J.D.; Adamopoulos, G.; Riachi, R.; Hassen, N.; et al. Lag-Llama: Towards Foundation Models for Probabilistic Time Series Forecasting. *arXiv* **2024**, arXiv:2310.08278.
55. Alexandrov, A.; Benidis, K.; Bohlke-Schneider, M.; Flunkert, V.; Gasthaus, J.; Januschowski, T.; Maddix, D.C.; Rangapuram, S.; Salinas, D.; Schulz, J.; et al. GluonTS: Probabilistic and Neural Time Series Modeling in Python. *J. Mach. Learn. Res.* **2020**, *21*, 4629–4634.
56. Rasul, K. pytorch-transformer-ts. 2021. Available online: https://github.com/kashif/pytorch-transformer-ts (accessed on 4 December 2024).
57. Makridakis, S.; Spiliotis, E.; Assimakopoulos, V. The M5 competition: Background, organization, and implementation. *Int. J. Forecast.* **2022**, *38*, 1325–1336. [CrossRef]
58. Makridakis, S.; Spiliotis, E.; Assimakopoulos, V. M5 accuracy competition: Results, findings, and conclusions. *Int. J. Forecast.* **2022**, *38*, 1346–1364. [CrossRef]
59. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 4–8 August 2019.
60. Gasthaus, J.; Benidis, K.; Wang, Y.; Rangapuram, S.S.; Salinas, D.; Flunkert, V.; Januschowski, T. Probabilistic Forecasting with Spline Quantile Function RNNs. In Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics, Naha, Japan, 16–18 April 2019; Chaudhuri, K., Sugiyama, M., Eds.; PMLR; *Proceedings of Machine Learning Research*; Volume 89, pp. 1901–1910.
61. Hyndman, R.J.; Koehler, A.B. Another look at measures of forecast accuracy. *Int. J. Forecast.* **2006**, *22*, 679–688. [CrossRef]
62. Koenker, R.; Hallock, K.F. Quantile Regression. *J. Econ. Perspect.* **2001**, *15*, 143–156. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Benchmarking Analysis of Railway Infrastructure Managers: A Hybrid Principal Component Analysis (PCA), Grey Best–Worst Method (G-BWM), and Assurance Region Data Envelopment Analysis (AR-DEA) Model

Snežana Tadić [1], Aida Kalem [2], Mladen Krstić [1,3,*], Nermin Čabrić [2], Adisa Medić [2] and Miloš Veljović [1]

[1] Logistics Department, Faculty of Transport and Traffic Engineering, University of Belgrade, Vojvode Stepe 305, 11000 Belgrade, Serbia; s.tadic@sf.bg.ac.rs (S.T.); m.veljovic@sf.bg.ac.rs (M.V.)
[2] Faculty of Traffic and Communications, University of Sarajevo, Zmaja od Bosne 8, 71000 Sarajevo, Bosnia and Herzegovina; aida.kalem@fsk.unsa.ba (A.K.); nermin.cabric@fsk.unsa.ba (N.Č.); adisa.medic@fsk.unsa.ba (A.M.)
[3] Department of Economic Sciences, University of Salento, Via Monteroni Snc, 73100 Lecce, Italy
* Correspondence: m.krstic@sf.bg.ac.rs or mladen.krstic@unisalento.it

Abstract: Benchmarking railway infrastructure managers (RIMs) has become a crucial tool in the context of European transport market liberalization, facilitating efficiency improvements and strategic decision-making. RIMs face challenges in increasing capacity, optimizing operations, and ensuring competitive, safe, and economically sustainable services. To address these challenges, this study proposes a hybrid benchmarking model that integrates Principal Component Analysis (PCA) to identify key performance indicators (KPIs) and reduce data dimensionality, the Grey Best–Worst Method (G-BWM) to determine KPI weight coefficients based on expert evaluations, and Assurance Region Data Envelopment Analysis (AR-DEA) to assess the relative efficiency of RIMs while incorporating real-world constraints. The research findings confirm that RIM8 is the most efficient unit, driven by high electrification levels, strong accident prevention measures, and optimal use of infrastructure. In contrast, RIM2 and RIM4 record the lowest efficiency scores, primarily due to poor safety performance, high infrastructure-related delays, and suboptimal resource utilization. By introducing weight constraints through AR-DEA, the model ensures that efficiency assessments reflect actual operational conditions, rather than relying on unrestricted weight allocations. The main contribution of this study lies in developing a systematic and objective framework for evaluating RIM efficiency, ensuring consistency and reliability in performance measurement. The practical implications extend to policy development and operational decision-making, providing insights for infrastructure managers, regulatory bodies, and policymakers to optimize resource allocation, enhance infrastructure resilience, and improve railway sector sustainability. The results highlight key efficiency factors and offer guidance for targeted improvements, reinforcing benchmarking as a valuable tool for long-term railway infrastructure management and investment planning. By offering a quantitatively grounded efficiency assessment, this model contributes to the competitiveness and sustainability of railway networks across Europe.

Keywords: benchmarking; railway infrastructure managers; KPIs; performance; PCA; G-BWM; AR-DEA

MSC: 03B52; 90B50

1. Introduction

Benchmarking, although defined in various ways, is widely accepted as a structured process aimed at improving standards through the adoption of best practices. Its primary purpose is the implementation of proven solutions in dynamic environments, serving as a tool for comparison and evaluation of all aspects of operations, products, or services against leading industry examples [1]. Traditionally, the focus of benchmarking has been on performance analysis in the later stages of the process, where retrospective evaluation of results supports planned organizational changes. However, to fully realize the potential of this method, its application needs to be expanded to anticipate unexpected changes and integrate it into the early stages of the process [2]. In addition to enabling organizations to identify key processes that require improvement, benchmarking provides an opportunity to adopt applicable solutions from industry best practices [3,4]. Modern approaches increasingly emphasize proactive and preventive indicators that extend beyond financial metrics. These indicators enable the prediction of future outcomes and active influence over them, ensuring business stability, more efficient utilization of knowledge resources, and the strengthening of an organization's competitive advantage [5,6]. Benchmarking has become an integral part of various sectors, including aviation, road, maritime, and railway transportation (performance analysis of airports and terminals [7], efficiency assessment of container terminals and ports [8], road safety evaluation [9], and assessment of service quality, safety, and punctuality of railway operators [10]). Also, it is widely applied in industries such as manufacturing, healthcare services, finance, education, and the public sector [11]. Specifically, in the research of railway infrastructure managers (RIMs), which are the subject of this study, the benchmarking framework has been used for measuring railways' safety on a national [12] and cross-continental level [13], environmental and economic impacts on high-speed rails [14], the success of railway construction megaprojects [15], railway power management systems [16], etc.

European railway infrastructure is facing increasing operational demands, with rail networks expected to accommodate a 25% growth in freight transport and a 40% rise in passenger traffic by 2050 [17]. However, inefficiencies in infrastructure management remain a major concern, with studies indicating that outdated systems and operational bottlenecks can reduce network capacity by up to 30% [18]. This underscores the critical need for robust benchmarking tools to enhance performance and sustainability in railway infrastructure management. Effective railway infrastructure management is crucial for enhancing transportation systems, facilitating sustainable economic development, improving connectivity for people and goods, and reducing environmental impacts. As an environmentally friendly mode of transport, rail transport plays a significant role in achieving sustainable development goals. However, the complexity of railway infrastructure management, coupled with challenges such as outdated infrastructure, necessitates advanced tools for performance analysis and improvement. Benchmarking has come to the fore as an important performance assessment, best practice identification, and strategic decision-making tool to enhance efficiency and competitiveness in the sector. Benchmarking is a valuable tool for railway infrastructure managers to compare their performance against reference values, identify and address areas for improvement, and adopt best practices. This process is led by the Platform of Rail Infrastructure Managers in Europe (PRIME), which has created a catalogue of standardized key performance indicators (KPIs). The indicators encompass almost all the performance dimensions of the project, such as its contextual framework, safety, environment, performance, service delivery, finance, and growth. The complexity of railway systems, the numerous factors that influence them, and heterogeneity of data are the major challenges to traditional methods of performance

analysis. With the use of standard KPIs and advanced analytical methods, benchmarking is not only a comparison tool for performance, but also a decision-making tool for strategic infrastructure management and resource optimization.

This study is based on the application of KPIs as input data for the development of a hybrid benchmarking model for assessing the performance of railway infrastructure managers. The proposed model integrates three methods: Principal Component Analysis (PCA) for data dimensionality reduction, the Grey Best–Worst Method (G-BWM) for determining the weight coefficients of KPIs, and Assurance Region Data Envelopment Analysis (AR-DEA) for evaluating relative efficiency. This approach enables a precise and comprehensive performance assessment, considering the complexity and diversity of the railway infrastructure sector. PCA allows for dimensionality reduction, identifying key KPIs that best explain variability within the dataset. The G-BWM provides a reliable and flexible approach for determining weight coefficients in conditions of uncertainty and limited information. By leveraging expert assessments, the G-BWM identifies the most important (best) and least important (worst) KPIs and, through pairwise comparisons, calculates optimal weight coefficients. This ensures that weight coefficients accurately reflect the true significance of each KPI, enabling a fair performance evaluation. This process is particularly important for sectors with complex and heterogeneous operational environments, such as the railway industry. Traditional DEA analysis often has limitations, due to the possibility that certain inputs or outputs receive zero weight, which compromises result accuracy and disregards critical factors. The AR-DEA (Assurance Region DEA) method overcomes this issue by introducing weight constraints ("assurance regions") into the optimization process. These assurance regions define boundaries for input and output weight coefficients, ensuring that all key factors are included in the analysis. The AR-DEA method utilizes the weight coefficients derived through the G-BWM as initial values for setting assurance region limits. This establishes a link between subjective expert assessments and objective efficiency analysis, balancing model flexibility and accuracy. This integration enables the assessment of the relative efficiency of decision-making units (DMUs) while maintaining consistency in weight allocation. The application of the proposed model aims to enhance the benchmarking process by precisely identifying efficient and inefficient railway infrastructure managers. The model allows for a detailed comparison of different managers' performances, highlighting best practices that can serve as guidelines for improving less efficient units. Furthermore, the model provides an in-depth understanding of key factors influencing performance, ensuring that decision-makers receive accurate and reliable information for strategic planning, resource optimization, and evidence-based decision-making in railway infrastructure management. This study makes a significant contribution to the development of methodologies for monitoring and improving railway infrastructure performance, establishing a foundation for future research and innovation in resource management optimization and railway system sustainability.

This study proposes a hybrid benchmarking model that integrates PCA, the G-BWM, and AR-DEA to address key efficiency challenges in railway infrastructure management. By offering a quantitatively grounded and decision-maker-informed efficiency assessment framework, the model provides actionable insights for optimizing infrastructure operations, guiding investment strategies, and ensuring long-term sustainability in the sector. Accordingly, the rest of the paper is organized as follows: Section 2 provides a review of the relevant literature, focusing on benchmarking in the railway sector, efficiency concepts, and the application of advanced methodologies, including PCA, the G-BWM, and AR-DEA. Section 3 presents a detailed description of the hybrid methodology proposed in this study, outlining the specific steps for integrating PCA, the G-BWM, and AR-DEA. Section 4

applies the methodology to a selected sample of nine RIMs, demonstrating its practical implementation and results. Section 5, the discussion section, analyzes the obtained results, exploring factors influencing efficiency, the advantages and limitations of the proposed methodology, and implications for both theory and practice. Finally, Section 6 concludes the study, summarizing the key findings and providing recommendations for future research.

2. Literature Review and Key Performance Indicators

This section presents a review of the relevant literature on benchmarking in the railway sector, with a focus on efficiency and the application of advanced methodological approaches. Special attention will be given to the use of methods such as Principal Component Analysis (PCA), the Grey Best–Worst Method (G-BWM), and Assurance Region Data Envelopment Analysis (AR-DEA), which are employed for performance analysis and improvement in railway transport. This review aims to provide a more profound understanding of existing theoretical and practical approaches that are relevant to this research.

2.1. Methodological Approaches in Benchmarking

For the development of this paper, the Systematic Literature Review (SLR) method was applied to evaluate research questions related to the application of hybrid models for benchmarking analysis of railway infrastructure managers. The systematic literature review was conducted through five phases. In the first phase, selection of scientific papers published in the last five years was performed to ensure the relevance and contemporaneity of the analysis. The selection of scientific papers published in the last five years ensured that the review would capture the most current and relevant research, aligning the study with contemporary advancements in hybrid models for benchmarking railway infrastructure managers. The second phase involved defining inclusion criteria, where key search terms were focused on rail transport, infrastructure managers, performance, efficiency, and benchmarking. Defining inclusion criteria with specific search terms helps to focus the review on key areas, such as rail transport, infrastructure managers, and performance benchmarking, ensuring that only relevant and targeted literature is included in the analysis. The third phase involved structuring the literature according to the applied methodologies (PCA, the BWM, DEA, the G-BWM, and AR-DEA). Structuring the literature by applied methodologies allowed for a systematic comparison of different approaches and their application to benchmarking analysis, facilitating the identification of trends and gaps in the current body of knowledge. This was followed by the fourth phase, in which the areas of application were analyzed to determine whether the combination of PCA, the G-BWM, and AR-DEA had been previously applied in railway transport or other sectors. The literature search was conducted using the following meta-query: (TITLE-ABS-KEY (“benchmarking” OR “performance” OR “efficiency”)) AND (TITLE-ABS-KEY (“railway infrastructure” OR “railway infrastructure managers” OR “rail transport” OR “traffic” OR “transport”)) AND (TITLE-ABS-KEY (“PCA” OR “principal component analysis” OR “G-BWM” OR “grey best-worst method” OR “BWM” OR “best-worst method” OR “AR DEA” OR “assurance region DEA” OR “DEA” OR “data envelopment analysis”)) AND ((LIMIT-TO (DOCTYPE, “ar”) OR LIMIT-TO (DOCTYPE, “re”) OR LIMIT-TO (DOCTYPE, “cp”) OR LIMIT-TO (DOCTYPE, “cr”) OR LIMIT-TO (DOCTYPE, “bo”))) AND (LIMIT-TO (LANGUAGE, “English”)). Finally, the fifth phase of the research focused on identifying the objectives and purposes of the proposed models in the reviewed papers. Identifying the objectives and purposes of the proposed models in the reviewed papers contributes to a deeper understanding of the practical outcomes and goals of applying hybrid models in benchmarking, guiding future research and practical implementation strategies. The results of the systematic literature review include an analysis of 30 scientific papers. The

review enabled the identification of key methodologies and approaches, highlighting PCA, the G-BWM, and AR-DEA as fundamental components for the hybrid model, integrating dimensionality reduction, weight coefficient determination, and relative efficiency evaluation. Table 1 presents an overview of the analyzed studies, structured according to the defined inclusion criteria, including year of publication, key terms, applied methods, application domains, and objectives/purposes of the proposed models. Additionally, key challenges and emerging trends were identified, which will be further analyzed in the following sections of the study.

Table 1. Literature review.

Ref.	Crit. I—Year of Publication	Crit. II—Keywords	Criterion III—Applied Methods						Scope of Model		Purpose/Purpose of Model
			PCA	BWM	DEA	GBWM	AR DEA	Other Methods	Railway Traffic	Other Areas	
[19]	2024	Railway Carrying Capacity, Coordination Optimization	x					EFA, Regression, Scenario Analysis	x		Identification of key energy efficiency factors and optimization of railway systems.
[20]	2019	Sustainable transport, rail freight market, PCA	x						x		Analysis of impact of market liberalization on competitiveness of rail freight transport.
[21]	2021	PCA, railway maintenance	x					ANOVA, Kruskal–Wallis	x		Improvement of monitoring and maintenance of railway wheels.
[22]	2022	Life management, maintenance planning	x					Decision Trees, Bayesian Networks	x		Optimization of railway infrastructure maintenance.
[23]	2019	Maintenance; PCA; factor analysis	x					FA	x		Identification of key human factors that affect safety and efficiency of railway system maintenance.
[24]	2024	Rail network, BWM, decision-making		x				Lingo 17 softver	x		Identification and ranking of risk factors affecting rail transport safety.
[25]	2021	Urban rail, performance evaluation, BWM		x				Large-Scale Group Evaluation Technology, CS	x		Performance evaluation of PPP projects in urban rail transport.
[26]	2023	Rail transportation projects, Fuzzy MCDM		x				Stochastic VIKOR	x		Assessment and selection of railway development projects.
[27]	2022	Cross-regional railways, MCA, BWM		x				FAISM, DFS	x		Analysis of impact of changes in demand and supply on inter-regional rail connectivity in post-COVID-19 period.
[28]	2024	Performance evaluation, MCDM, G-BWM				x		RATMI, MCDM		x	Performance evaluation of retail warehouses.

Table 1. *Cont.*

Ref.	Crit. I—Year of Publication	Crit. II—Keywords	Criterion III—Applied Methods						Scope of Model		Purpose/Purpose of Model
			PCA	BWM	DEA	GBWM	AR DEA	Other Methods	Railway Traffic	Other Areas	
[29]	2023	intermodal terminal; Governance model, GBWM, GCOBRA				x		GCOBRA		x	Evaluation of governance models in intermodal terminal operations.
[30]	2024	Supply chain, circular economy, GBWM, GCOBRA				x		GCOBRA		x	Evaluation of agri-food supply chain risks.
[31]	2022	City logistics, Industry 4.0, GBWM, GCODAS				x		GCODAS		x	Evaluation of smart city logistics solutions.
[32]	2023	Railway transport, three-stage DEA,			x				x		Evaluation of efficiency of rail transport.
[33]	2019	railways, efficiency, benchmarking			x				x		Assessment of efficiency of railway operations.
[34]	2019	DEA, performance measurement			x			Two-stage clustering approach	x		Assessment of socio-technical performance of TCC staff.
[35]	2020	Railway undertaking, efficiency, DEA			x			FAHP	x		Assessment of efficiency of freight railway companies.
[36]	2019	Efficiency, Fuzzy EDAS, AR DEA					x	Fuzzy EDAS		x	Evaluation of efficiency of small inland intermodal terminals.
[37]	2023	Railway, efficiency measurement, NDEA			x			SBM, Network SBM	x		Analysis of efficiency of railway stations.
[38]	2023	DEA			x			SBM		x	Assessment of regional differences in vehicle environmental efficiency.
[39]	2019	Efficiency, rail systems, railways						Meta-analysis, Permutation Test	x		Analysis of factors affecting efficiency of railway systems.
[40]	2020	Railway indicators, HSR performance						SLR, expert consultation	x		Analysis of sustainability reporting practices of HSR operators.
[41]	2022	PCA–DEA, efficiency, transport	x		x			CRITIC, Entropy, MARCOS		x	Evaluation of efficiency of transport companies.
[42]	2020	Eco-efficiency, transport sector	x		x			MLR		x	Assessing eco-efficiency of transport in Nairobi and identifying factors for improving sustainability.

Table 1. *Cont.*

Ref.	Crit. I—Year of Publication	Crit. II—Keywords	Criterion III—Applied Methods						Scope of Model		Purpose/Purpose of Model
			PCA	BWM	DEA	GBWM	AR DEA	Other Methods	Railway Traffic	Other Areas	
[43]	2020	DEA, BWM, road safety		x	x					x	Evaluation of efficiency of road traffic safety.
[44]	2020	DEA			x			AHP, VIKOR, MPI		x	Performance assessment of public road transport in India.
[45]	2023	Hospital waste management, DEA, PCA	x		x			ANP, CRS, Shannon's Entropy, CES		x	Assessment of efficiency of biomedical waste management in hospitals.
[46]	2023	BWM, DEA, PCA	x	x	x			SWOT		x	Assessing efficiency of private banks using CAMELS indicators.
[47]	2021	DEA			x			MLR, Graph-Theoretic Approach		x	Assessment of changes in efficiency and financial performance of companies in automotive industry.
[48]	2024	Hierarchical DEA			x					x	Assessment of logistics performance of countries.
[49]	2020	Logistics industry performance, SBM-DEA, PCA	x		x			Tobit Regression		x	Assessing efficiency of China's logistics industry.
[50]	2021	PCA; relative efficiency	x		x			VIKOR		x	Systematic assessment and ranking of suppliers in textile industry.
[51]	2022	DEA	x		x					x	Evaluation of supplier efficiency in food industry.
[52]	2020	PCA, DEA	x				x	Cluster Analysis, Tobit Regression		x	Energy system performance assessment of 125 countries.

Through the literature review, various applications of advanced analytical methods, including PCA, DEA, and the BWM, have been identified. PCA has been used for dimensionality reduction and the identification of key variables in studies on energy efficiency in railway systems [19], the market competitiveness of railway freight transport [20], optimization of railway wheel maintenance [21], key human factors [23], and railway infrastructure lifecycle management [22]. The BWM has been applied for determining weight coefficients of risk factors in railway safety analyses [24], evaluating public–private partnerships [25], analyzing the sustainability of railway projects [26], and assessing the impact of changes in demand and supply on inter-regional railway connectivity [27]. In addition to these methods, researchers have employed other techniques to enhance analytical frameworks. For example, meta-analysis has been used to aggregate results from multiple studies and identify key efficiency variables in the railway sector [39].

The DEA method, known for its ability to assess the relative efficiency of decision-making units, has found extensive application in the railway sector. It has been used for evaluating the efficiency of railway transport systems in international and regional contexts [32,33]. Furthermore, DEA has been applied in socio-technical performance analysis [34] and in assessing the efficiency of freight railway companies [35], identifying key criteria for improving competitiveness. Additionally, network DEA models have enabled a deeper analysis of railway station efficiency [37], while specific DEA variants, such as the SBM and AR-DEA, have been employed to evaluate the performance of small intermodal terminals [36]. Although these methods have significantly contributed to understanding various aspects of the railway sector, their integration into a hybrid benchmarking model for railway infrastructure managers remains unexplored. The motivation for developing such a model stems from the application of these methods in other sectors. While the combination of PCA, the BWM, and DEA has been applied in the banking sector [46], it refers to their basic forms. An enhanced version that incorporates the G-BWM for improved flexibility and precision in weight determination under uncertainty, and AR-DEA, which introduces constraints on weight coefficients for a more realistic efficiency evaluation, represents a significant methodological advancement. This improved combination of methods has not been previously applied in the railway sector or other industries, underscoring the novelty and potential of the proposed hybrid model for enhancing performance evaluation and strategic decision-making.

The development of a hybrid model that combines PCA, the G-BWM, and AR-DEA represents a logical advancement in methodological frameworks, particularly within the railway sector. This model enables a more systematic analysis of complex interactions between key performance indicators (KPIs), where each method contributes to specific aspects of evaluation—from identifying key variables and ensuring reliable weighting of criteria, to accurately measuring relative efficiency. The model enhances performance evaluation by addressing DEA's limitations, incorporating decision-maker preferences through weight bounds, and using the BWM to capture those preferences in a grey environment, which allows for handling imprecise information. Additionally, PCA reduces dimensionality and improves discrimination power, ensuring more accurate, efficient, and realistic identification of efficient DMUs, making the model valuable for performance assessment across various complex, multi-criteria environments. The proposed approach offers a unique opportunity for developing dynamic and adaptable infrastructure management strategies, thereby contributing to the long-term sustainability and competitiveness of the railway sector. Its application opens new perspectives for enhancing analytical practices in transportation and beyond, providing a methodological framework that can be adapted to the specific needs and challenges of other industries.

2.2. Role of KPIs in Railway Benchmarking

Benchmarking in the railway sector plays a key role in performance assessment by identifying best practices and comparing them with reference values. According to research, benchmarking facilitates the understanding of the relative efficiency of different railway systems, and supports strategic decision-making aimed at improving the operational and financial aspects of infrastructure management [33]. Various studies have utilized benchmarking analyses to evaluate the efficiency of railway companies and identify best practices for enhancing operational performance. KPI-based benchmarking analysis is widely applied in the railway sector to optimize different operational aspects. For instance, benchmarking analysis is used for improving the energy efficiency of urban railway systems [53], assessing service quality in railway transport [54], enhancing national railway system safety [12], comparing performance measurement standards among different rail-

way operators [55], evaluating railway system performance across Europe [56], assessing high-speed rail network operational readiness [57], and evaluating the performance of infrastructure managers in the railway sector [58].

Benchmarking utilizing KPIs serves as a critical tool for assessing and improving railway infrastructure management, enabling systematic performance comparisons across different networks and identifying best practices for efficient management. The European Commission, through the PRIME (Platform for European Railway Infrastructure Managers) initiative, has established a catalogue of KPIs to standardize performance monitoring and enhance railway infrastructure management. This system covers a broad range of aspects, including contextual framework, safety, environment, performance, service delivery, finance, and growth, allowing for a structured approach to analyzing and comparing infrastructure managers' performance across Europe [59]. By utilizing KPIs, railway infrastructure managers can systematically monitor and analyze key parameters that impact the efficiency and safety of the railway system. The benchmarking process facilitates performance comparisons among different railway infrastructure managers, thereby identifying best practices. Using KPIs as a benchmarking foundation ensures precise evaluation and transparency, as clear metrics can be applied to assess success across various domains. The PRIME initiative not only provides a framework for internal evaluation, but also enables comparisons between different national and regional networks, fostering best practices in the railway transport sector. Through this methodology, performance comparisons can drive innovation and alignment with European standards, ultimately contributing to better conditions for passengers and more efficient railway system operations across Europe. Key challenges in benchmarking analysis within the railway sector include limited data availability, inconsistencies between different databases, and variations in regulatory frameworks across countries [60]. Consequently, methodologically robust approaches must be applied to ensure reliable and accurate performance comparisons. To ensure precise and reliable benchmarking, this paper employs KPIs of railway infrastructure managers. The benchmarking analysis is based on the application of PCA, G-BWM, and AR-DEA methods, which enable a comprehensive evaluation of railway infrastructure managers' performance.

3. Methodology—Hybrid PCA–G-BWM–AR-DEA Model

Benchmarking the performance of railway infrastructure managers is a key tool for identifying best practices and improving efficiency. This analysis requires a systematic approach that incorporates KPIs to ensure an objective and reliable evaluation. Given the complexity of infrastructure management, these indicators encompass various efficiency aspects, including operational, financial, technical, and environmental factors. However, evaluation challenges often arise, due to variable interdependencies, data multidimensionality, and subjectivity in determining the importance of individual criteria. To address these challenges, this paper proposes a hybrid model that integrates PCA, the G-BWM, and AR-DEA, leveraging their combined advantages to establish a comprehensive and robust framework for benchmarking railway infrastructure managers' performance based on KPIs. The proposed hybrid model offers a significant advancement in performance evaluation by addressing key limitations of the traditional DEA method. DEA's automatic weight assignment often leads to unrealistic evaluations, and the AR DEA model overcomes this by incorporating decision-maker preferences through weight bounds. However, defining these bounds requires a reliable method for capturing preferences, which is why the BWM is employed. The grey environment further enhances the model's robustness by enabling it to handle imprecise or incomplete information. Additionally, PCA is integrated to reduce dimensionality and improve discrimination power by focusing on the most critical KPIs, which enhances both computational efficiency and the accuracy of the evaluation. This

combination of methods ensures more flexible, accurate, and realistic identification of efficient DMUs, contributing to a more effective performance evaluation framework. This model's contributions to the field include improved discrimination power, computational efficiency, and applicability to complex, multi-criteria environments, making it a valuable tool for performance assessment across various disciplines.

PCA is employed to identify and extract the most relevant variables within the model. This method transforms the original dataset into a new set of variables—principal components—that maximize variance and reduce redundancy among indicators [61,62]. The first step in PCA involves data standardization, followed by the calculation of the covariance matrix, the determination of eigenvalues and eigenvectors, and the formation of feature vectors that define the principal components. The objective of this process is to eliminate collinearity among variables and enhance model stability. After dimensionality reduction, the G-BWM is applied to determine the weight coefficients of KPIs. The G-BWM is a Multi-Criteria Decision-Making (MCDM) method based on the grey system theory, allowing experts to quantify subjective preferences by identifying the best and worst criteria. The key steps in this method include defining the evaluation scale, selecting reference criteria, calculating grey vectors that describe the relative relationships among criteria, and estimating the initial and final weight coefficients. This approach enhances evaluation consistency under uncertainty and reduces subjective biases in weight assignment.

In the final stage, the AR-DEA method is used to assess the relative efficiency of railway infrastructure managers. DEA is a non-parametric method for measuring the efficiency of DMUs, employing linear programming to distinguish between efficient and inefficient units. In this study, AR-DEA extends the standard DEA model by introducing additional constraints that enhance the discriminatory power of the analysis, allowing for more precise ranking of units. The process involves data collection and normalization, the formation of input and output variable matrices, and the calculation of the relative efficiency of each DMU. The combination of these methods provides a comprehensive performance evaluation of railway infrastructure managers, integrating KPIs and expert assessments into the benchmarking process. The steps of the proposed model are described below and illustrated in Figure 1.

Step 1: Define the problem structure, i.e., the set of DMUs and the inputs and outputs for the efficiency evaluation.

Step 2: Apply PCA to analyze the relationships among input and output variables, reduce multicollinearity, and extract the most relevant components for further analysis [63].

Step 2.1: For the initial data $X_t^{(i)}$, where $i = 1, 2, \ldots, N$, a $t = 1, 2, \ldots, T$, standardization is performed using the following formula:

$$\tilde{X}_t^{(i)} = \frac{\left(X_t^{(i)} - \tilde{X}^{(i)} \right)}{\sigma \left(X^{(i)} \right)} \tag{1}$$

where $\tilde{X}^{(i)}$ is the mean value, and $\sigma \left(X^{(i)} \right)$ is the standard deviation of the variable $X^{(i)}$.

Step 2.2: Let $\sum$ denote the covariance matrix of the standardized data $\tilde{X}$; then, it can be decomposed as follows:

$$\sum = \Gamma \Lambda \Gamma' \tag{2}$$

where Λ is a positive diagonal matrix containing the eigenvalues λ_1, λ_1, …, λ_N, and the columns of the matrix Γ are the corresponding eigenvectors. It is assumed that the eigenvectors (principal components) are ordered in descending order of their eigenvalues,

where the first K eigenvectors represent the factor loadings $\left[q^{(1)}, q^{(2)}, \ldots, q^{(K)}\right]$. Then, K features $\left[F^{(1)}, F^{(2)}, \ldots \text{and } F^{(K)}\right]$ are obtained, which are defined as follows:

$$F_t^{(k)} = q^{(k)'} \widetilde{X}_t \tag{3}$$

where je $\widetilde{X}_t$ represents the N-dimensional vector of the standardized term structure at time t.

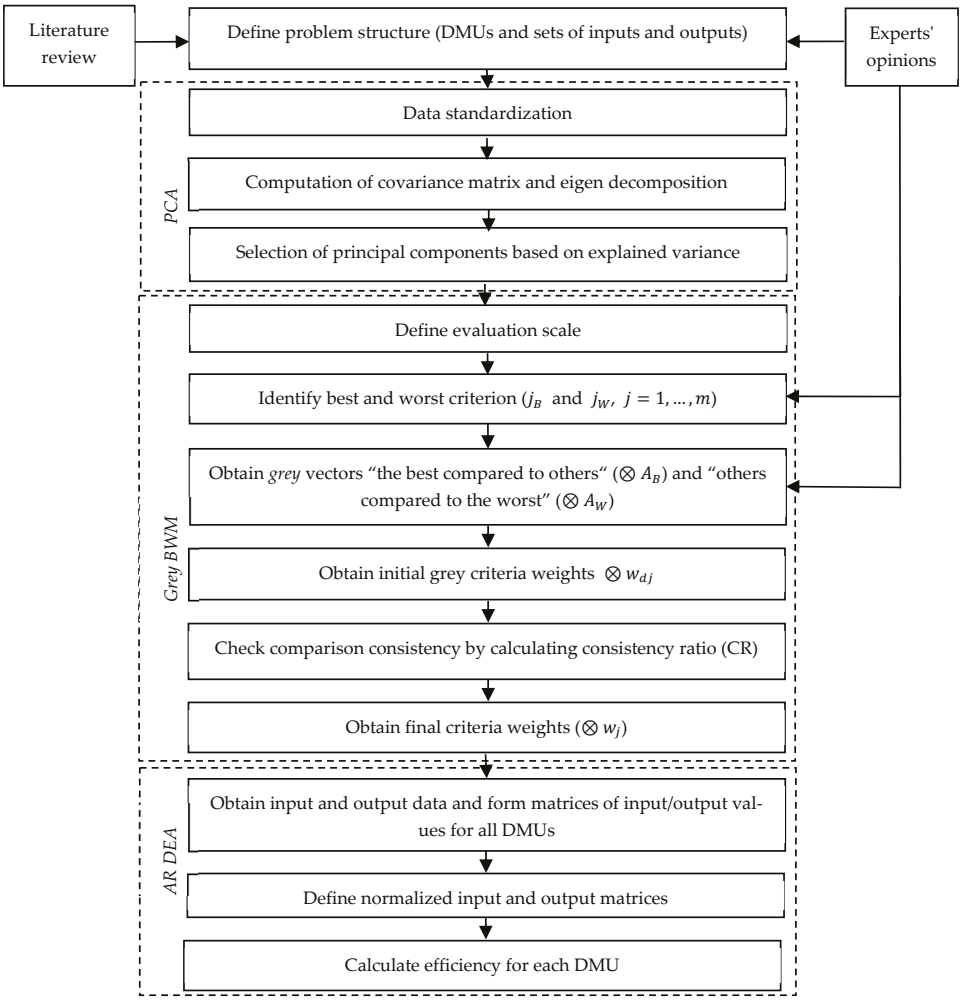


Figure 1. Structure of proposed model.

The covariance matrix captures the relationships between variables and helps to identify the directions of maximum variance, while the eigenvectors define the principal components along those directions, enabling dimensionality reduction and the transformation of data into a new uncorrelated feature space.

Step 2.3: To determine how well a principal component explains the data, the total variance and the proportion of variance of the m-th principal component are used, defined as follows:

$$\frac{\lambda_m}{\sum_{j=1}^{n} \lambda_j} \tag{4}$$

This proportion represents the explanatory power of each principal component. Accordingly, the principal component with the highest explanatory power becomes the first

principal component, followed by the second with the second highest explanatory power, and so on. The principal components that capture the largest portion of the total variance (typically 80–90%) are selected, ensuring that the most important aspects of the data are preserved for further analysis.

Step 3: Define the input/output weight limits by applying the G-BWM (adapted from [64].

Step 3.1: Define the grey scale for the evaluations (Table 2).

Table 2. Grey evaluation scale.

Linguistic Term	Abbreviation	Grey Scale
"None"	"N"	[0, 2]
"Very Low"	"VL"	[1, 3]
"Low"	"L"	[2, 4]
"Fairly Low"	"FL"	[3, 5]
"Moderate"	"M"	[4, 6]
"Fairly High"	"FH"	[5, 7]
"High"	"H"	[6, 8]
"Very High"	"VH"	[7, 9]
"Extremely High"	"EH"	[8, 10]

Step 3.2: Decision-makers (DMs) d $(d = 1, \ldots, f)$ select the best and the worst input/output (j_B and j_W, $j = 1, \ldots, m$), and then evaluate the other inputs/outputs according to them using the scale from Table 1, thus obtaining grey vectors „the best compared to others"—$\otimes A_B = (\otimes a_{B1}, \otimes a_{B2}, \ldots, \otimes a_{Bm})$, and "others compared to the worst"—$\otimes A_W = (\otimes a_{1W}, \otimes a_{2W}, \ldots, \otimes a_{mW})$.

Step 3.3: Obtain the optimal grey input/output weight limits $\otimes l_{d1}, \otimes l_{d2}, \ldots, \otimes l_{dm}$, $\forall d = 1, \ldots, f$ by solving the following optimization problem:

$$\text{s.t.} \begin{cases} \min \otimes \xi \\ \left| \frac{\otimes l_B}{\otimes l_{dj}} - \otimes a_{Bj} \right| \leq \otimes \xi \\ \left| \frac{\otimes l_{dj}}{\otimes l_W} - \otimes a_{jW} \right| \leq \otimes \xi \\ \sum_{j=1}^{m} L\left(\otimes l_{dj} \right) = 1 \\ \underline{l}_{dj} \leq \bar{l}_{dj} \\ \underline{l}_{dj} \geq 0 \\ j = 1, \ldots, m \end{cases} \tag{5}$$

where $\otimes \xi = \left[\underline{\xi}, \bar{\xi} \right]$, $\otimes l_B = \left[\underline{l}_B, \bar{l}_B \right]$ is the grey limit of the "best" criterion, $\otimes l_w = \left[\underline{l}_W, \bar{l}_W \right]$ is the grey limit of the "worst" input/output, $\otimes l_{dj} = \left[\underline{l}_{dj}, \bar{l}_{dj} \right]$ is the grey limit of the input/output j, $j = 1, \ldots, m$, $j \neq j_B, j_W$, $\otimes a_{Bj} = \left[\underline{a}_{Bj}, \bar{a}_{Bj} \right]$ is the grey preference of the "best" input/output over the input/output j, and $\otimes a_{jW} = \left[\underline{a}_{jW}, \bar{a}_{jW} \right]$ is the preference of the input/output j over the "worst" input/output. $L\left(\otimes l_j \right)$ is the white value of the grey number $\otimes l_j$:

$$L(\otimes l_{dj}) = (\underline{l}_{dj} + \bar{l}_{dj})/2 \tag{6}$$

Step 3.4: Check the comparison consistency by calculating the consistency ratio (*CR*) with the following formula:

$$CR = R(\otimes\xi)/CI \tag{7}$$

where $R(\otimes\xi)$ is the white value of the grey number $\otimes\xi$, obtained by (6), and *CI* is the consistency index:

$$CI^2 - (1 + 2\overline{a}_{BW})CI + \left(\overline{a}_{BW}^2 - \overline{a}_{BW}\right) = 0 \tag{8}$$

where

$$\otimes a_{BW} = \max_j \left\{ \otimes a_{Bj}, \otimes a_{jW} \right\} \tag{9}$$

The comparison is considered consistent if the *CR* value is close to 0.

Step 3.5: Obtain the final input/output limits, as follows:

$$\otimes l_j = \left[\underline{l}_j, \overline{l}_j \right], \ \forall j = 1, \ldots, m, \tag{10}$$

$$\underline{l}_j = \left(\prod_{d=1}^{l} \underline{l}_{dj} \right)^{1/d}, \tag{11}$$

$$\overline{l}_j = \left(\prod_{d=1}^{l} \overline{l}_{dj} \right)^{1/d}, \tag{12}$$

Step 4: Select the efficient DMUs by applying the AR DEA method (adapted from [65]).

Step 4.1: Define the input value matrices for all DMUs (X_{ij}), i.e., the output value matrices for all DMUs (Y_{is}), in the following manner:

$$X_{ij} = \left[x_{ij} \right]_{o \times e}, \tag{13}$$

$$Y_{is} = \left[y_{is} \right]_{o \times t} \tag{14}$$

where x_{ij} is the value of input j for DMU i, $i = 1, \ldots, o$, $j = 1, \ldots, e$, and y_{is} is the value of output s for DMU i, $s = 1, \ldots, t$. e and t represent the total number of inputs and outputs taken into consideration, respectively.

Step 4.2: Define the normalized matrices of inputs and outputs. The normalized matrices of inputs (X'_{ij}) and outputs (Y'_{is}) are formed by the normalized values of inputs and outputs, obtained in the following manner:

$$x'_{ij} = \frac{x_{ij}}{\max\limits_i x_{ij}}, \ \forall j = 1, \ldots, e, \tag{15}$$

$$y'_{is} = \frac{y_{is}}{\max\limits_i y_{is}}, \ \forall s = 1, \ldots, t \tag{16}$$

Step 4.3: Calculate the efficiencies for each DMU. In order to calculate the efficiencies for each DMU, it is necessary to solve the following linear optimization model:

$$max \sum_{s=1}^{t} u_s y_{is}, \tag{17}$$

subject to the following:

$$\sum_{j=1}^{e} v_j x_{ij} = 1, \tag{18}$$

$$\sum_{s=1}^{t} u_s y_{is} - \sum_{j=1}^{e} v_j x_{ij} \leq 0, \quad \forall i = 1, \ ..., \ o, \tag{19}$$

$$\underline{lv}_j \leq v_j \leq p\overline{lv}_j, \quad \forall j = 1, \ ..., \ e, \tag{20}$$

$$q\underline{lu}_s \leq u_s \leq q\overline{lu}_s, \quad \forall s = 1, \ ..., t, \tag{21}$$

$$p \geq 0, \ q \geq 0, \tag{22}$$

where v_j, $j = 1, \ldots, e$, and u_s, $s = 1, \ldots, t$, are the input and output weights, respectively; $\underline{lv}_j$ and $\overline{lv}_j$ are the lower and upper weight limits for the input j; and $\underline{lu}_s$ and $\overline{lu}_s$ are the lower and upper weight limits for the output s, calculated by applying Equations (11) and (12). p and q are the scaling factors, enabling the fulfilment of conditions for introducing the input and output weight bounds.

4. Application of Proposed Model for Benchmarking Analysis of RIMs

In this section, the hybrid model for benchmarking analysis of RIMs is applied, utilizing a combination of PCA, G-BWM, and AR-DEA methods. The main objective of this approach is to conduct a detailed evaluation of the performance of nine selected RIMs, using a set of 35 KPIs. These KPIs serve as the foundation for quantifying efficiency and identifying best practices in railway infrastructure management. This multidimensional approach enables a systematic analysis and comparison of results, taking into account various business aspects, including contextual framework, safety, environment, performance, service delivery, finance, and growth. In Step 1, the DMUs in this study consist of nine RIMs from different EU countries. To ensure data privacy and confidentiality, the identities of the RIMs will remain anonymous, with each manager being assigned an alphanumeric code (e.g., RIM1, RIM2, etc.). These units were carefully selected to ensure comparability, thereby enhancing the accuracy and validity of the results. For the selection of KPIs, this study relies on the set of indicators defined in the PRIME report of the European Commission, which covers key aspects of railway infrastructure performance. The selection of KPIs was based on their high priority and relevance for benchmarking analysis. The indicators were meticulously chosen to ensure measurability and comprehensiveness, covering all critical performance dimensions of railway infrastructure managers. Additionally, the KPIs used in this study were selected based on previous research [60], where the same indicators were applied to assess the performance of railway infrastructure managers. Data collection in this phase is of critical importance, as the data must be accurate, verified, and standardized to ensure reliable subsequent analysis. Ensuring high-quality data and aligning them with the required standards guarantees that the analysis remains valid and that the results are relevant for decision-making. All data used in this phase were collected from reliable sources, including relevant reports and the PRIME platform, ensuring consistency and analytical integrity [18]. In Step 2, the PCA method was applied to reduce the dimensionality of the dataset KPIs, thereby enhancing analytical efficiency and improving data interpretation. The PCA implementation was conducted using Minitab software version 17.1, which facilitated the necessary statistical computations and visualization of results, streamlining further analysis and data interpretation. Table 3 presents the results of the principal component analysis, including the eigenvalues, proportions, and cumulative proportions of variance for each principal component (PC1 to PC8).

Table 3. Eigenanalysis of correlation matrix.

	PC1	**PC2**	**PC3**	**PC4**	**PC5**	**PC6**	**PC7**	**PC8**
Eigenvalue	10.805	8.411	5.118	3.581	3.111	1.429	1.357	1.187
Proportion	0.309	0.240	0.146	0.102	0.089	0.041	0.039	0.034
Cumulative	0.309	0.549	0.695	0.798	0.887	0.927	0.966	1.000

The first principal component (PC1) has an eigenvalue of 10.805, explaining 30.9% of the total variance in the dataset. PC2 and PC3, with eigenvalues of 8.411 and 5.118, account for an additional 24% and 14.6% of the variance, respectively. By analyzing the eigenvalues and cumulative proportion of variance, it is evident that the first five components (PC1–PC5) collectively capture 88.7% of the total variance, which serves as an optimal threshold for retaining critical information while reducing dimensionality. The remaining components (PC6–PC8) contribute significantly less to the explained variance, and may be excluded from further analysis if the goal is to simplify the model. Figure 2 presents a scree plot of the extracted components, visually illustrating the distribution of eigenvalues and enabling the identification of key components that retain the highest variance. Table 4 shows the results, including factor loadings, which indicate the strength of the relationship between individual KPIs and principal components. These insights help to identify the most significant factors for decision-making and performance optimization in railway infrastructure management.

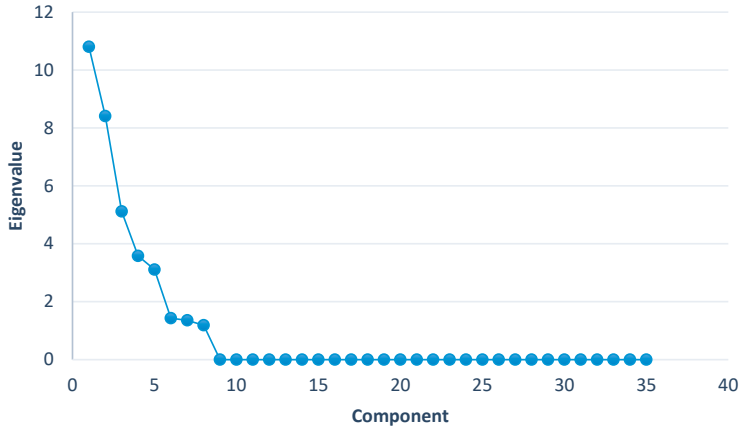


Figure 2. Scree plot of extracted components.

Following the application of PCA, selection of KPIs was performed, identifying those with the most significant impact on data variability within the first five principal components (PC1–PC5), which together accounted for 88.7% of the total variance. The selected variables were identified based on factor loadings, considering indicators with the highest absolute values, ensuring the retention of critical information necessary for further analysis. In the first principal component (PC1), the variables C1 (0.257), G5 (0.252), and D5 (−0.254) emerged as the most representative. These variables correspond to the degree of railway electrification, ATP (Automatic Train Protection) system coverage, and track failure frequency, highlighting their significant contribution to railway infrastructure management efficiency. The second principal component (PC2) was primarily influenced by the variable G6 (−0.315), which relates to the connectivity of the railway network with maritime ports. For the third component (PC3), the key indicators were C5 (−0.329), S3 (0.308), and S5 (0.306), which include the share of the national network managed by the RIM, safety indicators related to infrastructure management, and the share of electric trains in the total traffic. These indicators emphasize the relationship between safety aspects and energy efficiency within the railway system. Within the fourth component (PC4), the most significant indicator was C3 (0.404), which represents the modal share of rail transport in freight transport. This variable underscores the importance of freight transport as a critical efficiency factor in the railway network. The fifth component (PC5) identified S2 (0.359) and P4 (−0.53) as the most relevant KPIs. These variables are associated with the number of

serious accidents and the percentage of cancelled trains due to infrastructure-related issues, emphasizing the importance of safety and operational challenges in the railway sector.

Table 4. Factor loadings of KPIs.

Variables/KPIs	PC1	PC2	PC3	PC4	PC5
C1—Degree of electrification of total main track	0.257	0.012	0.143	0.121	0.196
C2—National modal share of railway in passenger transport	0.185	0.215	−0.032	0.018	0.118
C3—National modal share of railway in freight transport	0.048	0.183	0.084	0.404	0.08
C4—Length of national main lines	0.122	−0.17	−0.299	0.127	−0.049
C5—Share of national network managed % by RIM	0.07	−0.133	−0.329	0.175	0.004
C6—Length of high-speed lines	−0.201	−0.196	−0.015	−0.22	−0.133
C7—Active licences	0.074	−0.035	−0.32	0.234	−0.044
S1—Significant accidents	−0.078	−0.19	0.165	0.012	0.264
S2—Fatalities and serious injuries	−0.211	−0.005	0.04	0.068	0.359
S3—Infrastructure manager-related precursors to accidents	−0.045	−0.189	0.308	0.179	−0.046
S4—Share of diesel-powered trains	−0.207	0.011	−0.306	0.031	−0.031
S5—Share of electricity-powered trains	0.207	−0.011	0.306	−0.031	0.031
P1—Passenger train punctuality	0.188	0.097	−0.053	−0.311	−0.008
P2—Freight train punctuality	0.201	−0.067	0.206	−0.031	−0.079
P3—Delay minutes caused by infrastructure manager	−0.227	−0.014	0.021	−0.2	0.201
P4—Percentage of passenger train cancellations caused by RIM	−0.052	−0.066	0.059	−0.031	−0.53
D1—Asset failures in relation to network size	−0.221	0.2	−0.104	0.109	−0.004
D2—Signalling failures in relation to network size	−0.181	0.218	0.072	0.112	0.109
D3—Telecommunication failures in relation to network size	−0.227	0.219	−0.038	−0.027	−0.012
D4—Power supply failures in relation to network size	−0.235	0.076	0.116	0.175	0.023
D5—Track failures in relation to network size	−0.254	0.161	−0.06	−0.019	−0.028
D6—Tracks with temporary speed restrictions	0.191	−0.016	−0.134	0.267	−0.019
F1—Operational expenditures (OPEX) in relation to network size	0.065	0.299	−0.094	−0.125	−0.135
F2—Maintenance expenditures in relation to network size	−0.118	0.219	−0.125	−0.277	0.03
F3—Capital expenditures (CAPEX) in relation to network size	0.156	0.013	0.221	0.111	−0.274
F4—Proportion of Track Access Charges (TACs) in total revenue	0.134	−0.174	−0.166	−0.16	0.078
F5—Track Access Charge (TAC) revenue in relation to network size	0.177	−0.035	−0.301	−0.04	0.071
F6—Public funding for operational expenditures (OPEX)	−0.118	0.087	−0.028	0.21	−0.447
F7—Public funding for capital expenditures (CAPEX)	0.048	0.266	0.194	−0.15	−0.128
G1—Degree of network utilization—all trains	0.178	0.244	−0.05	−0.104	−0.066
G2—Degree of network utilization—passenger trains	0.171	0.226	−0.044	−0.186	−0.085
G3—Degree of network utilization—freight trains	0.134	0.216	−0.054	0.299	0.048
G4—ERTMS track-side deployment	0.15	0.269	0.097	0.05	0.136
G5—ATP (Automatic Train Protection) coverage	0.252	−0.01	−0.096	−0.163	0.118
G6—Core maritime port connection	0.039	−0.315	0.053	−0.066	−0.087

All the selected variables have high interpretative value, as they provide a more accurate assessment of the indicators influencing railway infrastructure efficiency. Furthermore, these indicators can serve as a foundation for optimizing existing operational strategies, and support informed decision-making in railway system management.

The further evaluation is based on ten KPIs, categorized into input and output variables. The input variables include the degree of main railway line electrification (C1), coverage of the Automatic Train Protection (ATP) system (G5), share of the national network managed by the RIM (C5), connectivity with maritime ports (G6), modal share of rail freight transport (C3), and track failures relative to network size (D5). On the other hand, the output variables consist of infrastructure manager-related safety indicators (S3), the share of electric trains in total traffic (S5), the number of serious accidents (S2), and the percentage of cancelled trains due to infrastructure-related issues (P4). These indicators serve as the foundation for assessing factor significance using the G-BWM and evaluating efficiency through the AR DEA method. In Step 3, the G-BWM was applied to determine the weight boundaries for inputs and outputs. The process began with the definition of the grey scale for indicator evaluation (Table 2), utilizing interval values derived from grey number theory. Once the scale was defined, experts identified the most important (Best) and least important (Worst) input and output, which acted as reference values for evaluating the remaining indicators. Based on these reference values, the remaining input and output variables were assessed relative to the best and worst indicators, forming grey vectors. The first vector represents the preference of the best indicator over the others, while the second vector reflects the preference of all indicators concerning the worst indicator. The obtained scores were then used in an optimization model, where formulas (5) to (12) were applied to compute the lower and upper weight boundaries for each input and output. This process also included a consistency check to ensure that the expert assessments were logically coherent and mutually aligned, thereby increasing the reliability and robustness of the weighting process. Table 5 presents the input evaluations obtained using the G-BWM, where inputs are assessed relative to the most important (Best) and least important (Worst) criteria, while the final column displays the corresponding grey weights. Based on the results, the degree of main railway line electrification (C1) was identified as the most important input, having the highest weight range. In contrast, the modal share of rail freight transport (C3), the share of the national network managed by the RIM (C5), and ATP system coverage (G5) were assigned approximately equal importance. The lowest weight was assigned to track failures relative to network size (D5), indicating that while this indicator remains relevant for analysis, it has the least impact on overall infrastructure efficiency.

Table 5. Input evaluations and grey weights of inputs.

Input KPI	Evaluation Relative to C1 (Best Input)	Evaluation Relative to D5 (Worst Input)	Grey Weights of Inputs
C1—Degree of electrification of total main track (Best input)	BI	FH [5, 7]	[0.412, 0.491]
C3—National modal share of railway in freight transport	M [4, 6]	VL [1, 3]	[0.098, 0.101]
C5—Share of national network managed by RIM (%)	M [4, 6]	VL [1, 3]	[0.098, 0.101]
D5—Track failures in relation to network size (Worst input)	FH [5, 7]	WI	[0.054, 0.078]
G5—ATP (Automatic Train Protection) coverage	M [4, 6]	VL [1, 3]	[0.098, 0.101]
G6—Core maritime port connection	L [2, 4]	FL [3, 5]	[0.152, 0.216]

Table 6 presents the output evaluations, where the output variables are assessed relative to the best (Best) and worst (Worst) output. The infrastructure manager-related safety indicator (S3) was identified as the most significant output, exhibiting the highest weight, while the number of serious accidents (S2) was recognized as the least significant output, with the lowest weight range. The share of electric trains (S5) and the percentage of cancelled trains due to infrastructure issues (P4) demonstrated moderate importance in the performance evaluation, with P4 having slightly lower weights compared to S5. These results indicate that in assessing the efficiency of railway infrastructure managers, the greatest emphasis is placed on safety indicators associated with infrastructure management.

Table 6. Output evaluations and grey weights of outputs.

Output KPI	Evaluation Relative to S3 (Best Output)	Evaluation Relative to S2 (Worst Output)	Grey Weights of Outputs
S2—Fatalities and serious injuries (Worst output)	M [4, 6]	WO	[0.078, 0.125]
S3—Infrastructure manager-related precursors to accidents (Best output)	BO	M [4, 6]	[0.475, 0.596]
S5—Share of electricity-powered trains	L [2, 4]	L [2, 4]	[0.181, 0.250]
P4—Percentage of passenger train cancellations caused by RIM	FL [3, 5]	VL [1, 3]	[0.145, 0.150]

After determining the input and output weights using the G-BWM, Step 4 applied the AR-DEA method to assess the efficiency of railway infrastructure managers (RIMs). This analysis was based on ten key performance indicators (KPIs), where C1, C3, C5, D5, G5, and G6 were defined as inputs, while S2, S3, S5, and P4 were designated as outputs. Based on the previous step, each input and output were assigned weight limits and optimized through the G-BWM.

In the first step of the AR-DEA method, input and output matrices for all decision-making units (DMUs) were formed, as defined in formulas (13) and (14). These matrices enabled the evaluation of each unit's efficiency based on the available data. Subsequently, the data were normalized to ensure comparability across all inputs and outputs, following formulas (15) and (16). Once normalization was completed, efficiency scores for each DMU were computed by solving an optimization problem, where the weighted sum of outputs was maximized under the condition that the weighted sum of inputs equaled 1, as shown in formulas (17) and (18). The predefined input and output weight limits obtained from the G-BWM were incorporated as constraints in the model, as represented in formulas (19) and (20). Finally, scaling factors were introduced to ensure that the weight limits remained within the defined intervals, as described in formulas (21) and (22). By solving this optimization problem, efficiency scores were determined for each DMU. Units with an efficiency score of 1 are considered fully efficient, while those with lower efficiency scores indicate areas for potential improvement. Table 7 presents the input and output values for the analyzed railway infrastructure managers, along with their efficiency scores calculated using the AR-DEA method. The table displays the six selected inputs (C1, C3, C5, D5, G5, G6) and four outputs (S2, S3, S5, P4), which serve as the basis for determining efficiency coefficients for each RIM.

The results indicate that RIM8 is the only infrastructure manager with an efficiency score of 1.000, making it the reference unit in the analysis, and demonstrating optimal resource utilization. The other RIMs exhibit varying levels of efficiency, with the lowest efficiency scores recorded for RIM2 (0.190) and RIM4 (0.222). Among the non-optimal units, RIM6 achieves the highest efficiency score (0.341), suggesting that it is approaching optimal

performance, but still has room for improvement. Analyzing the input and output values reveals that units with lower efficiency are often characterized by high values of negative outputs (S2 and P4) or inefficient use of inputs, such as a high number of track failures (D5) and suboptimal electrification utilization (C1). Conversely, efficient RIMs maintain a better balance between inputs and outputs, indicating more effective infrastructure management. These results enable the identification of potential improvements for less efficient units through resource optimization and performance enhancement of key indicators.

Table 7. Input/output values and efficiency.

DMU	C1	C3	C5	D5	G5	G6	S2	S3	S5	P4	Effic.
RIM1	66	15.3	4138	110	19	100	0.09	2.7	85	50	0.295
RIM2	78	19	55,198	1	98	100	0.1	0.16	70	18	0.190
RIM3	76	10.7	3225	145	78	100	0.54	1.75	80	4	0.261
RIM4	73	22	27,301	198	6	100	0.61	2.8	82	8	0.222
RIM5	71	6.4	5972	172	100	100	0.07	0.5	89	24	0.301
RIM6	100	33.6	7891	35	100	0	0.15	0.2	98	3	0.341
RIM7	70	10.6	47,870	8	79	87	0.12	0.9	80	22	0.222
RIM8	46	22.8	11,473	720	12	0	0.45	0.01	56	23	1.000
RIM9	86	28.8	11,834	145	85	100	0.1	3.7	95	25	0.227

Sensitivity Analysis

The sensitivity analysis presented in Table 8 evaluates the stability of the obtained efficiency results across different scenarios of weight coefficient modifications. In the first four scenarios (Sc.1–Sc.4), the weight of the most important input was gradually reduced by 25%, 50%, 75%, and 90%, while in the following four scenarios (Sc.5–Sc.8), the same modifications were applied to the most important output. In Sc.9, the weights of all outputs were equalized, whereas in Sc.10, the weights of both inputs and outputs were balanced to test the model's stability under completely uniform criteria. To enhance the clarity of the sensitivity analysis, the results are visually represented in Figure 3, which facilitates the identification of efficiency trends across different scenarios.

The results indicate that changes in the weights of inputs and outputs had an expected impact on efficiency scores, but without drastic fluctuations, confirming the stability of the solution. The reduction in the most important input's weight (Sc.1–Sc.4) led to a gradual decrease in efficiency for all RIMs, with the least efficient units experiencing the most significant drop. In Sc.4, where the weight of the key input was reduced by 90%, the efficiency of RIMs such as RIM2 and RIM7 significantly decreased, confirming that this indicator had a strong influence on the initial results. Conversely, in scenarios Sc.5–Sc.8, where the weight of the most important output was reduced, an increase in efficiency was observed for all RIMs. As expected, the least efficient units experienced the highest efficiency gains, as reducing the significance of the key output lowered their limiting constraints in the model. In Sc.8, where the weight of the key output was reduced by 90%, some RIMs (RIM6, RIM5) achieved efficiency scores close to 1, suggesting that this output was the main restricting factor in the baseline analysis. Sc.9 and Sc.10, in which all output weights and both input and output weights were equalized, demonstrated that efficiency values remained within an acceptable range, without significant deviations. This further confirms the robustness of the model, as the results do not exhibit extreme fluctuations, even when all factors are treated equally. The sensitivity analysis validates the stability and reliability of the obtained results, showing that changes in key input and output weights affect efficiency in the expected direction. More efficient RIMs displayed lower sensitivity to

weight modifications, while less efficient RIMs exhibited greater variations across different scenarios, indicating potential opportunities for optimizing their performance.

Table 8. Sensitivity analysis.

	Sc.0	**Sc.1**	**Sc.2**	**Sc.3**	**Sc.4**	**Sc.5**	**Sc.6**	**Sc.7**	**Sc.8**	**Sc.9**	**Sc.10**
RIM1	0.295	0.287	0.274	0.256	0.174	0.360	0.461	0.643	0.843	0.549	0.565
RIM2	0.190	0.177	0.161	0.139	0.090	0.227	0.284	0.386	0.501	0.344	0.244
RIM3	0.261	0.253	0.242	0.224	0.156	0.318	0.408	0.571	0.758	0.469	0.460
RIM4	0.222	0.214	0.203	0.186	0.129	0.271	0.347	0.486	0.641	0.371	0.357
RIM5	0.301	0.290	0.273	0.249	0.170	0.366	0.466	0.646	0.848	0.575	0.538
RIM6	0.341	0.341	0.342	0.343	0.311	0.411	0.520	0.717	0.957	0.629	0.627
RIM7	0.222	0.211	0.195	0.173	0.118	0.270	0.345	0.480	0.628	0.402	0.333
RIM8	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
RIM9	0.227	0.218	0.205	0.187	0.132	0.277	0.355	0.495	0.649	0.412	0.374

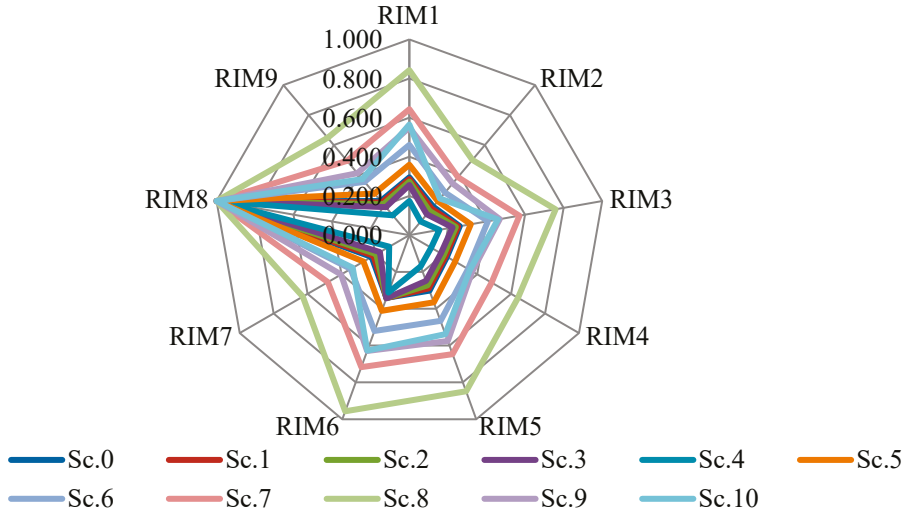


Figure 3. Sensitivity analysis.

5. Discussion

The application of a hybrid approach in this study, within the benchmarking analysis of RIMs, has enabled an objective assessment of their performance using a combination of PCA, G-BWM, and AR-DEA methods. This approach represents a significant advancement compared to traditional benchmarking methods, providing deeper insights into various aspects of railway infrastructure management and allowing for more precise strategic decision-making.

The proposed hybrid model is based on the DEA method, which identifies efficient DMUs using a set of KPIs as inputs and outputs. Compared to traditional performance evaluation approaches, DEA offers advantages such as handling multiple inputs and outputs simultaneously, not requiring predefined functional relationships, enabling direct comparisons between DMUs, and accommodating different units of measurement. However, conventional DEA has limitations, particularly in weight allocation, as it automatically assigns input and output weights without decision-maker (DM) influence. This can lead to extreme weighting scenarios in which a DMU appears efficient by emphasizing only one input and one output, while ignoring others. To address this, the AR DEA model introduces weight bounds, incorporating DM preferences and aligning the analysis with real-world conditions. Since defining these weight bounds relies on DM evaluations, a

Multi-Criteria Decision-Making (MCDM) method is required. This study employs the Best–Worst Method (BWM), due to its simplicity, higher consistency, and reduced comparison requirements compared to alternatives like AHP and ANP. Additionally, the BWM is applied within a grey environment, which enhances decision-making in cases where DM evaluations are imprecise or incomplete. The grey number structure, defined by upper and lower values, aligns well with the AR DEA model's need for weight constraints. Another drawback of conventional DEA is that the number of efficient DMUs tends to increase with more inputs and outputs, reducing discrimination power. To mitigate this, PCA is integrated into the model to identify the most critical KPIs, reducing dimensionality while preserving key information. PCA minimizes multicollinearity, enhances differentiation among DMUs, and improves computational efficiency. Thus, the hybrid model combines PCA for key input–output selection, the grey BWM for determining their priority and weight constraints, and AR DEA to derive truly efficient DMUs, ensuring a more realistic and effective evaluation framework.

In this specific case, a large number of KPIs are determined and monitored for the observed RIMs. Although at first glance, this seems like a positive thing, it makes it difficult to identify realistically effective RIMs. One of the reasons for this is that there is significant correlation and overlap between individual KPIs, and the other is that the existing and most commonly used methods for determining efficiency cannot adequately take into account a large number of inputs and outputs. In addition, there is no information on how important KPIs are in assessing the effectiveness of RIMs. The proposed model overcomes these problems by enabling the identification of key KPIs for the observed RIMs. Based on the factor loadings, which indicate the strength of the relationship between individual KPIs and principal components, PCA allowed a reduction in the problem size. At the same time, the AR DEA method identified effective RIMs, based on the limits (weight, importance) of the KPIs (input/output) obtained by applying the grey BWM for evaluation by DMs based on their experience and real-life circumstances. This made it possible to single out as efficient RIMs those that have high performance in the business segments that are truly important to the participants in these processes.

The results indicate that the most critical KPIs relate to the degree of electrification, ATP system coverage, and track failure rates. The degree of electrification directly influences energy efficiency, operational costs, and environmental sustainability, with higher electrification levels reducing reliance on fossil fuels and lowering carbon emissions. ATP system coverage enhances railway safety by preventing accidents caused by human error, ensuring compliance with speed limits, and reducing the risk of collisions, which is vital for maintaining operational reliability. Track failure rates serve as a key measure of infrastructure integrity, as frequent failures can lead to increased maintenance costs, service disruptions, and safety hazards, directly impacting network reliability. This underscores the essential role of infrastructure resources and safety measures in achieving effective railway infrastructure management. The results show that more efficient railway infrastructure managers exhibit a better balance between infrastructure resources and safety indicators, while less efficient RIMs are often burdened by a higher number of failures and lower coverage of advanced safety systems. A detailed analysis of the results reveals that RIM8 is the only fully efficient manager, with an efficiency score of 1.000, meaning it optimally utilizes available resources and can serve as a benchmark in the analysis. Other RIMs exhibit varying levels of efficiency, with RIM2 and RIM4 showing the lowest efficiencies of 0.190 and 0.222, respectively, indicating significant room for improvement. The efficiency results are primarily driven by the weighting of KPIs using the G-BWM and their values in the AR-DEA model. The proportion of the national network managed by

a railway infrastructure manager affects strategic planning and investment efficiency, as larger networks require advanced asset management strategies to optimize performance and minimize operational risks. The degree of electrification of the total main track (C1) and infrastructure manager-related precursors to accidents (S3) had the highest weights, making them decisive factors. Precursors to accidents managed by infrastructure operators serve as early warning indicators, enabling proactive risk management and targeted interventions to enhance railway safety and prevent major disruptions. A higher share of electric trains in total traffic is critical for achieving sustainability goals, as electric traction significantly reduces air pollution, lowers energy costs, and enhances network efficiency. RIM2's inefficiency stems from its low value in infrastructure manager-related precursors to accidents (S3), which carries the highest weight. RIM8, despite having the highest track failures relative to network size (D5), achieved full efficiency due to strong performance in infrastructure manager-related precursors to accidents (S3), which had a greater influence in the AR-DEA model. Conversely, RIM4's inefficiency is driven by its high number of fatalities and serious injuries (S2) and infrastructure manager-related precursors to accidents (S3), both of which significantly impact efficiency due to their high weights. The number of fatalities and serious injuries is a fundamental safety KPI, as reducing these incidents is crucial for maintaining public trust, regulatory compliance, and the overall reliability of railway operations. High rates of passenger train cancellations due to infrastructure failures indicate operational inefficiencies and service reliability challenges, which can negatively impact customer satisfaction and financial performance. Strong connectivity with maritime ports is essential for intermodal transport efficiency, as it facilitates seamless freight transfers, reduces logistics costs, and enhances the competitiveness of rail freight networks. A higher modal share of rail freight transport indicates a well-utilized railway network, contributing to economic efficiency and sustainability by reducing road congestion, fuel consumption, and greenhouse gas emissions. These findings highlight that while infrastructure resources influence efficiency, safety, and operational stability, they are the most critical factors in performance evaluation.

The conducted sensitivity analysis confirmed the stability of the model, where the reduction in the weight of the key input was gradually reflected in lower efficiency scores, while the reduction in the most important output weight led to an increase in efficiency. Notably, less efficient RIMs were more sensitive to weight changes, suggesting that their performance is more dependent on specific infrastructure factors. In scenarios where input and output weights were equalized, efficiency scores remained within an acceptable range, further confirming the robustness of the approach.

The findings of this paper provide valuable insights that can assist infrastructure managers in making informed decisions, optimizing operational strategies, and defining future investments. Identified areas for improvement allow for targeted actions that can enhance the efficiency of the railway sector as a whole. Additionally, benchmarking analysis enables comparative evaluation, ensuring transparency and fostering competitiveness among infrastructure managers. Despite the significant advantages of the developed model, certain challenges and limitations remain. The reliability of results depends on the availability of accurate and up-to-date data, while subjectivity in expert judgement may affect the precision of weight coefficients in the G-BWM. Furthermore, the complexity of the AR-DEA method requires careful interpretation of results to ensure their practical applicability.

The hybrid model presented in this study could be adapted for other transport sectors, such as road and air transport, by incorporating sector-specific KPIs and adjusting the weighting process to reflect industry-specific efficiency drivers. For example, in road transport, key indicators could include congestion levels, vehicle emissions, and infrastruc-

ture maintenance costs, while in air transport, factors such as runway utilization, on-time performance, and the environmental impact of airport operations could be integrated. Additionally, the model's structured approach to efficiency evaluation could serve as a foundation for transport policy-making, offering data-driven insights to regulatory bodies for optimizing resource allocation, setting performance benchmarks, and promoting sustainable infrastructure development.

One of the key challenges in collecting and validating data for the evaluation of RIMs is the inaccessibility and inconsistency of data across different countries. The PRIME platform, an initiative of the European Commission for collecting data from RIMs, plays a crucial role in standardizing and making information available. However, data submission to the PRIME platform is not mandatory, leading to issues with data incompleteness. Many RIMs lack unified or standardized performance tracking systems, and some do not monitor all KPIs, which further complicates the benchmarking analysis. Additionally, a key limitation of this study is its exclusive focus on RIMs within the EU, where regulatory frameworks, operational standards, and data availability exhibit a relatively high degree of uniformity. Nevertheless, the proposed model and methodology can be adapted for application to RIMs outside Europe, provided that necessary modifications are made to accommodate distinct regulatory landscapes, operational complexities, and data accessibility constraints. As previously discussed, data unavailability poses a significant challenge even within the EU, and this issue becomes more pronounced in non-EU contexts, due to inconsistencies in data collection methodologies and KPI reporting standards. While the PRIME platform facilitates a certain level of data standardization among European RIMs, the non-mandatory nature of data submission results in gaps that impede direct comparability across different railway infrastructure networks.

Future research could refine benchmarking models by incorporating additional KPIs, such as infrastructure resilience metrics (e.g., track maintenance backlog, climate-related service disruptions) or digitalization indicators (e.g., level of automated train control, real-time monitoring adoption). Moreover, examining external variables like regulatory policy changes, economic shifts, and emerging technologies (e.g., AI-driven asset management) would enhance the adaptability and robustness of efficiency assessments in railway infrastructure management. The proposed hybrid model confirms its usefulness in the railway infrastructure sector, enabling managers to enhance their performance through objective and quantitative insights, thereby contributing to sustainable development and more efficient resource management.

6. Conclusions

The performance of RIMs is a key factor in ensuring the efficiency, reliability, and sustainability of the railway sector. As rail networks face increasing demands for capacity expansion and service optimization, ensuring efficient infrastructure management is essential to achieving long-term competitiveness and sustainability in the railway sector. Given the complexity of infrastructure operations and the necessity for data-driven decision-making, benchmarking provides a valuable framework for evaluating performance, identifying best practices, and guiding strategic improvements. Benchmarking analysis, as a methodological framework for comparative performance assessment, enables the identification of best practices, the determination of reference values, and the recognition of areas that require improvement. Through systematic performance comparison among different RIMs, benchmarking provides objective insights that support informed decision-making and the optimization of operational processes to enhance the sector's competitiveness and efficiency. This study has developed an integrated framework for

evaluating the efficiency of railway infrastructure managers, combining PCA, the G-BWM, and AR-DEA. The proposed approach enables the identification of key performance factors, the determination of the relative importance of selected indicators, and an objective assessment of the efficiency of RIMs. The research results highlight that infrastructure and safety factors are crucial for the efficiency of railway infrastructure management. From an initial set of 35 KPIs, the PCA method identified six key inputs (C1, C3, C5, D5, G5, G6) and four outputs (S2, S3, S5, P4), covering the most relevant aspects of infrastructure, safety, and operational performance. The application of the G-BWM identified the degree of electrification (C1) as the most important input, while safety indicators (S3) and the share of electric trains (S5) were recognized as the most significant outputs. The AR-DEA enabled the assessment of the relative efficiency of infrastructure managers, with RIM8 identified as the only fully efficient unit, while other RIMs demonstrated varying levels of efficiency. RIM2 and RIM4 recorded the lowest efficiency scores, indicating substantial room for improvement. The sensitivity analysis confirmed the robustness of the model, showing that less efficient RIMs were more sensitive to changes in input and output weights, whereas more efficient RIMs demonstrated greater resilience to variations in indicator weights.

The key contributions of the study are the following:

- Development of a Hybrid Benchmarking Framework: A novel methodological framework integrating PCA, the G-BWM, and AR-DEA for systematically assessing the performance of railway infrastructure managers (RIMs).
- Reliable Dimensionality Reduction with PCA: PCA effectively reduces data dimensionality, identifying the most relevant performance indicators while eliminating redundancy and multicollinearity.
- Objective Weight Determination Using the G-BWM: The Grey Best–Worst Method (G-BWM) ensures a structured and objective approach to weighting KPIs, even in conditions of uncertainty and incomplete information.
- Enhanced Efficiency Measurement with AR-DEA: The Assurance Region Data Envelopment Analysis (AR-DEA) method improves efficiency evaluation by incorporating realistic operational constraints and decision-maker preferences.
- Practical Guidelines for Performance Improvement: The study identifies key efficiency gaps and provides actionable recommendations for infrastructure managers to optimize resource allocation, improve maintenance strategies, and enhance overall network performance.
- Support for Strategic Decision-Making: The model offers a data-driven tool for policymakers and infrastructure managers to develop investment strategies, optimize operational performance, and ensure long-term railway sector sustainability.

The practical implications of this study provide actionable insights that RIMs can implement to enhance efficiency and optimize operations. For instance, RIMs with lower efficiency scores, such as RIM2 and RIM4, can prioritize targeted investments in infrastructure upgrades, particularly by increasing network electrification and enhancing accident prevention measures, both of which were identified as key efficiency drivers. Additionally, by leveraging the benchmarking results, underperforming RIMs can adopt best practices from high-performing counterparts, such as RIM8, which demonstrated optimal resource utilization and safety management. For maintenance optimization, infrastructure managers can use the findings to develop predictive maintenance programmes, reducing downtime and infrastructure-related delays. Moreover, regulatory bodies can utilize this model to allocate funding based on efficiency gaps, directing resources toward RIMs that require urgent improvements in operational reliability and safety. By integrating these insights into long-

term strategic planning, RIMs can systematically improve network performance, enhance service quality, and contribute to a more sustainable and competitive railway sector.

Future research should explore the impact of various external factors that directly influence efficiency and performance, including economic, regulatory, and technological changes. For example, evolving EU regulations on railway liberalization, infrastructure funding, and safety standards can significantly alter how resources are allocated and performance is measured. Technological innovations—such as IoT-based monitoring, AI-driven asset management, and predictive maintenance—are also transforming operations by enabling more accurate and proactive decision-making. Moreover, the way in which infrastructure managers are organized—whether under state control, private management, or hybrid models—can impact investment priorities and operational effectiveness. Considering these external drivers will help to ensure that future benchmarking models remain robust and practical for improving performance in an ever-changing industry. Additionally, the model could be expanded by incorporating additional criteria and adapting it to specific national and regional conditions, including the consideration of RIMs outside of the EU, where differences in regulatory frameworks and data availability may require further methodological adjustments. The proposed framework demonstrates high potential for practical application, providing reliable and objective insights that can serve as a foundation for decision-making aimed at improving the efficiency of the railway sector and achieving sustainable transport infrastructure development.

Author Contributions: Conceptualization, A.K., S.T. and M.K.; methodology, M.K.; software, M.K.; validation, S.T. and M.K.; formal analysis, A.K., S.T., N.Č., A.M. and M.V.; investigation, A.K. and S.T.; data curation, A.K. and S.T.; writing—original draft preparation, A.K., S.T. and M.K.; writing—review and editing, S.T. and M.K.; visualization, A.K. and M.V.; supervision, S.T., N.Č. and A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All data used in the study are presented within the paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lema, N.M.; Price, A.D. Benchmarking: Performance Improvement Toward Competitive Advantage. *J. Manag. Eng.* **1995**, *11*, 28–37. [CrossRef]
2. Anand, G.; Kodali, R. Benchmarking the benchmarking models. *Benchmarking Int. J.* **2008**, *15*, 257–291. [CrossRef]
3. Ganguly, S.; Ray Chaudhury, S.; Mukherjee, A.; Si, A. Performance Analysis of Indian Railway Zones using MCDM Approaches. *Decis. Mak. Adv.* **2025**, *3*, 111–125. [CrossRef]
4. Lee, Y.; Zailani, S.; Soh, K.L. Understanding Factors for Benchmarking Adoption. *Benchmarking Int. J.* **2006**, *13*, 548–565.
5. Moffett, S.; Anderson-Gillespie, K.; McAdam, R. Benchmarking and Performance Measurement: A Statistical Analysis. *Benchmarking* **2008**, *15*, 368–381. [CrossRef]
6. Gus, M. Overcoming obstacles to developing effective performance measures. *Work. Study Emerald Group Publ. Ltd.* **1999**, *48*, 223–229. [CrossRef]
7. Yoshida, Y.; Fujimoto, H. Japanese-airport benchmarking with the DEA and endogenous-weight TFP methods: Testing the criticism of overinvestment in Japanese regional airports. *Transp. Res. Part E Logist. Transp. Rev.* **2004**, *40*, 533–546. [CrossRef]
8. Dias, J.; Azevedo, S.; Ferreira, J.J.; Palma, S. A comparative benchmarking analysis of main Iberian container terminals: A DEA approach. *Int. J. Shipp. Transp. Logist.* **2009**, *1*, 260–275. [CrossRef]
9. Egilmez, G.; McAvoy, D. Benchmarking road safety of U.S. states: A DEA-based Malmquist productivity index approach. *Accid. Anal. Prev.* **2013**, *53*, 55–64. [CrossRef] [PubMed]
10. Sharma, M.G.; Debnath, R.M.; Oloruntoba, R.; Sharma, S.M. Benchmarking of rail transport service performance through DEA for Indian railways. *Int. J. Logist. Manag.* **2016**, *27*, 629–649.

11. Hong, P.; Hong, S.; Roh, J.; Park, K. Evolving benchmarking practices: A review for research perspectives. *Benchmarking Int. J.* **2012**, *19*, 444–462. [CrossRef]
12. Blumenfeld, M.; Lin, C.Y.; Jack, A.; Abdurrahman, U.T.; Gerstein, T.; Barkan, C.P. Towards measuring national railways' safety through a benchmarking framework of transparency and published data. *Saf. Sci.* **2023**, *164*, 106188. [CrossRef]
13. Lin, C.Y.; Blumenfeld, M.; Gerstein, T.; Barkan, C.P.; Jack, A.; Abdurrahman, U.T. International benchmarking of railroad safety data systems and performance–a cross-continental case study. *J. Rail Transp. Plan. Manag.* **2023**, *26*, 100384. [CrossRef]
14. Rungskunroch, P.; Shen, Z.J.; Kaewunruen, S. Benchmarking environmental and economic impacts from the HSR networks considering life cycle perspectives. *Environ. Impact Assess. Rev.* **2021**, *90*, 106608. [CrossRef]
15. He, Q.; Wang, T.; Chan, A.P.; Xu, J. Developing a list of key performance indictors for benchmarking the success of construction megaprojects. *J. Constr. Eng. Manag.* **2021**, *147*, 04020164. [CrossRef]
16. Pugi, L.; Berzi, L.; Spedicato, M.; Cirillo, F. Hydrogen for railways: Design and simulation of an industrial benchmark study. *Int. J. Model. Identif. Control.* **2023**, *43*, 43–53. [CrossRef]
17. Zintel, M.; Pluchet, J.; Hensler, A.; Bamberger, V.; Baron, R.; Watson, S. *RAIL 2040 Scenarios and CEO Agenda: Report 2023*; Arthur D. Little: Brussels, Belgium, 2023. Available online: https://www.adlittle.com/sites/default/files/reports/ADL_Rail_2040_2023.pdf (accessed on 24 February 2025).
18. Platform of Rail Infrastructure Managers in Europe (PRIME). *2021 PRIME Benchmarking Report*; European Commission: Brussels, Belgium, 2021. Available online: https://wikis.ec.europa.eu/download/attachments/44167372/PRIME%20External%20Report%202021.pdf?version=2&modificationDate=1687506392630&api=v2 (accessed on 24 February 2025).
19. Yi, F.; Zhijiang, L. Factor Analysis of Railway Carrying Capacity Coordination Optimization Considering Energy Consumption. *J. Electr. Syst.* **2024**, *20*, 676–683.
20. Duleba, S.; Farkas, B. Principal Component Analysis of the Potential for Increased Rail Competitiveness in East-Central Europe. *Sustainability* **2019**, *11*, 4181. [CrossRef]
21. Braga, J.A.P.; Andrade, A.R. Multivariate statistical aggregation and dimensionality reduction techniques to improve monitoring and maintenance in railways: The wheelset component. *Reliab. Eng. Syst. Saf.* **2021**, *216*, 107932. [CrossRef]
22. Lajevardi, S.; Lourenco, P.; Sousa, H.; Matos, J. Railway reinforced concrete infrastructure life management and sustainability index. *Acta Polytech. CTU Proc.* **2022**, *33*, 316–321. [CrossRef]
23. Panja, S.; Patra, S.; Gangopadhyaya, M.; Poddar, S. Human factor analysis for railway coach and bogie maintenance using principal component analysis and factor analysis. *Int. J. Hum. Factors Ergon.* **2019**, *6*, 35. [CrossRef]
24. Kamran, G.; Dragan, P.; Sarbast, M.; Parastou, B.; Domokos, E.K.; Iraj, M. Decoding rail derailments: Unraveling the weighted factors influencing safety and sustainability using the best-worst method. *Results Eng.* **2024**, *23*, 102539. [CrossRef]
25. Wang, Y.; Liang, Y.; Li, C.; Zhang, X. Operation Performance Evaluation of Urban Rail Transit PPP Projects: Based on Best Worst Method and Large-Scale Group Evaluation Technology. *Adv. Civ. Eng.* **2021**, *2021*, 4318869. [CrossRef]
26. Noruzi, M.; Naderan, A.; Zakeri, J.A.; Rahimov, K. A Novel Decision-Making Framework to Evaluate Rail Transport Development Projects Considering Sustainability under Uncertainty. *Sustainability* **2023**, *15*, 13086. [CrossRef]
27. Binxin, Y.; Guangdong, W. Multi-criteria analysis of cross-regional railways interconnection under the post COVID-19 pandemic crisis: A hybrid BWM-FAISM-DFS evaluation framework. *Transp. Res. Part E Logist. Transp. Rev.* **2022**, *179*, 103316. [CrossRef]
28. Barasin, A.M.; Alqahtani, A.Y.; Makki, A.A. Performance Evaluation of Retail Warehouses: A Combined MCDM Approach Using G-BWM and RATMI. *Logistics* **2024**, *8*, 10. [CrossRef]
29. Krstić, M.; Tadić, S.; Agnusdei, L. Evaluating governance models in intermodal terminal operations: A hybrid grey MCDM approach. *J. Intell. Manag. Decis.* **2023**, *2*, 179–191. [CrossRef]
30. Krstić, M.; Elia, V.; Agnusdei, G.P.; De Leo, F.; Tadić, S.; Miglietta, P.P. Evaluation of the agri-food supply chain risks: The circular economy context. *Brit. Food J.* **2024**, *126*, 113–133. [CrossRef]
31. Tadić, S.; Krstić, M.; Kovač, M.; Brnjac, N. Evaluation of smart city logistics solutions. *Promet-Traffic Transp.* **2022**, *34*, 725–738. [CrossRef]
32. Niu, Y.; Li, X.; Zhang, J.; Deng, X.; Chang, Y. Efficiency of railway transport: A comparative analysis for 16 countries. *Transp. Policy* **2023**, *141*, 42–53. [CrossRef]
33. Li, W.; Hilmola, O.-P. Belt and Road Initiative and Railway Sector Efficiency—Application of Networked Benchmarking Analysis. *Sustainability* **2019**, *11*, 2070. [CrossRef]
34. Topcu, T.G.; Triantis, K.; Roets, B. Estimation of the workload boundary in socio-technical infrastructure management systems: The case of Belgian railroads. *Eur. J. Oper. Res.* **2019**, *278*, 314–329. [CrossRef]
35. Blagojević, A.; Vesković, S.; Kasalica, S.; Gojić, A.; Allamani, A. The application of the fuzzy AHP and DEA for measuring the efficiency of freight transport railway undertakings. *Oper. Res. Eng. Sci. Theory Appl.* **2020**, *3*, 1–23. [CrossRef]

36. Tadić, S.; Krstić, M.; Brnjac, N. Selection of efficient types of inland intermodal terminals. *J. Transp. Geogr.* **2019**, *78*, 170–180. [CrossRef]
37. Foroutan, S.L.F.; Bamdad, S. Efficiency measurement of railway passenger stations through network data envelopment analysis. *Res. Transp. Bus. Manag.* **2023**, *46*, 100767. [CrossRef]
38. Tang, G.; Zhang, M.; Bu, F. Vehicle Environmental Efficiency Evaluation in Different Regions in China: A Combination of the Life Cycle Analysis (LCA) and Two-Stage Data Envelopment Analysis (DEA) Methods. *Sustainability* **2023**, *15*, 11984. [CrossRef]
39. Marchetti, D.; Wanke, P.F. Efficiency in rail transport: Evaluation of the main drivers through meta-analysis with resampling. *Transp. Res. Part A Policy Pract.* **2019**, *120*, 83–100. [CrossRef]
40. Labib, A.; Anson, J. Benchmarking the sustainability reporting of high-speed railways (HSRs): Towards a state-of-the-art benchmarking and reporting framework for HSRs. *J. Clean. Prod.* **2020**, *250*, 119505. [CrossRef]
41. Stević, Ž.; Miškić, S.; Vojinović, D.; Huskanović, E.; Stanković, M.; Pamučar, D. Development of a Model for Evaluating the Efficiency of Transport Companies: PCA–DEA–MCDM Model. *Axioms* **2020**, *11*, 140. [CrossRef]
42. Muge, A.K.; Jia, Q.; Wang, Y.; Li, F.T. Exploring the Eco-Efficiency of Transport-Related Particulate Matter Pollution in Nairobi City, Kenya. *Open Access Libr. J.* **2020**, *7*, e7067. [CrossRef]
43. Omrani, H.; Amini, M.; Alizadeh, A. An integrated group best-worst method—Data envelopment analysis approach for evaluating road safety: A case of Iran. Meas. *J. Int. Meas. Confed.* **2020**, *152*, 107330. [CrossRef]
44. Anand, K.; Gurmeet, S.; Omkarprasad, S.V. A Comparative Evaluation of Public Road Transportation Systems in India Using Multicriteria Decision-Making Techniques. *J. Adv. Transp.* **2020**, *2020*, 8827186. [CrossRef]
45. Ramaganesh, M.; Rebekka, S.; Watson, V.; Subramanian, A.; Sankaranarayanan, B. Measurement of HOSPITAL efficiency in biomedical waste management. *Mater. Today Proc.* **2021**. [CrossRef]
46. Taghipour, F.; Hamid, M.; Aghakarimi, E.; Rabbani, M. An integrated framework to evaluate and improve the performance of emergency departments during the COVID-19 pandemic: A mathematical programing approach. *Proc. Inst. Mech. Eng. Part H J. Eng. Med.* **2023**, *237*, 683–705. [CrossRef]
47. Brandenburg, M.; Hahn, G.J. *Financial Performance and Firm Efficiency of Automotive Manufacturers and Their Suppliers: A Longitudinal Data Envelopment Analysis*; Logistics Research; Bundesvereinigung Logistik (BVL): Bremen, Germany, 2021; Volume 14, pp. 1–26, ISSN 1865-0368. [CrossRef]
48. Kok, F.S.; Yongli, G.; Ming, M.Y. Enhancing logistics performance measurement: An effectiveness-based hierarchical data envelopment analysis approach. *INFOR Inf. Syst. Oper. Res.* **2024**, *62*, 449–479. [CrossRef]
49. Fumin, D.; Lin, X.; Yuan, F.; Qunxi, G.; Zhi, L. PCA-DEA-tobit regression assessment with carbon emission constraints of China's logistics industry. *J. Clean. Prod.* **2020**, *271*, 122548. [CrossRef]
50. Karami, S.; Ghasemy, Y.R.; Mousazadegan, F. Supplier selection and evaluation in the garment supply chain: An integrated DEA–PCA–VIKOR approach. *J. Text. Inst.* **2021**, *112*, 578–595. [CrossRef]
51. Adel, H.M.; Siavash, H.; Per, J.A. A strategy-based framework for supplier selection: A grey PCA-DEA approach. *Oper. Res.* **2020**, *22*, 263–297.
52. Wu, T.-H.; Chung, Y.-F.; Huang, S.-W. Evaluating global energy security performances using an integrated PCA/DEA-AR technique. *Sustain. Energy Technol. Assess.* **2021**, *45*, 101041. [CrossRef]
53. González-Gil, A.; Palacin, R.; Batty, P. Optimal energy management of urban rail systems: Key performance indicators. *Energy Convers. Manag.* **2015**, *90*, 282–291. [CrossRef]
54. Menglei, L. *Evaluation of Railway Performance Through Quality of Service*; College of Engineering and Physical Sciences—University of Birmingham: Birmingham, UK, 2016.
55. Siripong, P.; Wasaporn, T.; Waranon, D. A comparative study of performance measurement standards of railway operator. In Proceedings of the MATEC Web of Conferences, Sibiu, Romania, 7–9 June 2017; Volume 138, pp. 1–10. [CrossRef]
56. Fraszczyk, A.; Lamb, T.; Marinov, M. Are railways really that bad? An evaluation of rail systems performance in Europe with a focus on passenger rail. *Transp. Res. Part A Policy Pract.* **2016**, *94*, 573–591. [CrossRef]
57. Rungskunroch, P. Benchmarking Operation Readiness of the High-Speed Rail (Hsr) Network. Ph.D. Thesis, University of Birmingham, Birmingham, UK, 2022.
58. Kalem, A.; Tadić, S.; Krstić, M.; Čabrić, N.; Branković, N. Performance Evaluation of Railway Infrastructure Managers: A Novel Hybrid Fuzzy MCDM Model. *Mathematics* **2024**, *12*, 1590. [CrossRef]
59. Platform of Rail Infrastructure Managers in Europe. *Key Performance Indicators for Performance Benchmarking*; European Commission: Brussels, Belgium, 2022.
60. Hansen, I.A.; Wiggenraad, P.B.L.; Wolff, J.W. Benchmark Analysis of Railway Networks and Undertakings. In Proceedings of the International Seminar on Railway Operations Modelling and Analysis, Copenhagen, Denmark, 13–15 May 2013; pp. 1–20.

61. Cao, L.J.; Chua, K.S.; Chong, W.K.; Lee, H.; Gu, Q.M. A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine. *Neurocomputing* **2003**, *55*, 321–336. [CrossRef]
62. Brint, A.; Genovese, A.; Piccolo, C.; Taboada-Perez, G.J. Reducing Data Requirements When Selecting Key Performance Indicators for Supply Chain Management: The Case of a Multinational Automotive Component Manufacturer. *Int. J. Prod. Econ.* **2021**, *233*, 107967. [CrossRef]
63. Chae, S.C.; Choi, S.-Y. Analysis of the Term Structure of Major Currencies Using Principal Component Analysis and Autoencoders. *Axioms* **2022**, *11*, 135. [CrossRef]
64. Krstić, M.; Tadić, S.; Miglietta, P.P.; Porrini, D. Biodiversity Protection Practices in Supply Chain Management: A Novel Hybrid Grey Best–Worst Method/Axial Distance-Based Aggregated Measurement Multi-Criteria Decision-Making Model. *Appl. Sci.* **2025**, *15*, 1354. [CrossRef]
65. Podinovski, V.V. Side effects of absolute weight bounds in DEA models. *Eur. J. Oper. Res.* **1999**, *115*, 583–595. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Modeling the Relationship Between Radon Anomalies and Seismic Activity Using Artificial Neural Networks and Statistical Methods

Kostadin Yotov, Emil Hadzhikolev and Stanka Hadzhikoleva *

Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 24, Tzar Asen Str., 4000 Plovdiv, Bulgaria; kostadin_yotov@uni-plovdiv.bg (K.Y.); hadjikolev@uni-plovdiv.bg (E.H.)
* Correspondence: stankah@uni-plovdiv.bg

Abstract: The paper presents an approach for detecting anomalies in radon concentration in seismically active areas. It involves training multiple artificial neural networks (ANNs) to predict radon concentration during periods without seismic events. The trained ANNs model the typical radon variations under non-seismic conditions, and the predicted values for normal radon behavior are compared with actual radon concentrations around the time of recorded earthquakes. Significant deviations from the predicted values are interpreted as radon anomalies potentially associated with upcoming seismic events. The methodology includes wavelet transformation for noise removal, a multilayer ANN trained using the Levenberg–Marquardt algorithm, and a segmentation approach based on radial zones (annuli) for localized predictions. Large datasets from three radon measurement stations in Bulgaria—Yambol, Dimitrovgrad, and Krupnik—were used. Data from seismic periods were excluded during the training of the neural networks to ensure that the models learn only the natural radon variations under non-seismic conditions. Key results indicate that, in Yambol and Dimitrovgrad, the actual radon concentration exceeds the predicted normal levels during earthquakes, whereas in Krupnik, radon concentration is lower than expected during seismic events. Analysis of the pre-seismic period shows elevated radon levels 48 h before earthquakes at some stations, while expected anomalies were not observed at others. Through this study, we demonstrate the effectiveness of ANN models in modeling radon behavior under non-seismic conditions and identifying deviations that may be linked to seismic activity. We believe that the obtained results contribute to the ongoing discussion on radon concentration anomalies as potential earthquake precursors and suggest that local geological and environmental factors may further influence radon emissions in different ways.

Keywords: artificial neural networks; computational intelligence models; earthquake; soil radon; seismic events

MSC: 68T01

1. Introduction

Earthquakes cause significant damage to people and the environment. Their prediction is the subject of extensive scientific research. Timely warning of the population can enable evacuation and save human lives. People can take measures to protect their property and minimize potential damage. Early notification of civil protection and emergency

services provides time for preliminary preparation, better coordination, and more effective response by those involved in these activities. For this reason, many scientists study various phenomena and processes as potential precursors of seismic activity and seek methods for earthquake prediction.

For centuries, unusual animal behavior has been reported before critical events such as earthquakes. Observations include decreased activity among mammals and birds up to three weeks before an earthquake [1], strange animal behavior without vocalizations, and cases of various animals giving birth days to almost a month earlier prior to major earthquakes. Other studies have documented unusual changes in the animal kingdom days and hours before major earthquakes, including anxiety, chaotic movements, or even mass migrations [2–4]. Animals are particularly sensitive to physical and chemical changes in the environment. Before an earthquake, electromagnetic anomalies, changes in atmospheric pressure, alterations in groundwater, and other phenomena are observed. Such unexpected changes and anomalies are being investigated and studied as potential precursors to earthquakes.

In [5], geophysical and astronomical factors such as Earth–Moon gravitational forces, their varying distances, and localized gravity fluctuations are utilized to develop machine learning regression models that enhance earthquake predictions. Hypotheses on the relationship between space weather and earthquakes, as well as the potential triggering of seismic activity by strong solar flares, have also been investigated [6]. Other studies indicate a correlation between tides and earthquakes, suggesting that tidal forces could be used as a predictor for potential seismic events [7].

During the development of a seismic process, disturbances in the geomagnetic field are observed. Volvach et al. utilized magnetic field data to identify changes in geophysical fields associated with earthquakes of magnitude greater than 7. They found that the parametric resonance of surface geomagnetic oscillations could be considered a precursor to earthquakes [8].

How the ground shakes and the damage an earthquake can cause depend on the properties of the soil and rocks beneath it. One of the most important indicators of this is the so-called wave velocity (Vs), which shows how quickly seismic waves travel through soil or rock. Comina et al. used a combination of geological modeling and geophysical measurements to create a database of shear wave velocity values for the Piedmont region in Italy [9]. Such information can be used for more accurate predictions of seismic motions in the region; however, specific geological characteristics limit its universal application to other areas. Rău et al. developed a model to study how seismic waves (P and S waves) propagate through Earth's layers. This enhances the understanding of seismic activity in earthquake-prone zones and helps more accurately determine the location and characteristics of earthquakes [10]. Ji et al. investigated the mechanisms and environment of earthquake swarms by analyzing data on seismic wave propagation through underground layers. They created three-dimensional models of P-wave velocity (waves that propagate first), S-wave velocity (slower but more destructive waves), and the ratio between them (Vp/Vs). These models revealed that, beneath the highly seismically active area of Changdao, there are significant anomalies with high Vp/Vs values reaching depths of up to 30 km [11].

A thorough review of the scientific literature reveals that earthquake prediction is a highly complex process. Researchers have studied a wide variety of potential precursors, including seismic, geodetic, geochemical, geological, atmospheric, geomagnetic, electrical, and others. On the other hand, the unique conditions in each seismic region also play a significant role in how an earthquake develops. These include region-specific characteristics

such as soil type, terrain slope, fault structure, and the thickness of sedimentary layers. Human activities, such as construction, changes in urban infrastructure, mining, and the development of water facilities, can also influence the way seismic waves propagate. The specific characteristics of different regions motivate the development of individual mathematical models for earthquake prediction tailored to each specific area. It is not necessary to examine all possible environmental characteristics as predictors. Instead, it is sufficient to identify and analyze the factors that have the strongest correlation with seismic activity in the particular region.

Radon in groundwater and soil is considered one of the most significant geochemical precursors, used to detect chemical and physical changes preceding earthquakes. Stress in the Earth's crust causes the release of radon, leading to changes in its concentration in soil and groundwater. Therefore, this gas can be a valuable predictor in regions where faults are active and strong earthquakes are anticipated. Radon as a precursor to earthquakes has been specifically studied and utilized for developing a model in our research.

The approach we adopt for this study involves the use of ANNs trained with the Levenberg–Marquardt algorithm, as well as the segmentation of data into multidimensional layers (annuli) to improve the accuracy of predictions. Our choice of ANNs is based on several key characteristics of these structures that are important for our research. ANNs are particularly effective in predicting complex non-linear relationships, such as those between seismic activity and radon concentration. Furthermore, ANNs are capable of adapting to different stations by identifying patterns in the data that are difficult to detect using classical statistical methods. ANNs perform well with large and high-dimensional datasets, which are the focus of our study. One of the most important reasons for choosing ANNs was the noise characteristics in the dataset we worked with. These data were obtained from stationary monitoring stations in three different cities in Bulgaria. The measuring instruments at each of these stations, along with geological and climatic characteristics, as well as random factors, introduce a certain amount of noise into the data. ANNs do not directly memorize the input data; instead, they extract patterns and build representations of them. When the network is properly regularized, it does not "overfit" to the specific noise in the training dataset but rather focuses on the essential features in the data. Additionally, ANNs transform input data through multiple layers and activation functions, which helps smooth out non-systematic variations (noise). The hyperbolic tangent function (tanh) used in our approach acts as a limiting function, reducing the influence of extremely high or low values, which are often artifacts of noise. In the study, we applied an approach in which ANNs operate with different rings (annuli) of the input data, with each ANN being trained in a separate region. This allowed us to create local models that are more accurate for their respective zones and do not become "defocused" by random fluctuations outside their range. This approach reduced the impact of noisy data in different areas by dividing the data processing into independent subspaces. And although we also use wavelet transform for noise removal, the choice of ANNs, along with all their other advantages, further minimized the effect of noise in the data.

For the first time in Bulgaria, we employ a multitude of ANNs to analyze and approximate the typical behavior of radon in the soil at several geolocations across the country. We then compare these data with periods when seismic activity is recorded. This enables us to evaluate the impact of earthquakes on radon concentrations in the respective regions.

To address all these objectives and challenges, we structured the paper in a way that follows the stages of the conducted research. The study's context (in our case, radon as a predictor), the methodology involving statistical analysis and ANNs, the results, and the discussion are presented sequentially. Thus, Section 2 presents the conventional sci-

entific perspective on radon and its role in earthquake prediction, while Sections 3–5 introduce the innovative methods—statistical analyses, neural networks, and the segmentation approach. We believe that this structure clearly distinguishes the traditional from the novel approach in the study and facilitates the reader's understanding by gradually building comprehension of the research problem and its solution. More specifically, the paper follows this sequence: Section 2 "Radon as an Earthquake Predictor" provides a detailed discussion of radon as a potential earthquake predictor, summarizing previous studies and their significance for seismic forecasting. Section 3 "Statistical Study of the Data Obtained from Radon Stations" describes the statistical methods applied in the study, including preprocessing of the data, exploratory analysis, and key statistical indicators. Section 4 "Use of ANNs in Noisy Data with Large Sample Sizes" focuses on the application of ANNs for modeling radon concentration, particularly in the presence of noise in the data. The methodology for training, validation, and testing of the ANN is described in detail. Section 5 "Model for Predicting Radon Levels through the Construction of Multiple ANNs" presents the predictive model for radon anomalies, addressing segmentation of the input space, wavelet transformation for noise removal, and interpolation techniques used. Section 6 "Experiments" contains the results and discussion, emphasizing key findings and model performance. The paper concludes with Section 7 "Conclusions" which summarizes the main contributions and outlines directions for future research.

2. Radon as an Earthquake Predictor

Traditional methods for analyzing and describing radon concentrations include regression analysis [12–14] and the use of time series [15–17]. Numerical methods and polynomial approximations are also commonly employed [18–20]. Many approaches based on machine learning methods have also gained prominence and are widely used by researchers studying radon levels. Notable works include those presented by [21–23], among others. Recently, many authors have focused their attention on the field of artificial intelligence, represented by ANNs. Examples of such works include [24–26]. The interest in these artificial intelligence tools is entirely understandable—they are powerful approximation tools, especially when dealing with highly non-linear problems. Considering their adaptive nature and ability to continuously learn, it is clear why they are favored for approximation tasks characterized by strong indeterminacy and stochasticity. Additionally, their excellent compatibility with classical methods further enhances their appeal.

The authors of [27] utilize intelligent algorithms such as ANNs, multiple linear regression (MLR), and decision trees (DT) to detect radon anomalies caused by seismic activity. The dataset is divided into two categories—periods of seismic activity and stable periods—using time intervals of 7 days before and after the earthquake. Data from the stable periods are used for training and cross-validation of the intelligent algorithms. Statistically significant radon anomalies are identified through the difference between simulated and measured radon levels. The authors find that all the algorithms show significant deviations in radon concentration around the time of the earthquake, with the ANN model demonstrating the best performance.

To identify radon anomalies in groundwater, Feng et al. developed three models: the long short-term memory (LSTM) model with auxiliary data, the empirical mode decomposition long short-term memory (EMD-LSTM) model with auxiliary data, and the EMD-LSTM model without auxiliary data. They examined different time windows around earthquakes, including ± 10, ± 30, ± 50, and ± 100 days before and after seismic events. The highest prediction accuracy was achieved for the ± 30-day period. The EMD-LSTM model demonstrated the best predictive capability, identifying five potential radon anomalies

associated with seven earthquakes [28]. The applicability of LSTM models for detecting potential hydrological anomalies preceding earthquakes has also been confirmed in other studies [29].

Investigating the relationship between soil radon levels and seismic activity, Tareena et al. developed three models using a feed-forward neural network (FFNN), recurrent neural network (RNN), and support vector regression (SVR). They utilized data from a fault line and searched for anomalies in radon time series caused by seismic events. To achieve this, they compared actual and calculated radon concentrations while accounting for various meteorological and statistical parameters. Their analyses demonstrated that, with the application of machine learning techniques, anomalies in radon time series caused by earthquakes can be successfully identified, provided the environment is carefully characterized, and noise is excluded. The best results for radon assessment were achieved using FFNN, while SVR performed the weakest [30].

Shepelev et al. utilized machine learning methods to analyze the relationships between changes in radon concentration, seismic activity, and meteorological data. Radon variations are described using temporal, statistical, and complexity features, which are used to train a linear classification method to classify seismic activity as either "strong" or "weak". Two classification models are investigated, differing in how they classified seismic events as "strong" or "weak"—one based on the magnitude of events and the other on intensity in points. The researchers found that changes in subsurface radon concentration could serve as an implicit precursor to earthquakes [31].

In [32], the interrelationship, interdependence, and connectivity of various factors with earthquakes are examined, including water level, radon gas, animal behavior, hydrochemical, and dilatancy–diffusion. The authors analyze time series using various machine learning algorithms, including SVM, regression, decision tree, Bayesian model, and ANN models.

Planinić et al. used soil radon data collected over 4 years with LR-115 nuclear track detectors. They studied the influence of meteorological parameters on temporal variations in radon levels while also observing seismic activities. They determined the linear equation of multiple regression, which allows for the reduction of radon variations caused by barometric pressure, rainfall, and air temperature. Based on radon anomalies, the authors identified earthquakes with magnitudes greater than 3 at epicentral distances of less than 200 km [33].

In [34], a study of soil radon concentration in a geological fault zone is presented, with data collected over the course of one year. The ensemble empirical mode decomposition (EEMD) method is used to decompose the signal into its characteristic modes. On the physically significant modes obtained through EEMD, the Hilbert–Huang transform (HHT) is applied to represent the time–energy–frequency profile of the soil radon time series. After removing the periodic components from the time series, the analyses revealed a strong correlation between the recorded radon anomalies and local seismic events.

In another large-scale study covering data from the past 50 years in Greece, soil radon time series are also analyzed. The authors used the method of singular spectrum analysis to identify change points in the radon time series. They eliminated the influence of meteorological parameters and discovered correlations between radon variations and earthquakes with magnitudes ranging from M3.5 to 6.5. The anomalies lasted from days to weeks, depending on the type of fault [35].

A study on the various factors influencing soil radon concentrations is presented in [36]. The authors classify radon anomalies in groundwater, explore their causes, examine the environmental influences, and analyze the effects of soil and rock structures on radon

migration. They emphasize the potential of machine learning for classifying and predicting radon anomalies caused by earthquakes, experimenting with boosted trees, support vector machines, neural networks, multivariate linear regression, decision tree algorithms, and others.

A study of radon data from a hot spring spanning a 38-year period is presented in [37]. Using wavelet coherence analysis, the authors identified the factors influencing radon fluctuations, including spring discharge, water temperature, rainfall, and barometric pressure. They developed decision tree models to simulate the "background" radon levels. Anomalies are identified by comparing observed radon changes with the modeled "background" fluctuations. The modeled "background" variations in the radon time series are then compared with observations during seismic activity periods. Using this method, they identified 15 potential radon anomalies among 24 selected earthquakes. The authors found that the detected anomalies were also accompanied by changes in water temperature and spring discharge. They attribute the anomalous increase in radon to its continuous supply from newly formed internal surfaces of cracks to the aquifer system. The anomalous decrease in radon is explained by radon partitioning into the gas phase and changes in the mixing ratio of shallow and deep water.

In [38], a study is presented where the empirical mode decomposition (EMD) method is used to detect oscillations in radon time series. The radon time series are decomposed into various oscillatory modes, known as intrinsic mode functions (IMFs). After applying the Hilbert–Huang transform (HHT) to the significant IMFs, several interesting non-linear characteristics are identified. The temporal variation of the instantaneous energy showed a strong correlation with four local earthquakes during the study period. Most interestingly, interruptions in the temporal evolution of the instantaneous energy function are observed prior to these local earthquakes. Many more studies on radon precursors for earthquake forecasting have been systematically reviewed in [39,40].

Another commonly used approach for modeling uncertainty and dynamically updating predictions based on new data is Bayesian methods. For example, Bayesian dynamic modeling is used for the probabilistic prediction of pavement conditions, integrating multiple uncertain factors—weather conditions, mechanical load, pavement age, and other parameters [41–44]. This technique enables adaptive real-time model adjustments, improving the accuracy of long-term forecasts. In addition to various scientific and engineering disciplines, Bayesian methods are also applied in geophysical and seismological analyses [45–48].

While many studies consider radon as an indicator of seismic activity, we use ANNs to predict normal radon levels under non-seismic conditions and to detect anomalies around the time of earthquakes. This approach allows for better isolation of the effect of earthquakes on radon concentration. We apply segmentation using annuli in the feature space, which provides more precise results by enabling localized data analysis—a novelty compared to other more global models. Integrating wavelet transformation for noise removal further enhances the reliability of predictions. This study also examines regional differences in radon anomalies, analyzing data from three monitoring stations in Bulgaria—Yambol, Dimitrovgrad, and Krupnik. The obtained results indicate that local geological and environmental conditions may play a key role in radon emissions—a factor often overlooked in previous studies. The findings contribute to the ongoing discussion on the role of radon as a potential earthquake precursor, while also demonstrating the advantages of ANN models in identifying anomalies in geophysical time series.

3. Statistical Study of the Data Obtained from Radon Stations

Before conducting the main study related to the approximation of radon concentration from multiple ANNs, we examine and analyze the data as statistical series from large samples. The analysis includes:

- examining the data as grouped statistical series;
- descriptive statistics analyzing measures of central tendency, dispersion, and shape;
- constructing and analyzing the quartile diagram and interquartile range;
- confidence intervals for the mean values.

The analysis is conducted on the data measured by the radon stations for radon concentration, temperature, and atmospheric pressure. This part of the article presents the studies for the Yambol station, based on data from 55,954 measurements for the period from 26 July 2022 to 25 December 2023. Similar studies have been conducted for the Dimitrovgrad and Krupnik stations.

3.1. Grouped Statistical Series

In the analyzed data for the Yambol station, the minimum and maximum values of radon concentration are 79.55 Bq/m^3 and 2404.63 Bq/m^3, respectively, determining the range of the sample as $Range_Rn$ = 2325.08 Bq/m^3. Atmospheric pressure varies from 956.6 hPa to 1016.8 hPa, meaning $Range_Ap = 60.2$ hPa. The range of the temperature statistical series is $Range_{Temp} = 19.2$ °C, with minimum and maximum values of 8.9 °C and 28.10 °C, respectively.

When considering the measured radon concentration as a grouped statistical series, 16 groups with a width of 145.3175 were determined (Table 1).

Table 1. Grouped statistical series of radon measurements at the Yambol station.

Lower Bound (Li)	Upper Bound (Ui)	Midpoint (Mi)	Frequency (fi)	Relative Frequency (pi)	Cumulative Frequency	Cumulative Relative Frequency
79.55	224.87	152.21	1276	0.0228	1276	0.0228
224.87	370.19	297.53	5505	0.0984	6781.00	0.1212
370.19	515.50	442.84	9197	0.1644	15,978.00	0.2856
515.50	660.82	588.16	10,885	0.1945	26,863.00	0.4801
660.82	806.14	733.48	8900	0.1591	35,763.00	0.6392
806.14	951.46	878.80	6694	0.1196	42,457.00	0.7588
951.46	1096.77	1024.11	4937	0.0882	47,394.00	0.8470
1096.77	1242.09	1169.43	3416	0.0611	50,810.00	0.9081
1242.09	1387.41	1314.75	2224	0.0397	53,034.00	0.9478
1387.41	1532.73	1460.07	1575	0.0281	54,609.00	0.9760
1532.73	1678.04	1605.38	734	0.0131	55,343.00	0.9891
1678.04	1823.36	1750.70	371	0.0066	55,714.00	0.9957
1823.36	1968.68	1896.02	146	0.0026	55,860.00	0.9983
1968.68	2114.00	2041.34	75	0.0013	55,935.00	0.9997
2114.00	2259.31	2186.65	13	0.0002	55,948.00	0.9999
2259.31	2404.63	2331.97	6	0.0001	55,954.00	1.0000

The most common radon concentration at the Yambol station is in the range of (370.19–806.14) Bq/m^3. This interval is relatively narrow: *range* = 332.16 Bq/m^3, considering the total sample range of 2325.08 Bq/m^3. Figure 1 presents the histogram of the grouped radon series.

Similarly, the grouped series for the factors of atmospheric pressure and temperature have been studied. For instance, regarding atmospheric pressure at the Yambol station, it can be concluded that the most frequently measured values of atmospheric pressure fall

within the range of (986.7–999.60) hPa. For the grouped series of measured temperature, the most frequently measured temperatures at the Yambol station are in the ranges of (14.39–15.76) and (26.73–28.10) °C.

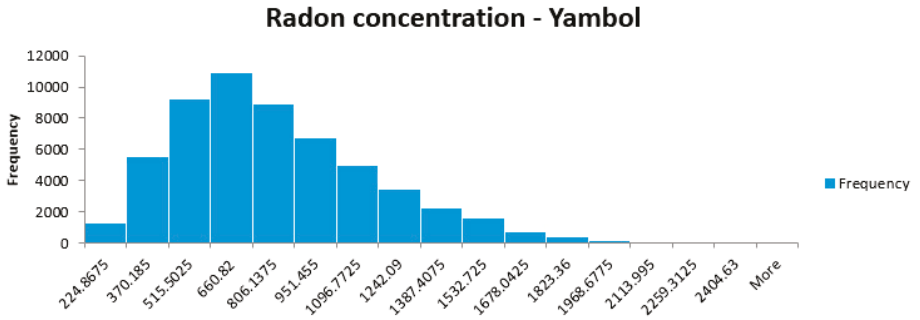


Figure 1. Histogram of the grouped statistical series of the measured radon concentration at the Yambol station.

3.2. Descriptive Statistics

The analysis included in the descriptive statistics of the data consists of measuring the central tendency, dispersion, and shape (Table 2).

Table 2. Statistics for radon concentration, atmospheric pressure, and temperature at the Yambol station.

			Radon Concentration	Atmospheric Pressure	Temperature
Central Tendency	Mean	$\overline{C}$	737.4624	993.8365	19.22757
	Mode	Mo	564.62	992.2	12.7
	Median	Md	676.73	993.7	18.2
Dispersion	Variance	σ^2	115,842.3	53.99225	32.59624
	Standard deviation	σ	340.3561	7.347942	5.709312
	Range	R	2325.08	60.2	19.2
Shape	Skewness	Sk	0.798991	0.087905	0.107303
	Kurtosis	Ku	0.460747	0.61482	−1.45447

The study of the statistical series for radon concentration shows that the most frequently observed radon concentration at the Yambol station is 564.62 Bq/m^3. Fifty percent of the measurements showed concentrations lower than 676.73 Bq/m^3, while the other 50% indicated concentrations above this value. The horizontal skewness of the distribution curve is 0.798. The numerical measure of the accumulation of concentrations at the ends of the observed interval is 0.46.

The study of atmospheric pressure shows that the most frequently observed value of atmospheric pressure at the Yambol station is 992.2 hPa. Fifty percent of the measurements showed values lower than 993.7 hPa, while the other 50% were above this value. The horizontal skewness of the distribution curve is 0.088. The numerical measure of the accumulation of atmospheric pressure values at the ends of the observed interval is 0.614.

From the analysis of the series of measured temperatures, it can be stated that the most frequently observed temperature at the Yambol station is 12.7 °C. Fifty percent of the measurements showed temperatures lower than 18.2 °C, while the other 50% indicated temperatures above this value. The horizontal skewness of the distribution curve is 0.107. The numerical measure of the accumulation of temperatures at the ends of the observed interval is −1.4544, indicating that the distribution has reduced kurtosis.

3.3. Quartile Diagrams

The analysis of the quartile diagram for the sample of radon concentration measurements shows that 25% of all measured radon concentration values are below 489.51 Bq/m^3, and 25% of the measurements indicate concentrations above 938.32 Bq/m^3 (Table 3). It is important to note the existence of abnormal radon concentration values greater than the upper boundary of 1611.535 Bq/m^3.

Table 3. Quartile data and interquartile range of the radon sample.

Quartile	Radon Concentration (Bq/m^3)
Median $= Q2$	676.73
First quartile, $Q1$	489.51
Third quartile, $Q3$	938.32
Minimum, *min*	79.55
Maximum, *max*	2404.63
Interquartile range	448.81
Lower bound	−183.705
Upper bound	1611.535

Abnormal values were also detected in the statistical series of atmospheric pressure. When determining the interval

$$[Q_1 - 1.5Q_{3,1};\ Q_3 + 1.5Q_{3,1}],$$

where

$$Q_{3,1} = Q_3 - Q_1$$

is the interquartile range formed by the quartiles Q_3 and Q_1, it turns out that there are abnormal atmospheric pressure values lower than the lower bound of 975.1 hPa and higher than the upper bound of 1011.9 hPa (Figure 2).

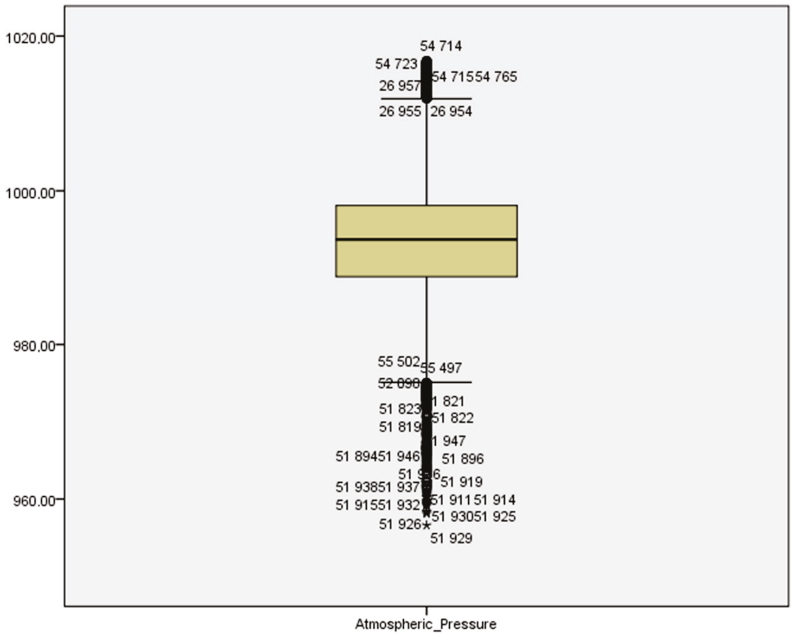


Figure 2. Quartile diagram of the statistical series of measured atmospheric pressure data at the Yambol station.

In Figure 2, we observe multiple values of atmospheric pressure below the lower bound and above the upper bound of the interquartile range of the sample. It is evident

that atmospheric pressure in Yambol exhibits high variability. In fact, atmospheric pressure in the Yambol region, as well as across Bulgaria, is subject to seasonal and short-term meteorological influences, which can lead to significant fluctuations. During winter, high-pressure areas known as thermal anticyclones form over Bulgaria, resulting in increased atmospheric pressure during this season. In contrast, during summer, atmospheric pressure is generally lower due to the warming of the Earth's surface and the associated processes. Additionally, the dynamics of atmospheric pressure are strongly influenced by the movement of cyclones (low-pressure areas) and anticyclones (high-pressure areas). They are often responsible for rapid and significant changes in pressure. The passage of cold and warm fronts also leads to abrupt shifts in atmospheric pressure over short periods. All these factors, when considered together, explain the high variability of atmospheric pressure in the region. As a result, numerous values of this factor fall outside the boundaries of the interquartile range of the statistical dataset.

The study of the quartile diagram for temperature shows that there are no abnormal values outside the interval determined by the extreme values and the interquartile range.

3.4. Confidence Intervals for the Mean Values

The results of the confidence interval calculations for the measurements at the Yambol station are presented in Table 4. The significance level used is $\alpha = 0.05$.

Table 4. Confidence intervals for the mean values of radon concentration, atmospheric pressure, and temperature measured at the Yambol station.

Quartile	Radon Concentration	Atmospheric Pressure	Temperature
Sample Size	55,954	55,954	55,954
Mathematical Expectation/ Point Estimate from the Sample	737.46	993.84	19.23
Standard Deviation	340.36	7.35	5.71
Variance	115,842.31	53.99	32.60
Significance Level	0.05	0.05	0.05
Margin of Error of the Confidence Interval (d)	2.82	0.06	0.05
Lower Bound of the Confidence Interval	734.64	993.78	19.18
Upper Bound of the Confidence Interval	740.28	993.90	19.27

As a conclusion from the calculations, with 95% confidence it can be stated that:

- The population mean of radon concentration lies within the interval $(734.64,\ 740.28)$;
- The population mean of atmospheric pressure lies within the interval $(993.78,\ 993.90)$;
- The population mean of temperature lies within the interval $(19.18,\ 19.27)$.

4. Use of ANNs in Noisy Data with Large Sample Sizes

Based on the research of Dobrovolsky [49], in his work Sergey Biryulin [50] shows that the distance R, at which an earthquake with a given magnitude M has an impact, causing deformations in the lithospheric plates, can be determined. The following expression is provided:

$$R[km] = \begin{cases} 10^{0.5M}, & M < 5 \\ 10^{0.433M} & M \geq 5 \end{cases}. \quad (1)$$

Measurements of radon concentration and related parameters were conducted using stationary measuring stations located in different parts of Bulgaria. During the study, a large dataset of over half a million measurements was collected from dozens of locations across the country. The total number of measurements is a result of both the observation period and

the operating parameters of the measuring instruments. The radon monitoring devices are equipped with a specialized RD200M ionization sensor, two Geiger–Müller counters (SBM-20, shielded with a lead casing, and SBT-7), as well as sensors for temperature, relative humidity, and atmospheric pressure. The measuring instruments are calibrated for automatic readings every 15 min, enabling a detailed time-series analysis of radon concentrations and their dynamics. This time interval is a standard practice in long-term monitoring, as it provides a balance between data frequency and the workload on the measuring equipment.

In our study, for training the ANNs and assessing their ability to approximate large datasets containing noise, we use a sample of 57,740 measurements (1 every 15 min) from the radon concentration monitoring station in Yambol ($Latitude_GPS = 42.481816$, $Longitude_GPS = 26.497799$), Bulgaria, over the period from 26 July 2022 to 25 December 2023. The measurements from 10 March 2023 were excluded from the sample, as an earthquake with a magnitude of 3.6 was detected at a distance of 75.6 km from the station, which, according to (1), would be expected to influence the lithospheric plates and therefore potentially impact radon concentration in the area. For training the ANNs, data from non-earthquake days and days with earthquakes that, according to (1), occurred at a distance unlikely to have any effect were used. Thus, 57,597 samples were utilized to build the radon concentration model in the absence of radon-influencing earthquakes.

The variables used as input data for the ANNs are:

- atmospheric pressure at the measurement location Ap (hPa);
- relative humidity Rh (%);
- temperature T (°C).

The target variable for the ANNs is the radon concentration C_{Rn}(Bq/m^3).

The inclusion of atmospheric pressure, temperature, and humidity as input features in the ANN model is justified through a correlation analysis between these parameters and radon concentration C_{Rn}. For this purpose, a Pearson correlation matrix was calculated, which shows the degree of linear relationship between the parameters (Figure 3).

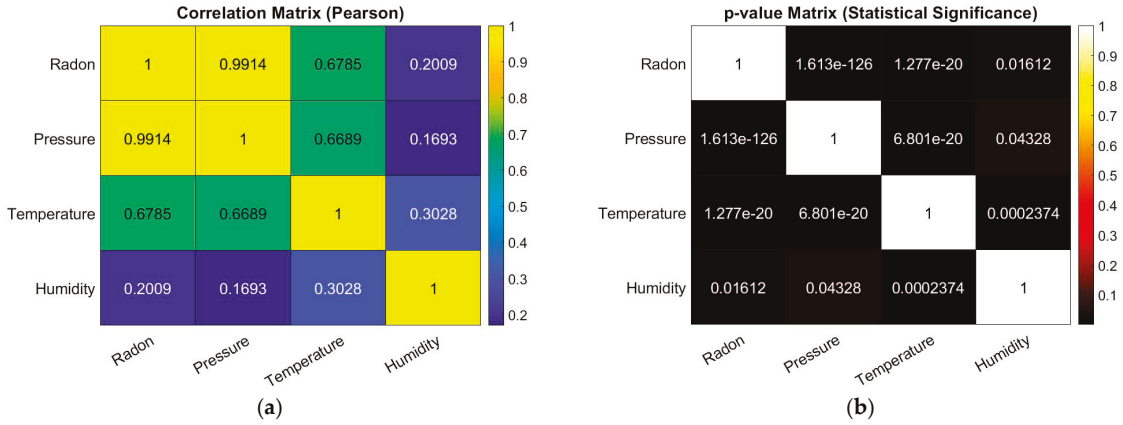


Figure 3. (**a**) Correlation matrix of radon concentration and the factors used; (**b**) Table of correlation significance. The significance level used is $\alpha = 0.05$.

In this analysis, atmospheric pressure (P) exhibits a very strong positive correlation with radon concentration (r = 0.991). This suggests that pressure is a key factor influencing radon behavior. The strong relationship between pressure and radon confirms existing hypotheses regarding the impact of meteorological conditions on radon emissions and their interpretation in the context of seismic activity. Temperature (T) also shows a moderate positive correlation with radon (r = 0.678), indicating a significant but weaker influence compared to pressure. Based on these results, atmospheric pressure and temperature

were included as input parameters in the ANN model, as they have a notable relationship with radon concentrations. Humidity, on the other hand, demonstrates a weak linear dependence (r = 0.2), which raises the question of its necessity in the model. The Spearman coefficient is even lower (0.18).

Nevertheless, we prefer to retain air humidity as an input variable for the neural network, and our reasoning is as follows. The Pearson and Spearman correlation coefficients are primarily designed to capture linear or monotonic dependencies, which may not fully reflect the complex environmental interactions. On the other hand, ANNs can detect non-linear and multifactorial relationships that are not apparent through traditional correlation analysis. For example, humidity may influence radon diffusion only under specific values of atmospheric pressure and temperature, or its effect may exhibit a delayed response. Additionally, from a physical standpoint, humidity can reduce soil permeability, potentially affecting radon emissions in ways that cannot be directly measured through correlation analysis. To account for these possibilities, we included humidity as an input parameter in the ANN model. This allows the neural networks to assess whether humidity contributes to variations in radon concentration through non-linear and multivariate interactions, even if its direct correlation with radon appears weak.

In its initial form, the sample of input–output pairs contains a certain amount of noise due to various factors, such as electromagnetic interference, thermal fluctuations in the materials of the devices, mechanical vibrations, insufficient calibration, or anomalies in the data caused by different reasons. When attempting to approximate the target function

$$C_{Rn} = C_{Rn}(Ap,\ Rh,\ T) \tag{2}$$

using ANNs, the best result is achieved with a six-layer feedforward ANN, with 53 neurons in each layer. In the neurons of the hidden layers, the hyperbolic tangent function is used:

$$Tanh\ (x) = \frac{2}{1 + e^{-2x}} - 1 \tag{3}$$

and the training is conducted using the Levenberg–Marquardt algorithm. All 57,597 measurements are divided into three groups:

- Training samples—75% (40,317);
- Validation samples—15% (8640);
- Independent test samples, which determine the predictive (interpolating and extrapolating) capabilities of the created ANN—15% (8640).

Under these conditions, and without noise removal from the data, the results rounded to the nearest whole number obtained with this ANN are useless (Table 5).

Table 5. Errors in the approximation of radon concentration dependency with a sample size of 57,740 containing noise in the data.

MSE		
Training	**Validation**	**Test**
16,731	16,232	17,111

The mean square error (MSE) is used to evaluate the performance:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(C_{Rni} - \widehat{C_{Rni}})^2, \tag{4}$$

where n is the number of samples, C_{Rni} is the actual value of the radon concentration, and $\hat{C}_{Rni}$ is the value of radon predicted by the ANN for the corresponding example. The correlation coefficient R was determined to be 0.89. It is evident that these large error values,

in all three groups, are unacceptable, especially considering that the radon concentration is measured in units several orders of magnitude smaller. To conduct a comparative analysis under noisy conditions and after noise removal, control groups of ANNs with reduced architectures were constructed.

Table 6 shows the training results for some of the ANNs from the control group. All ANNs in the group are single-layered using the transfer function (3) and trained with the Levenberg–Marquardt algorithm. The errors in the control group range from 27,567.56 for the ANN with a single neuron to 18,752.43 for the most optimal single-layer ANN with 4831 neurons. The correlation coefficient after training the ANNs in the control group varies from 0.67 to 0.91.

Table 6. Partial results for single-layer ANNs from the experimental group, trained on original data (without noise removed).

Number of Neurons	*MSE*	*R*
8	23,400.35	0.76
9	23,200.23	0.76
10	23,161.81	0.77
...	...	...
50	22,215.81	0.83
...	...	...
153	20,001.85	0.85
...	...	...

To examine the impact of noise in the sample, the noise was removed using the "Wavelet Signal Denoiser" tool in the MATLAB v2018a environment. The noise removal process with this tool involves transforming the data and includes the following steps:

1. The signal is represented in the time–frequency domain through wavelet transformation.
2. Noise is identified and analyzed in the time–frequency domain.
3. Based on the noise analysis, parts of the signal deemed to contain noise are filtered or removed.
4. The wavelet transformation is inverted to return to the original time-domain signal, but with the noise removed.

To achieve optimal noise reduction, several options were used for tuning the filter and wavelet parameters:

1. A symmetric wavelet of order 4 (sym4) was used for the signal decomposition function.
2. The signal was decomposed 13 times during wavelet decomposition, ultimately resulting in 13 levels.
3. The "Bayes" method was applied to distinguish between the noise and the actual signal.
4. The median of the difference between the wavelet coefficients and the estimated noise value was used to determine the threshold.
5. The noise parameter was set to "Noise: Level Independent," meaning that the noise was considered independent of the decomposition level.

Figure 4 shows the data for the measured radon concentrations after noise removal.

The choice of sym4 and 13 levels of decomposition was made to maximize noise suppression efficiency while preserving the structural information of radon concentration anomalies. This approach allowed for better representation of actual variations in the data and minimized artifacts that could potentially affect the analysis performed by the neural networks. A symmetric wavelet function (sym4) was selected because it provides an optimal balance between time and frequency localization—wavelet functions from the "symlet" family are modified versions of Daubechies wavelets (Db) but with improved symmetry. In our case,

symmetry is crucial, as it reduces phase distortions, which could otherwise impact signal reconstruction accuracy. Additionally, sym4 allows for minimal boundary distortion. Unlike other wavelet functions such as Daubechies (Db) or Coiflet (coif), sym4 ensures a smoother signal reconstruction and reduces edge artifacts in the time series.

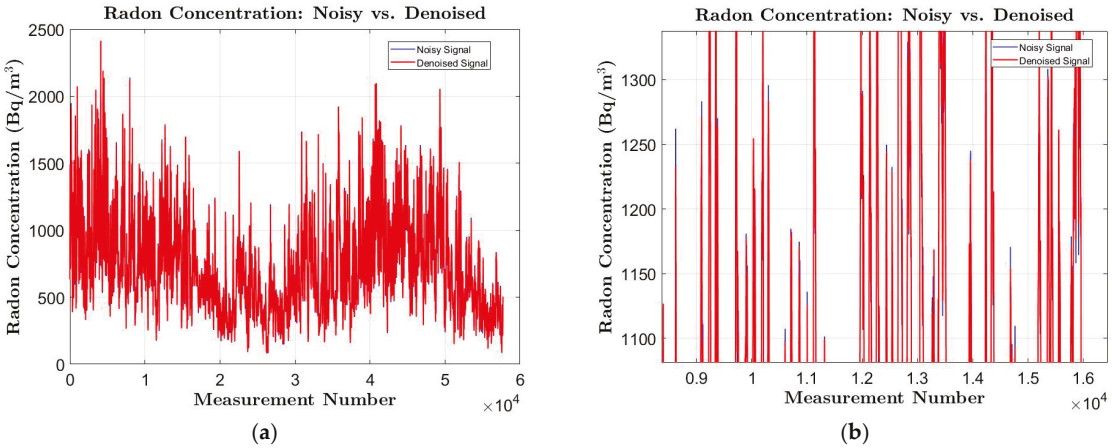


Figure 4. (**a**) Data for the measured radon concentrations at the Yambol station after noise removal. (**b**) Magnified view of noise in the data.

Since wavelet transformation is applied to a dataset with a large number of measurements (57,740), we have the capability to use deeper decomposition levels. This is why we chose 13 levels of decomposition. A preliminary analysis of radon time series revealed that the primary fluctuations related to radon concentration are retained in the lower-frequency components (detail coefficients at levels 10–13), while high-frequency noise is predominantly present in the first few levels. By using 13 levels, we ensure that most low-frequency radon variations remain in the signal, while high-frequency disturbances are effectively removed. Tests conducted with different decomposition levels showed that, when using fewer than 10 levels, a significant amount of noise remains, whereas using more than 13 levels does not lead to substantial improvement but unnecessarily increases computational complexity. As a result, we ultimately selected 13 levels for the signal decomposition.

The noise removal procedure was applied to all input factors: atmospheric pressure at the measurement location, relative humidity, and temperature. After this process, the input signals exhibit much clearer characteristics, which significantly facilitate the subsequent training of the ANNs.

To compare the training of ANNs in conditions with noisy data and those where the noise has been removed, an experimental group of ANNs was constructed with the same architecture and characteristics as the ANNs in the control group. Once again, the hyperbolic tangent (3) was used in the neurons of the hidden layers, the training was conducted using the Levenberg–Marquardt algorithm, and all 57,597 measurements were divided into three groups: training samples—75% (40,317); validation samples—15% (8640); independent test samples—15% (8640).

Table 7 presents some of the results from the experimental group of ANNs, which correspond to the control group ANNs shown in Table 6. It is evident that the removal of noise from the training samples leads to a significant reduction in the MSE. In the experimental ANN with 153 neurons, the MSE is now 7716, which is several times smaller than the MSE for this architecture when noise is present in the data.

In experiments with multilayer ANNs, the best result was achieved with an eight-layer ANN, each hidden layer containing 15 neurons, yielding an MSE of 2456.79.

Table 7. Partial results for single-layer ANNs from the experimental group, trained on data with noise removed.

Number of Neurons	*MSE*	*R*
8	19,216.11	0.89
9	19,202.39	0.89
10	18,633.62	0.92
...	...	...
50	12,241.97	0.94
...	...	...
153	7716.20	0.98
...	...	...

Despite the clear advantage of using input–output samples with noise removed, models remain quite complex in terms of architecture when working with large sample sizes (in this case, 57,597). They are slow in both implementation and construction, and the resulting errors are still not small enough to be acceptable.

5. Model for Predicting Radon Levels Through the Construction of Multiple ANNs

Very often, with large sample sizes, ANNs develop complex architectures where the number of layers and neurons is large. The training error can remain high despite the large number of neurons, due to the strong indeterminacy of the target function in relation to the predictors used. Theoretically, an ANN model that covers all the data from a large sample is convenient because, once created, it can be used for interpolation across the entire sample, as well as for extrapolation beyond its limits. However, from a practical standpoint, this can sometimes be unnecessary.

Imagine having a sample of 100,000 instances for approximating a target function, such as $y = x^5$, distributed across the segment $[-50,000,\ \ 50,000]$ (Figure 5). If the practical need is to predict the value of the target function at a point within the sample, for instance at point $x = 4.7$, there is no need to train the model to return results at far-off points like $x = 47,899$. If we have a sufficiently rich set of samples in the vicinity of $x = 4.7$, we could use them to create an accurate model that describes a much smaller segment, perfectly suited to our needs.

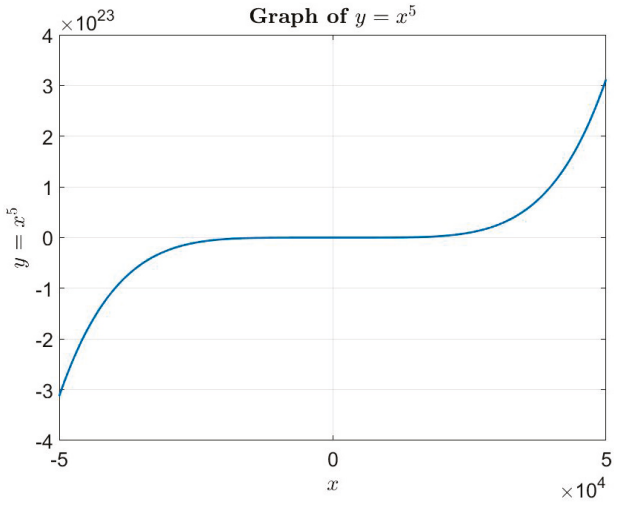


Figure 5. Graph of the function $y = x^5$.

5.1. Algorithms for Building and Using Multiple ANNs for Predicting Soil Radon

The method we propose consists of creating a set of ANNs, each operating within different ranges of samples characterized by large volume and a high level of indeterminacy. Figure 6 presents a schematic of the algorithm for building multiple ANNs, each trained on separate multidimensional annuli of the input–output data.

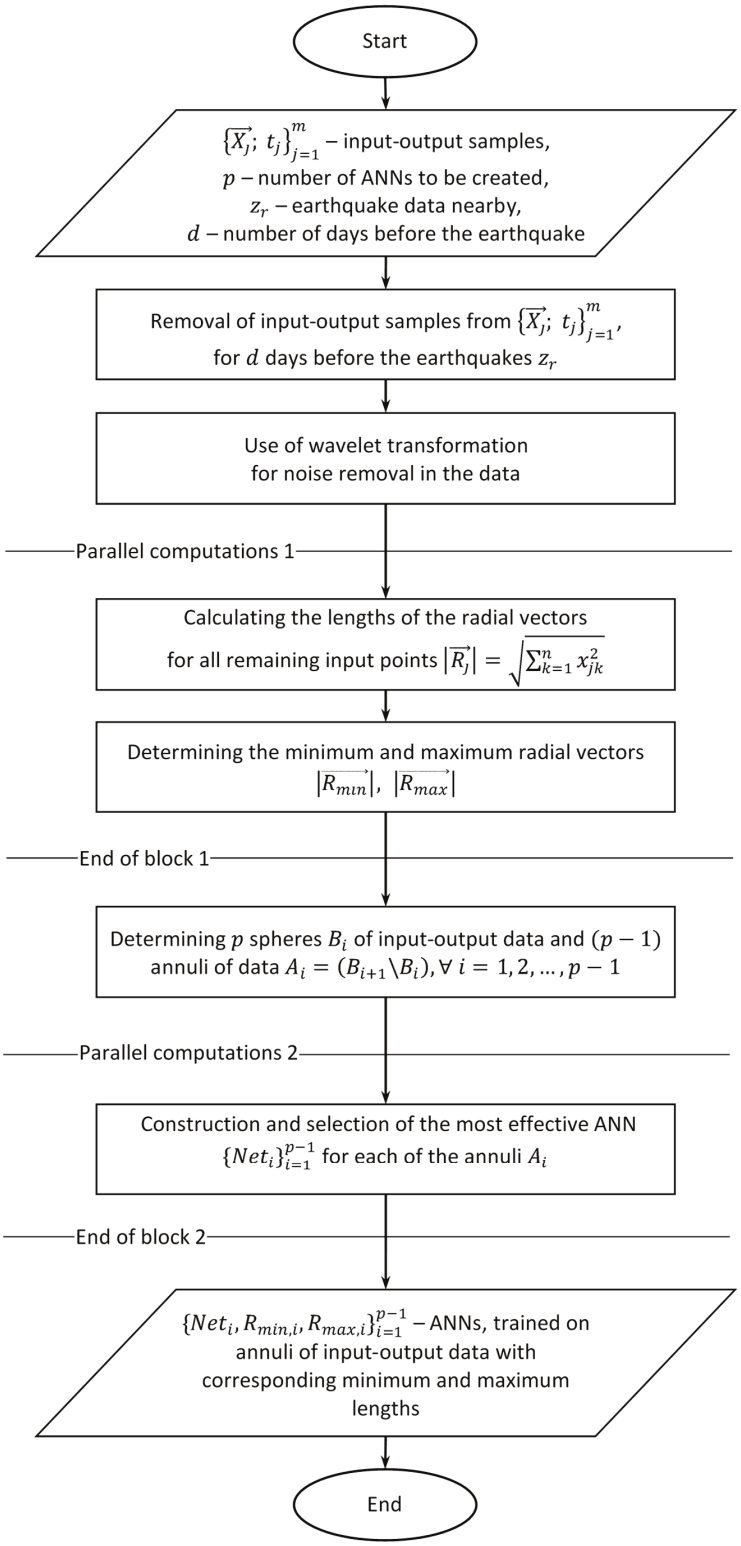


Figure 6. Algorithm for building multiple ANNs for predicting soil radon levels for a given station.

Key steps in the algorithm for building multiple ANNs to predict soil radon levels for a given station are:

1. Set the required parameters:
    - Input–output data, including date and time of measurements, humidity, temperature, atmospheric pressure, and radon (selected as the target variable);
    - number of predictive ANNs;
    - earthquake data during the period of the input–output data;
    - number of days before the earthquakes for which input–output data will be removed.
2. Remove input–output samples for the specified number of days before the earthquakes. This ensures that only input–output data potentially unaffected by processes preceding an upcoming earthquake are used. For more universal use, the number of days is set as a parameter in the algorithm.
3. Remove noise from the data, using appropriate wavelet transformations for this purpose.
4. Calculate the radial vectors of all input samples.
5. Divide the input–output data into annuli, based on the lengths of the radial vectors.
6. Build multiple ANNs on each of the annuli and select the most effective one, which will be used for prediction.

As a result of the algorithm, the most effective ANN is selected for each of the annuli, which is then used to predict radon values for new input samples. Some activities within the model—such as independent calculations of vector lengths and ANN training—can be performed either iteratively or in parallel, depending on the available computational resources. Parallel computations would significantly accelerate the training process of the ANNs.

The prediction process is shown in Figure 7. For each new input sample, the length of its radial vector is first calculated. Based on this length, the appropriate ANN from the constructed set of ANNs is selected. The radon level is then predicted using the chosen ANN.

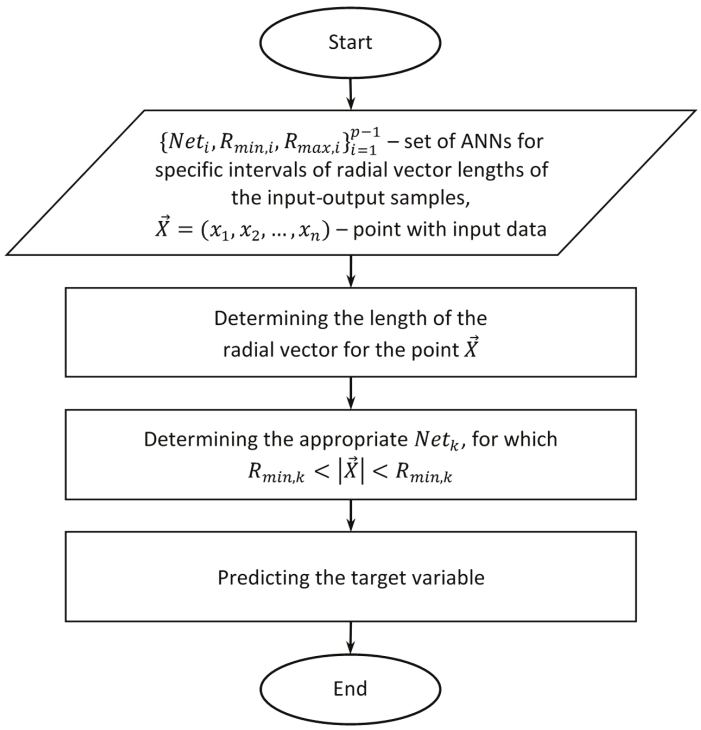


Figure 7. Algorithm for prediction on a single input sample.

By applying this procedure to all measured points that were not used in building the ANNs, it will be possible to determine whether there are clear dependencies that suggest the potential for real-time earthquake prediction.

5.2. Mathematical Aspects of the Model

The key aspects of the mathematical model of the algorithm are related to:

- Dividing the sample into subsamples;
- Constructing a set of ANNs;
- Predicting the value of the target function.

5.2.1. Dividing the Sample

Let the sample of input–output pairs be given as:

$$\left\{ \left[x_{j1}, x_{j2}, \ldots, x_{jn};\ t_j = F\left(x_{j1}, x_{j2}, \ldots, x_{jn}\right) \right] \right\}_{j=1}^{m}. \tag{5}$$

Each sample consists of an ordered n-tuple of input data $\overrightarrow{X_j} = (x_{j1}, x_{j2}, \ldots, x_{jn})$ and its corresponding functional value of the target function $t_j = F\left(\overrightarrow{X_j}\right)$. If we use a coordinate system with the origin at point $O\ (0,\ 0, \ldots,\ 0)$, then each point $(x_{j1}, x_{j2}, \ldots, x_{jn})$ in the input data, located in the n-dimensional predictor space, corresponds to a radial vector $\overrightarrow{R_j}$ with the following length:

$$|\overrightarrow{R_j}| = \sqrt{\sum_{k=1}^{n} x_{jk}^2},\ \forall\ j = 1, 2, \ldots, m. \tag{6}$$

Let $\overrightarrow{R_{max}}$ and $\overrightarrow{R_{min}}$ denote the radial vectors with the largest and smallest lengths, respectively, i.e.,

$$|\overrightarrow{R_{max}}| = \max(|\overrightarrow{R_j}|)_{j=1}^{m}, \tag{7}$$

$$|\overrightarrow{R_{min}}| = min(|\overrightarrow{R_j}|)_{j=1}^{m}, \forall j = 1,\ 2, \ldots,\ m. \tag{8}$$

The sphere $B_p(O,\ |\overrightarrow{R_{max}}|)$ is divided into $(p-1)$ n-dimensional annuli through nested spheres with the center at the origin of the coordinate system:

$$B_1 \subset B_2 \subset B_3 \subset \ldots \subset B_p, \tag{9}$$

where $B_1(O, |\overrightarrow{R_1}|)$, and for any two spheres $B_i(O, |\overrightarrow{R_i}|)$ and $B_{i+1}(O, |\overrightarrow{R_{i+1}}|)$, satisfies the condition $|\overrightarrow{R_i}| < |\overrightarrow{R_{i+1}}|, \forall i = 1, 2, \ldots, p-1$. Each of the annuli $A_i = (B_{i+1} \backslash B_i)$ contains the points from the predictor sample with radial vectors whose lengths lie in the interval $[|\overrightarrow{R_i}|, |\overrightarrow{R_{i+1}}|]$.

5.2.2. Constructing a Set of ANNs

At this stage of the algorithm, a set of $p-1$ different ANNs $\{Net_i\}_{i=1}^{p-1}$ is constructed, where each ANN Net_i is trained on data from the annulus A_i, i.e., on samples from the dataset (5), for which:

$$|\overrightarrow{R_i}| \ \le\ \sqrt{\sum_{k=1}^{n} x_{rk}^2} \ \le\ |\overrightarrow{R_{i+1}}|,\ \forall\ i = 1,\ 2, \ldots, p-1\ \ 1 \ \le\ r \ \le\ m. \tag{10}$$

The overall MSE of the set of ANNs is

$$MSE = \frac{\sum_{i_1=1}^{q_1} (C_{Rni_1} - \widehat{C_{Rni_1}})^2 + \sum_{i_2=1}^{q_2} (C_{Rni_2} - \widehat{C_{Rni_2}})^2 + \ldots \sum_{i_{p-1}=1}^{q_{p-1}} (C_{Rni_{p-1}} - \widehat{C_{Rni_{p-1}}})^2}{q_1 + q_2 + \ldots + q_{p-1}}, \quad (11)$$

where q_i is the number of samples in each annulus A_i, and C_{Rni_j} and $\hat{C_{Rni_j}}$ are the actual and predicted radon concentrations by the ANNs for each sample in the annulus, respectively.

Given that the sum of the samples in all annuli is actually equal to the total sample size, i.e.,

$$\sum_{i=1}^{p-1} q_i = m \quad (12)$$

it follows that

$$MSE = \frac{1}{m} \sum_{j=1}^{p-1} \sum_{i=1}^{q_j} \left(C_{Rni_j} - \widehat{C_{Rni_j}} \right)^2. \quad (13)$$

5.2.3. Predicting the Value of the Target Function

When it is necessary to predict the value of the target function at a new point $(x_1, x_2, \ldots x_n)$ the algorithm calculates the length of the radial vector of the point in the predictor space:

$$|\overrightarrow{R_i}| = \sqrt{\sum_{k=1}^{n} x_k^2}. \quad (14)$$

Then, the algorithm selects the ANN $Net_q \in \{Net_i\}_{i=1}^{p-1}$, which was trained in the region of the n-dimensional annulus A_q that contains the point $(x_1, x_2, \ldots, x_n)$, and performs the prediction using that ANN.

6. Experiments

The main expectations during the experiments are:

1. Predictions of radon quantities for the input–output samples that were removed from the training data due to approaching earthquakes will differ from the actual measured values of the target variable.
2. Depending on the stations, the influence of the predictors on radon levels will vary in strength.
    - It is possible that at some stations there will be a clear difference between the measured and predicted radon values, while at others, the difference may be minimal.
    - It is possible that at some stations the predicted radon levels will be higher than the measured ones, at others they may be lower, and at some, there may be no significant difference.

In the conducted experiments, we divide the space containing all the points from the real sample into several annuli, each with an equal number of input–output data points. The data in the sample are for:

- atmospheric pressure at the measurement location Ap (hPa);
- relative humidity Rh (%);
- temperature T (°C).

When dividing into 10 annuli, each group of input–output samples consists of 5760 elements, and when divided into 20 annuli, each group consists of 2880 elements. The input–

output samples from each annulus are further divided into groups used for the creation of the corresponding ANN: training—75%; validation—15%; independent test—15%.

Table 8 describes the results from training single-layer ANNs in two specific regions $(B_2 \backslash B_1)$, when dividing the space into 10 and 20 annuli. The results for other regions are similar, indicating that ANNs become more efficient with a larger number of annuli (and consequently a smaller number of input–output samples per annulus) and with an increase in the number of neurons. Of course, both the number of neurons and the number of input–output samples have critical limits, which have been explored in various literature sources [51,52].

Table 8. Partial results from training single-layer ANNs on one of the annuli, when dividing the predictor space into 10 and 20 annuli.

Number of Neurons	Results for 1 of the 10 Annuli		Results for 1 of the 20 Annuli	
	MSE	***R***	***MSE***	***R***
8	6052.68	0.87	3661.32	0.90
9	5451.17	0.89	3341.62	0.93
10	5148.43	0.91	3048.93	0.94
...	...	...	...	...
50	1982.38	0.97	882.66	0.98
...	...	...	...	...
153	892.61	0.98	641.25	0.99
...	...	...	...	...

The experiments conducted with multilayer ANNs also showed a significant reduction in error. For example, with 10 annuli, the MSE for a six-layer ANN with 53 neurons in each layer is only 2.68 units. Similarly, with 20 annuli, the MSE for the same six-layer ANN with 53 neurons per layer decreases to 1.7 units.

To evaluate the effectiveness of this approach, a comparative analysis was conducted using four popular regression models:

i. Linear Regression (LR). One of the most basic methods, which assumes a linear relationship between the input parameters (atmospheric pressure, temperature, and humidity) and radon concentration. Given the strong linear correlation between radon concentration and atmospheric pressure, as well as the moderate correlation with temperature, we were particularly interested in examining the results of LR and ARIMA.
ii. ARIMA. An auto regressive integrated moving average model, widely used for time series analysis and forecasting. The selected model, ARIMA(1,1,1), includes one autoregressive term, one differencing step, and one moving average term.
iii. Random Forest (RF). An ensemble model based on multiple regression trees (a 100-tree ensemble was used in this case), where the mean prediction is taken to improve the model's stability.
iv. Support Vector Regression (SVR). A regression model that applies non-linear transformations using radial basis function (RBF) kernels to determine the optimal hyperplane for predicting radon values.
v. Artificial Neural Network. The ANN model consists of six layers with 53 neurons per layer and is trained using the backpropagation algorithm.

The comparison of the models is based on the mean squared error (MSE), and the comparative analysis itself was conducted under two main scenarios:

1. Prediction across the entire region

The ANN and the other models were trained and tested on the entire dataset, using atmospheric pressure, temperature, and humidity as input parameters.

2. Prediction after spatial segmentation of input parameters

The input parameter space was divided into 20 separate ring-shaped zones (annuli). The ANN and the alternative models were trained and validated separately for each zone to assess how segmenting the input factors affects prediction accuracy.

Table 9 presents the results of the comparative analysis.

Table 9. Model errors during training and prediction across the entire region (MSE_{Global}) and after segmenting the feature space into 20 annuli ($MSE_{Segmented}$).

Model	Linear Regression	ARIMA	Random Forest	SVR	ANN
MSE_{Global}	4801	1.682×10^3	2749	3152	2856.79
$MSE_{Segmented}$	12.411	3429	8.47	10.59	1.7

When the models are applied directly to the entire dataset (MSE_{Global}), the best performance is observed in random forest and ANNs. The ARIMA model produces the weakest results, indicating that the previously observed linear dependencies in the data are insufficient for accurate predictions. This somewhat justifies our decision to retain air humidity as a variable in the model. After spatial segmentation ($MSE_{Segmented}$), the ANN once again achieves the lowest error (1.7), significantly improving its results compared to other models. Linear regression and support vector regression also show improvements, but do not reach the accuracy of the ANN. Random forest achieves a relatively good MSE (8.47) but still lags behind the ANN in terms of accuracy.

Thus, the final conclusion is that the ANN outperforms all the alternative models considered, especially after the spatial segmentation of the factors. The segmentation of the feature space into annular zones improves the predictive accuracy of all models, but the effect is most pronounced in the ANN (MSE reduction from 2856.58 to 1.7). ARIMA shows the weakest results, suggesting that radon concentrations do not simply follow linear time-dependent processes but instead depend on complex non-linear interactions, which the ANN successfully captures.

Before presenting the test results, it is important to note that both the training and testing of the ANN models were performed on data that had been pre-denoised using wavelet transformation. The predicted radon concentration is compared with actual measurements, which have also undergone wavelet filtering, without the need for reconstruction or the addition of filtered noise components. This ensures full compatibility between the ANN output and the data used for evaluation.

To test the method, we selected data measured at the Yambol station on 3 October 2023, when an earthquake with a magnitude of 3.6 was recorded 75.6 km away. We chose to use a set of 10 single-layer ANNs with 153 neurons, trained with the Levenberg–Marquardt algorithm in the 10 annuli of the initial sample of 57,957 instances. After submitting the test sample of 143 measurements from the day of the earthquake, the radial vectors of the input points were determined, which fell into seven of the annuli, and predictions were made using the seven ANNs trained in these regions. According to the models, fluctuations (Figure 8) in radon concentration (the red line) were observed between 671.34 Bq/m^3 and 1260.01 Bq/m^3, with an average value of 950.72 Bq/m^3. The actual radon measurements at the Yambol station for these values were 695.34 Bq/m^3 and 1398.97 Bq/m^3, with an average value of 1002.70 Bq/m^3. An important conclusion from this is that, on the day of the earthquake, the measured radon concentrations were higher than the predicted normal levels by the model.

Similar studies for the earthquake on October 3 2023 have been conducted for the Dimitrovgrad station ($Latitude_GPS = 42.054814$, $Longitude_GPS = 25.6051$), which is

located 11.9 km from the earthquake's epicenter. A total of 112 measurements were taken at this station.

When applying the set of ANNs, slightly elevated real radon levels are again observed (Figure 9a) compared to the predicted normal levels by the ANNs.

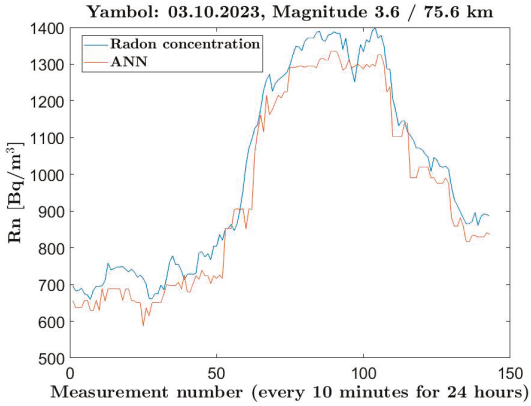


Figure 8. Actual measured radon concentration on the day of the earthquake—3 October 2023 (blue line) and the superimposed model (red line).

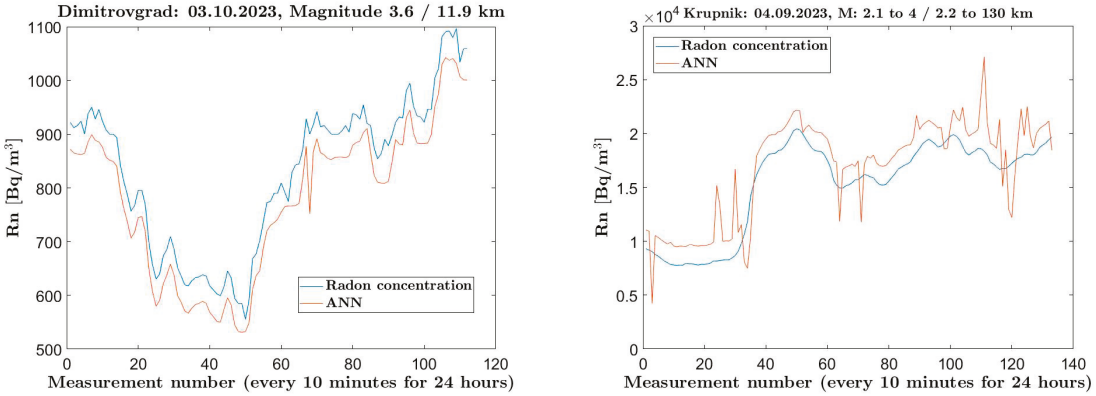


Figure 9. Actual measured radon concentration and the superimposed model on the day of the earthquake for: (**a**) Dimitrovgrad station on 3 October 2023; (**b**) Krupnik station on 4 September 2023.

In contrast to the two mentioned stations, at the Krupnik station ($Latitude_GPS = 41.84748$; $Longitude_GPS = 23.11773$), lower radon levels were observed on the day of a nearby earthquake. On 4 September 2023, multiple earthquakes (17 in total) with magnitudes ranging from 2.1 to 4 were detected, with epicenters located between 2.2 and 130 km from the Krupnik station. Eleven of these quakes had epicenters less than 10 km away. Under these circumstances, contrary to expectations, the actual radon levels were lower than those predicted as normal by the ANNs (Figure 9b). The actual measured radon concentration ranged between 7743 and 20,433 Bq/m^3, with an average value of 15,322.92 Bq/m^3, while the ANNs' prediction for normal behavior of radon concentration indicated higher values: approximately 4249 to 27,107 Bq/m^3, with an average of 16,688.80 Bq/m^3.

These differences in radon anomalies at Krupnik compared to the other stations can be explained by the influence of multiple factors. Krupnik is located in southwestern Bulgaria, near the Kresna Gorge, which is part of the active seismic zone Krupnik-Kresna. This area is known for its complex fault system and high seismic activity. According to Dobrev, the Krupnik-Kresna zone has been the subject of long-term 3D monitoring of active fault structures, highlighting its tectonic complexity and significance [53].

In contrast, Yambol and Dimitrovgrad are located in southeastern Bulgaria, within the Upper Thracian Lowland. This area is also intersected by fault structures, but they are not as active or complex as those in Krupnik. According to the River Basin Management Plan for the East Aegean Region [54], superimposed grabens have formed in the Upper Thracian Depression, filled with neogene and quaternary materials, shaped by major tectonic faults. These geological differences may explain the observed variations in radon concentration behavior between the stations. The complex fault system in the Krupnik area may influence subsurface gas flows, leading to different radon emission behavior compared to the more stable tectonic conditions in Yambol and Dimitrovgrad.

Furthermore, the Krupnik area has documented groundwater presence, confirmed in the Register of Hydrogeological Points from the Quantitative Groundwater Monitoring Network, maintained by the National Institute of Meteorology and Hydrology. This register even lists a tube well in Krupnik with coordinates 41°51′8.355″ N latitude and 23°7′5.856″ E longitude, located within the Struma River Basin [55]. These hydrological conditions may also influence the behavior of radon gas—it is well known that groundwater can retain or dilute radon before it is released into the atmosphere. This could explain why the measured radon concentrations in Krupnik are lower than expected, compared to the predictions of the ANN model.

In addition to geological and hydrological factors, it is important to note that 17 earthquakes of varying magnitudes occurred in Krupnik during the observed periods, making the process even more complex. It is possible that some of these earthquakes caused different shifts in the Earth's layers and changes in gas flows, leading to a dispersion of radon in different directions rather than a clear, unambiguous increase in concentration.

The results from applying the same approach for the 48 h period prior to the day of the considered earthquakes are presented in Figure 10.

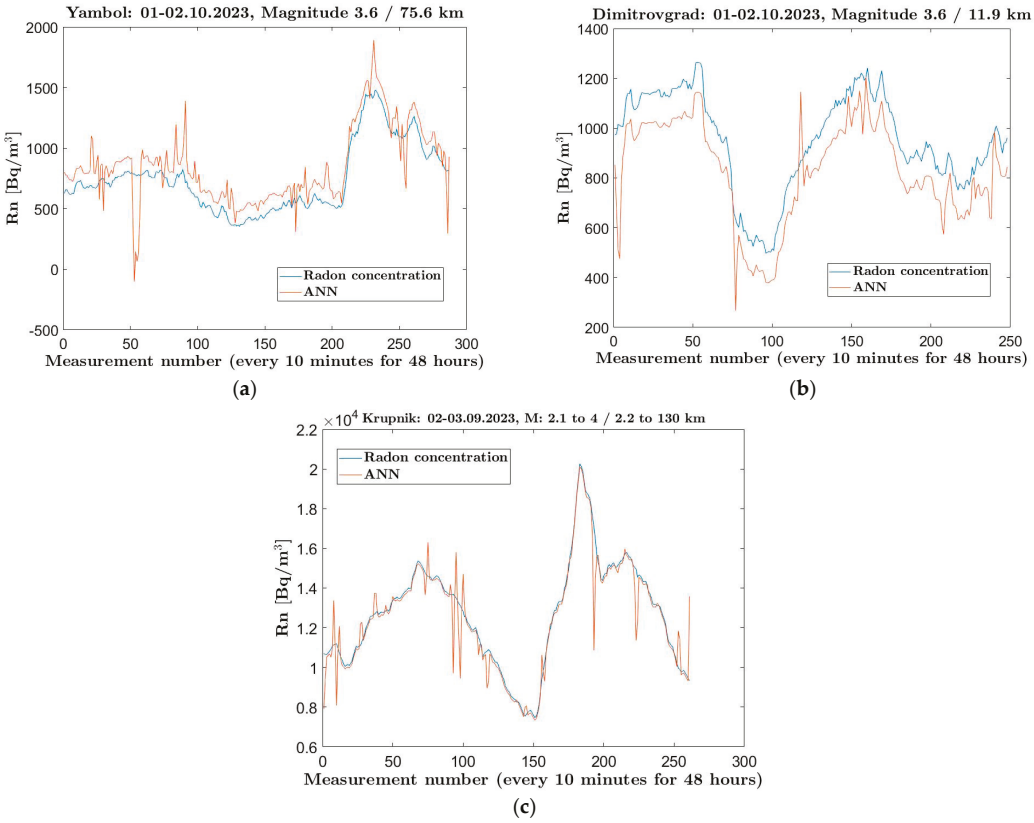


Figure 10. Forecast and actual radon values during the 48 h period before the earthquake (every 10 min) at the following stations: (**a**) Yambol; (**b**) Dimitrovgrad; (**c**) Krupnik.

The results show elevated levels of actual radon concentrations during the 48 h period before the earthquake at the Dimitrovgrad and Krupnik stations. At the Yambol station, however, the actual radon concentrations recorded 48 h before the earthquake were lower than those predicted as normal by the set of ANNs.

The obtained results show that, despite some deviations, the ANNs are able to capture the trend in radon behavior. This indicates that the artificial neurons have identified a pattern in the progression of radon concentration from the preceding and following days and are able to interpolate it for time periods they were not trained on. However, the captured trend may suggest that, in addition to the presumed dependence of radon concentration on earthquakes, there is another factor that strongly influences radon levels, even during periods without seismic activity.

7. Conclusions

The proposed model for dividing input–output data into multidimensional annuli for building multiple ANNs accelerates the training process and creates more efficient ANNs. While training a single ANN on all the input–output data is a time-consuming process that requires extensive resources, training multiple ANNs on smaller datasets allows for the development of a faster and more efficient algorithm. The model enables the use of parallel computations during both the construction and utilization of individual ANNs, further enhancing the overall performance and speed. Another positive aspect of the proposed model is that all ANNs, except for the first and last, are used for interpolation in the prediction process, which helps reduce errors. Extrapolation is only expected with the first and last networks when making predictions for points outside the range of the input samples. In practice, one of the greatest advantages of the proposed model is in single-point predictions or predictions for sets of points with similar radial vectors. In such cases, only one ANN trained in the corresponding annulus can be used, eliminating the need for modeling an ANN trained on the entire initial dataset, which could be of considerable size.

The proposed method is not suitable for small samples or for creating multiple annuli with small samples, as this would result in inefficient ANNs. An interesting problem in this context is finding a function that determines the optimal number of annuli into which a given sample of input–output data should be divided. This would help balance the size of the sample with the performance of the ANNs, ensuring that the model remains both efficient and accurate. The conducted studies make it difficult to provide a definitive answer to the question of whether elevated radon levels are a reliable indicator of an impending earthquake. While the one-day forecasts for the Yambol and Dimitrovgrad stations confirm the preliminary expectation that rising radon levels may be considered an indicator of an earthquake, the opposite behavior is observed at the Krupnik station—decreasing radon levels—even in the presence of multiple nearby tremors. The two-day forecasts for the earthquakes and stations under consideration also do not provide a clear answer regarding radon concentration behavior.

Future research involving multiple earthquakes and a larger number of stations will reveal whether the observed results are consistent for each individual station, i.e., whether there is or is not a relationship between radon levels and an approaching earthquake at different stations. Although external atmospheric influences have been minimized at the stations studied, it would be appropriate to also investigate the impact of seasonal changes as a factor influencing radon concentration. Another future objective set by the team is related to Bayesian methods, which could be useful for further quantifying the probability of radon anomalies before earthquakes. Our idea is to develop models that

reassess probabilities based on new measurements, thus providing a more dynamic approach to analyzing radon concentration time series. Furthermore, we consider it highly beneficial to combine ANNs with Bayesian methods, where the latter would serve as a posterior assessment of the reliability of ANN predictions, adding probabilistic bounds to the forecasted anomalies. We believe that such an approach would enhance the robustness and reliability of predictive models, particularly in the context of varied geological and climatic conditions.

Author Contributions: Conceptualization, K.Y. and E.H.; methodology, K.Y. and E.H.; software, K.Y.; validation, K.Y. and E.H.; formal analysis, K.Y.; investigation, K.Y., E.H. and S.H.; resources, K.Y., E.H. and S.H.; data curation, K.Y.; writing—original draft preparation, K.Y., E.H. and S.H.; writing—review and editing, S.H.; visualization, K.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This study is financed by the European Union—NextGenerationEU, through the National Recovery and Resilience Plan of the Republic of Bulgaria, project № BG-RRP-2.004-0001-C01.

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Grant, R.A.; Raulin, J.P.; Freund, F.T. Changes in animal activity prior to a major (M = 7) earthquake in the Peruvian Andes. *Phys. Chem. Earth Parts A/B/C* **2015**, *85*, 69–77. [CrossRef]
2. Woith, H.; Petersen, G.M.; Hainzl, S.; Dahm, T. Can Animals Predict Earthquakes? *Bull. Seismol. Soc. Am.* **2018**, *108*, 1031–1045. [CrossRef]
3. Sánchez, J.J. Reports of abnormal animal behavior in relation to earth-quakes in Colombia. *Boletín Geológico* **2024**, *51*, 1–19. [CrossRef]
4. Hayakawa, M.; Yamauchi, H. Unusual Animal Behavior as a Possible Candidate of Earthquake Prediction. *Appl. Sci.* **2024**, *14*, 4317. [CrossRef]
5. Kushwaha, A. ML Predictive Model for Earthquakes Integrating Mass, Distance, Gravity, and Magnitude. Master's Thesis, CCT College Dublin, Dublin, Ireland, 2024. Available online: https://arc.cct.ie/ict/63/ (accessed on 23 March 2025).
6. Novikov, V.; Ruzhin, Y.; Sorokin, V.; Yaschenko, A. Space weather and earthquakes: Possible triggering of seismic activity by strong solar flares. *Ann. Geophys.* **2020**, *63*, PA554. [CrossRef]
7. Yan, R.; Chen, X.; Sun, H.; Xu, J.; Zhou, J. A review of tidal triggering of global earthquakes. *Geod. Geodyn.* **2023**, *14*, 35–42. [CrossRef]
8. Volvach, A.E.; Kurbasova, G.S.; Volvach, L.N. Parametric resonance of surface geomagnetic waves as a precursor of local earthquakes. *Phys. Chem. Earth Parts A/B/C* **2025**, *137*, 103802. [CrossRef]
9. Comina, C.; Adinolfi, G.M.; Bertok, C.; Bertea, A.; Giraud, V.; Pieruccini, P. Regional scale shear wave velocity profiles for ground response analyses uncertainties evaluations–the Piedmont Region (NW Italy) Database. *Earth Syst. Sci. Data Discuss.* **2024**, *2024*, 1–21. [CrossRef]
10. Rău, A.; Dinescu, R.; Popa, M.; Radulian, M.; Lungu, M. A Proposed Velocity Model Applied for Banat and Danubian Seismic Zones. *Rom. J. Phys.* **2024**, *in print*.
11. Ji, G.; Lei, J.; Zhao, D. 3-D crustal velocity structure and seismogenic environment of earthquake swarms in the Jiaodong area using the multiple-phase travel-time tomography. *Chin. J. Geophys.* **2025**, *68*, 123–138.
12. Nikkilä, A.; Arvela, H.; Mehtonen, J.; Raitanen, J.; Heinäniemi, M.; Lohi, O.; Auvinen, A. Predicting residential radon concentrations in Finland. *Scand. J. Work. Environ. Health* **2020**, *46*, 278–292. [CrossRef] [PubMed]
13. Ptiček Siročić, A.; Kovač, S.; Stanko, D.; Pejak, I. Dependence of Concentration of Radon on Environmental Parameters-Multiple Linear Regression Model. *Environ. Eng.-Inženjerstvo Okoliša* **2021**, *8*, 17–25. [CrossRef]
14. Dicu, T.; Cucoş, A.; Botoş, M.; Burghele, B.; Florică, Ş.; Baciu, C.; Ştefanm, B.; Bălc, R. Exploring statistical and machine learning techniques to identify factors influencing indoor radon concentration. *Sci. Total Environ.* **2023**, *905*, 167024. [CrossRef] [PubMed]
15. Bulko, M.; Holý, K.; Müllerová, M. Analysis of time series of radon activity concentration in outdoor air and their reconstruction. *Radiat. Prot. Dosim.* **2020**, *191*, 244–249. [CrossRef]

16. Zoran, M.A.; Savastru, R.S.; Savastru, D.M.; Tautan, M.N. Impacts of exposure to air pollution, radon and climate drivers on the COVID-19 pandemic in Bucharest, Romania: A time series study. *Environ. Res.* **2022**, *212*, 113437. [CrossRef]
17. Mir, A.A.; Çelebi, F.V.; Alsolai, H.; Qureshi, S.A.; Rafique, M.; Alzahrani, J.S.; Mahgoub, H.; Hamza, M.A. Anomalies prediction in radon time series for earthquake likelihood using machine learning-based ensemble model. *IEEE Access* **2022**, *10*, 37984–37999. [CrossRef]
18. Mustafa, H.S.; Kheder, M.H.; Mahmood, S.G. Hermite Numerical Method to Estimate the Radon and Radium Effects of the Soil in Bartella Region. *EUREKA Phys. Eng.* **2020**, *1*, 58–64. [CrossRef]
19. Arif, G.E. The Construction of the Best Mathematical Model for Calculating Radon Concentration in Mineral Water. *Int. J. Appl. Sci. Technol.* **2022**, *4*, 45–53. [CrossRef]
20. Miklyaev, P.S.; Petrova, T.B.; Shchitov, D.V.; Sidyakin, P.A.; Murzabekov, M.A.; Tsebro, D.N.; Marennyy, A.M.; Nefedov, N.A.; Gavriliev, S.G. Radon transport in permeable geological environments. *Sci. Total Environ.* **2022**, *852*, 158382. [CrossRef]
21. Valcarce, D.; Alvarellos, A.; Rabuñal, J.R.; Dorado, J.; Gestal, M. Machine Learning-Based Radon Monitoring System. *Chemosensors* **2022**, *10*, 239. [CrossRef]
22. Elío, J.; Petermann, E.; Bossew, P.; Janik, M. Machine learning in environmental radon science. *Appl. Radiat. Isot.* **2023**, *194*, 110684. [CrossRef]
23. Wu, P.Y.; Johansson, T.; Mangold, M.; Sandels, C.; Mjörnell, K. Evaluating the indoor radon concentrations in the Swedish building stock using statistical and machine learning. *J. Phys. Conf. Ser.* **2023**, *2654*, 012086. [CrossRef]
24. Timkova, J.; Fojtikova, I.; Pacherova, P. Bagged neural network model for prediction of the mean indoor radon concentration in the municipalities in Czech Republic. *J. Environ. Radioact.* **2017**, *166*, 398–402. [CrossRef] [PubMed]
25. Duong, V.H.; Ly, H.B.; Trinh, D.H.; Nguyen, T.S.; Pham, B.T. Development of Artificial Neural Network for prediction of radon dispersion released from Sinquyen Mine, Vietnam. *Environ. Pollut.* **2021**, *282*, 116973. [CrossRef] [PubMed]
26. Farai, I.P.; Fajemiroye, J.A.; Oni, O.M.; Aremu, A.A. Artificial neural network modeling of soil gas radon concentration on different lithologies for Southwest Nigeria. *J. Radiat. Res. Appl. Sci.* **2022**, *15*, 346–352. [CrossRef]
27. Haider, T.; Barkat, A.; Hayat, U.; Ali, A.; Awais, M.; Alam, A.; Rehman, A.; Shah, M.A. Identification of radon anomalies induced by earthquake activity using intelligent systems. *J. Geochem. Explor.* **2021**, *222*, 106709. [CrossRef]
28. Feng, X.; Zhong, J.; Yan, R.; Zhou, Z.; Tian, L.; Zhao, J.; Yuan, Z. Groundwater radon precursor anomalies identification by EMD-LSTM model. *Water* **2022**, *14*, 69. [CrossRef]
29. Yan, X.; Shi, Z.; Wang, G.; Zhang, H.; Bi, E. Detection of possible hydrological precursor anomalies using long short-term memory: A case study of the 1996 Lijiang earthquake. *J. Hydrol.* **2021**, *599*, 126369. [CrossRef]
30. Tareen, A.D.K.; Asim, K.M.; Kearfott, K.J.; Rafique, M.; Nadeem MS, A.; Iqbal, T.; Rahman, S.U. Automated anomalous behaviour detection in soil radon gas prior to earthquakes using computational intelligence techniques. *J. Environ. Radioact.* **2019**, *203*, 48–54. [CrossRef]
31. Shepelev, I.E.; Zhegulin, A.V.; Makarov, E.O.; Kosenko, P.O.; Chebrov, D.; Kiroy, V.N.; Arsenyev, F.V. A Study of the Relationship between Dynamics of Subsurface Radon Concentration and Ensuing Seismicity Using Machine Learning Approach. *SSRN* **2024**, 1–13. [CrossRef]
32. Choudhary, M.S.; Gupta, P.; Musau, F. Modelling through different data mining techniques illustrating relativity, interdependence and interconnection of Earthquake precursors worthy for Earthquake prediction. *Int. J. Comput. Sci. Inf. Secur. (IJCSIS)* **2019**, *17*, 94–106.
33. Planinić, J.; Radolić, V.; Vuković, B. Radon as an earthquake precursor. *Nucl. Instrum. Methods Phys. Res. Sect. A Accel. Spectrometers Detect. Assoc. Equip.* **2004**, *530*, 568–574. [CrossRef]
34. Barman, C.; Ghose, D.; Sinha, B.; Deb, A. Detection of earthquake induced radon precursors by Hilbert Huang Transform. *J. Appl. Geophys.* **2016**, *133*, 123–131. [CrossRef]
35. Stoulos, S.; Papadimitriou, E.; Karakostas, V.; Kourouklas, C.; Atac–Nyberg, A.; Wyss, R.; Bäck, T.; Tallini, M.; DeLuca, G. Radon signals in soil gas associated with earthquake occurrence in Greece: Review and perspective. *J. Radioanal. Nucl. Chem.* **2024**, *333*, 6107–6120. [CrossRef]
36. Huang, P.; Lv, W.; Huang, R.; Luo, Q.; Yang, Y. Earthquake precursors: A review of key factors influencing radon concentration. *J. Environ. Radioact.* **2024**, *271*, 107310. [CrossRef]
37. Zhang, S.; Shi, Z.; Wang, G.; Yan, R.; Zhang, Z. Groundwater radon precursor anomalies identification by decision tree method. *Appl. Geochem.* **2020**, *121*, 104696. [CrossRef]
38. Sahoo, S.K.; Katlamudi, M.; Barman, C.; Lakshmi, G.U. Identification of earthquake precursors in soil radon-222 data of Kutch, Gujarat, India using empirical mode decomposition based Hilbert Huang Transform. *J. Environ. Radioact.* **2020**, *222*, 106353. [CrossRef]

39. Nikolopoulos, D.; Cantzos, D.; Alam, A.; Dimopoulos, S.; Petraki, E. Electromagnetic and radon earthquake precursors. *Geosciences* **2024**, *14*, 271. [CrossRef]
40. Walia, V.; Kumar, A.; Fu, C.C. Radon Time Series Data for Earthquake Precursory Studies in Taiwan: An Overview. In *Geohazards: Analysis, Modelling and Forecasting*; Springer: Singapore, 2023; pp. 113–124. [CrossRef]
41. Wang, J.; Zhang, Z. A Bayesian inference based adaptive lane change prediction model. *Transp. Res. Part C Emerg. Technol.* **2021**, *132*, 103363. [CrossRef]
42. Yang, E.; Yang, Q.; Li, J.; Zhang, H.; Di, H.; Qiu, Y. Establishment of icing prediction model of asphalt pavement based on support vector regression algorithm and Bayesian optimization. *Constr. Build. Mater.* **2022**, *351*, 128955. [CrossRef]
43. Zhang, Y.; d'Avigneau, A.M.; Hadjidemetriou, G.M.; de Silva, L.; Girolami, M.; Brilakis, I. Bayesian dynamic modelling for probabilistic prediction of pavement condition. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108637. [CrossRef]
44. Vagdatli, T.; Petroutsatou, K. A Dynamic Bayesian Network for Pavement Roughness Prediction. In *Bituminous Mixtures and Pavements VIII*, 1st ed.; Vagdatli, K.P.T., Ed.; CRC Press: Boca Raton, FL, USA, 2024; p. 9.
45. Grana, D.; Azevedo, L.; De Figueiredo, L.; Connolly, P.; Mukerji, T. Probabilistic inversion of seismic data for reservoir petrophysical characterization: Review and examples. *Geophysics* **2022**, *87*, M199–M216. [CrossRef]
46. Siahkoohi, A.; Rizzuti, G.; Herrmann, F. Deep Bayesian inference for seismic imaging with tasks. *Geophysics* **2022**, *87*, S281–S302. [CrossRef]
47. Jiao, Z.; Shan, X.A. Bayesian Approach for Forecasting the Probability of Large Earthquakes Using Thermal Anomalies from Satellite Observations. *Remote Sens.* **2024**, *16*, 1542. [CrossRef]
48. Zhang, Y.; Chu, J.; Yang, H.Q.; Wu, S.; Zhang, Y.; Chiam, K. Integrated geophysical data and Bayesian evidential learning approach for rockhead estimation and uncertainty quantification. *Eng. Geol.* **2025**, *347*, 107944. [CrossRef]
49. Birulin, S. Spatial-Temporal Patterns of Radon Volume Activity Anomalies Before Tectonic Earthquakes in the South Kuril Region. Ph.D. Thesis, Russian Academy of Sciences, Yekaterinburg, Russia, 2022.
50. Dobrovolsky, I.P. *Theory of Tectonic Earthquake Preparation*; Russian Academy of Sciences: Moscow, Russia, 1991.
51. Yotov, K.; Hadzhikolev, E.; Hadzhikoleva, S.; Cheresharov, S. Finding the optimal topology of an approximating neural network. *Mathematics* **2023**, *11*, 217. [CrossRef]
52. Hunter, D.; Yu, H.; Pukish, M.S., III; Kolbusz, J.; Wilamowski, B.M. Selection of proper neural network sizes and architectures: A comparative study. *IEEE Trans. Ind. Inform.* **2012**, *8*, 228–240. [CrossRef]
53. Dobrev, N. 3D monitoring of active fault structures in the Krupnik-Kresna seismic zone, SW Bulgaria. *Acta Geodyn. Geomater.* **2011**, *8*, 377–388.
54. River Basin Management Plan for the East Aegean Region 2022–2027. East Aegean River Basin Directorate. Available online: https://earbd.bg/files/File/PURB%20-%203/Project_RBMP_2022_2027/%D0%A4%D0%B8%D0%BD%D0%B0%D0%BB%20%D0%A0%D0%B0%D0%B7%D0%B4%D0%B5%D0%BB%201/Razdel%201%20EARBD.pdf (accessed on 23 March 2025). (In Bulgarian).
55. Register of Hydrogeological Points from the Quantitative Groundwater Monitoring Network. National Institute of Meteorology and Hydrology. 2021. Available online: https://hydro.bg/bg/data/Registur_GW.pdf (accessed on 23 March 2025). (In Bulgarian).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Dual Kriging with a Nonlinear Hybrid Gaussian RBF–Polynomial Trend: The Theory and Application to $PM_{2.5}$ Estimation in Northern Thailand

Somlak Utudee [1,2,3], Pharunyou Chanthorn [1,2,3] and Sompop Moonchai [1,2,3,*]

[1] Advanced Research Center for Computational Simulation, Chiang Mai University, Chiang Mai 50200, Thailand; somlak.u@cmu.ac.th (S.U.); pharunyou.chanthorn@cmu.ac.th (P.C.)
[2] Department of Mathematics, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand
[3] Centre of Excellence in Mathematics, MHESI, Bangkok 10400, Thailand
* Correspondence: sompop.m@cmu.ac.th

Abstract: Accurate spatial interpolation of environmental data requires utilizing flexible models that can capture complex spatial patterns. In this paper, we present two improved dual kriging (DK) models comprising a nonlinear trend function that combines Gaussian radial basis functions with a first-order polynomial. The proposed model, DK–RBFP, and its extension, DK–RBFPGA, which includes k-means clustering and a genetic algorithm for parameter optimization, respectively, exhibit enhanced performance in capturing spatial variation. The complete monotonicity of the covariance function and the strict positive definiteness of the coefficient matrix provide theoretical support for the uniqueness of the DK solution. When applied to datasets of $PM_{2.5}$ concentrations for northern Thailand, both models perform better than the conventional DK model using a second-order polynomial trend (DK–POLY), as evidenced by accuracy metrics including the mean absolute percentage error (MAPE), the mean squared error (MSE), and the root mean square error (RMSE). The outcomes indicate that integrating nonlinear trend components with data-driven optimization significantly enhances accuracy and flexibility in environmental spatial predictions.

Keywords: spatial interpolation; dual kriging; kriging with external drift; Gaussian radial basis function; k-mean; distance correlation coefficient; genetic algorithm

MSC: 62P12; 68T09

1. Introduction

Spatial interpolation refers to methods used to estimate the value of a target variable at a specific location based on observed values from nearby sites. Common techniques include inverse distance weighting (IDW), geographically weighted regression (GWR), and kriging. Kriging is a geostatistical approach that provides the best linear unbiased prediction with minimum variance by modeling a spatial process as the sum of a deterministic trend and a stochastic residual. Ordinary kriging (OK), the most widely used variant, assumes a constant mean across the study area. In contrast, kriging with an external drift (KED) incorporates auxiliary variables that are highly correlated with the target variable, defining the trend as a function of these variables to improve estimation accuracy [1].

One of the main computational challenges in KED is that a distinct kriging system must be solved for each interpolation point to calculate the linear estimator weights. To

overcome this, DK, which is derived from KED, uses a single system to compute weights for all points while producing identical estimation results [2]. Compared with OK, DK offers greater flexibility by simultaneously modeling global trends and local variations. Its trend component enables the capturing of complex, nonlinear spatial structures, while solving a single system makes DK more computationally efficient for large datasets. Unlike machine learning (ML) methods such as support vector machines (SVMs) or neural networks, DK explicitly incorporates spatial autocorrelation, providing more interpretable spatial predictions. ML approaches, by contrast, require large datasets and are not inherently designed to account for spatial dependencies [3]. Recent studies have confirmed the effectiveness of DK in relevant research areas [4–7].

The trend component in DK is critical for representing large-scale spatial variation. By explicitly modeling the trend, DK can better capture nonlinear spatial patterns, integrate auxiliary information, and improve prediction accuracy and interpretability. Polynomial functions are commonly used to describe trends in KED and DK; however, they often fail to represent nonlinear interactions between predictors and the response variable. While several studies have explored KED and DK, few focus on nonlinear trend functions. Notable exceptions include Snepvangers et al. [8], who applied a logarithmic trend, and Freier and Lieres [9], who extended universal kriging to include nonlinear trends using Taylor-based linearization. More recently, Baisad et al. [10] employed least squares support vector regression (LSSVR) as the trend in KED. Despite these advances, research on nonlinear trends in DK remains limited. The choice of trend function directly influences the structure of the coefficient matrix in the DK system. This structure determines the existence and uniqueness of the optimal weight solutions. Therefore, selecting an appropriate trend function is a vital step in ensuring that DK provides reliable and consistent estimates.

In this paper, we propose a nonlinear trend function for the DK framework that integrates a radial basis function (RBF) with a first-order polynomial to capture both linear and nonlinear dependencies of the target variable on auxiliary variables. RBFs are widely used in machine learning and geostatistics for interpolation and function approximation, as they model relationships based on distances from a set of centers, enabling flexible representation of nonlinear spatial patterns [11–13]. Commonly used RBFs include the thin-plate spline (TPS), the multiquadric function (MQF), the inverse multiquadric function (IMQF), and the Gaussian function (GF) [14]. Among these, the GF, also known as the Gaussian radial basis function (GRBF), is particularly popular for spatial interpolation, regression [15–17], and kernel-based ML methods [18,19].

The performance of the GRBF depends on two key parameters: the center, which defines where the function peaks, and the bandwidth, which controls its width. Selecting centers typically involves data reduction methods such as k-means clustering or subset selection methods to identify representative points from the dataset [20,21], while bandwidth is often estimated using cross-validation, maximum likelihood, or rule-of-thumb approaches based on inter-center distances [22,23]. One practical approach is to set the bandwidth proportional to the average distance between centers [24]. In this study, k-means clustering is used to choose centers, and a genetic algorithm (GA) is employed to estimate the bandwidth. GAs are population-based optimization techniques inspired by natural selection [25,26], and they have been increasingly applied to estimate parameters in nonlinear regression models [27–30].

Finally, we apply the proposed DK approach with nonlinear trend functions to interpolate $PM_{2.5}$ concentrations using air pressure and relative humidity as auxiliary variables and to compare its performance against DK with a second-order polynomial trend.

The following summarizes the main contributions of this work: (1) the development of a novel nonlinear trend function within the DK framework by combining GRBFs with a first-order polynomial; (2) a theoretical guarantee of the uniqueness of the DK system solution, established by demonstrating the nonsingularity of the coefficient matrix based on the theorems of complete monotonicity and strict positive definiteness; and (3) the implementation of a hybrid approach in which the centers of the GRBFs are determined using k-means clustering, and the bandwidth parameter is estimated using GA.

The remainder of this paper is organized as follows: Section 2 outlines the mathematical backgrounds of KED and DK, as well as the theoretical concepts related to strictly positive definite matrices and completely monotone functions. In Section 3, we present a hybrid nonlinear trend model that combines the Gaussian radial basis function with the first-order polynomial. Section 4 addresses the requirements necessary to ensure the DK system's nonsingularity. Section 5 gives a detailed explanation of the dataset and determines the application of distance correlation coefficients in finding appropriate auxiliary variables. Then, the performance of the proposed DK method is tested by estimating $PM_{2.5}$ concentrations in Northern Thailand. Conclusions and discussions are given in Section 6.

2. Theoretical Background

2.1. Kriging with an External Drift

Let $Z(\boldsymbol{s})$ be a random function at spatial locations $\boldsymbol{s} \in D$, where d is the spatial dimension and $D \subset \mathbb{R}^d$ is the spatial domain of interest. The random function $Z(\boldsymbol{s})$ for KED is modeled as

$$Z(\boldsymbol{s}) = \mu(\boldsymbol{s}) + \epsilon(\boldsymbol{s}), \tag{1}$$

where $\mu(\boldsymbol{s})$ is the drift or trend component and $\epsilon(\boldsymbol{s})$ is a stochastic residual with zero mean and a stationary covariance.

The drift component $\mu(\boldsymbol{s})$ is a deterministic function, often referred to as the trend function. It incorporates external covariates or auxiliary variables, which are predetermined variables having a systematic effect on the underlying spatial process. The trend function $\mu(\boldsymbol{s})$ is expressed as

$$\mu(\boldsymbol{s}) = \sum_{\ell=0}^{L} \alpha_\ell f_\ell(X(\boldsymbol{s})), \tag{2}$$

where α_ℓ denotes an unknown coefficient to be estimated and $X(\boldsymbol{s}) = [X_1(\boldsymbol{s}), \ldots, X_m(\boldsymbol{s})]^T$ represents the vector of m auxiliary variables at location $\boldsymbol{s}$. The function f_ℓ denotes a basis function defined in terms of the auxiliary variables. In general, f_0 is defined as the constant function (e.g., $f_0(x) = 1$), and $L + 1$ is the total number of basis functions.

We can rewrite Equation (1) in the following form:

$$Z(\boldsymbol{s}) = \mathcal{F}^T \boldsymbol{\alpha} + \epsilon(\boldsymbol{s}), \tag{3}$$

where $\mathcal{F} = [f_0(X(\boldsymbol{s})), f_1(X(\boldsymbol{s})), \ldots, f_L(X(\boldsymbol{s}))]^T$ and $\boldsymbol{\alpha} = [\alpha_0, \alpha_1, \ldots, \alpha_L]^T$.

The residual term $\epsilon(\boldsymbol{s})$ is assumed to be a second-order stationary random function with a mean of zero and a covariance function $C(\boldsymbol{h})$ [31], where $\boldsymbol{h}$ is the spatial lag vector between two locations. Specifically, the residual satisfies the following properties:

$$\mathbb{E}(\epsilon(\boldsymbol{s})) = 0, \quad \text{Cov}(\epsilon(\boldsymbol{s}), \epsilon(\boldsymbol{s} + \boldsymbol{h})) = C(\boldsymbol{h}), \tag{4}$$

where $\boldsymbol{s}, \boldsymbol{s} + \boldsymbol{h} \in D \subseteq \mathbb{R}^d$.

Typically, the covariance function is modeled using the variogram $\gamma(\boldsymbol{h})$ of the residual process and is defined as

$$\begin{aligned}\gamma(\boldsymbol{h}) &= \frac{1}{2}\mathrm{Var}(\epsilon(\boldsymbol{s}+\boldsymbol{h}) - \epsilon(\boldsymbol{s})),\\ &= \frac{1}{2}\mathbb{E}(\epsilon(\boldsymbol{s}+\boldsymbol{h}) - \epsilon(\boldsymbol{s}))^2,\end{aligned} \tag{5}$$

which leads to

$$C(\boldsymbol{h}) = \sigma^2 - \gamma(\boldsymbol{h}), \tag{6}$$

where σ^2 is the variance in the residual process [32].

Given n observed values, $Z(\boldsymbol{s}_1), \ldots, Z(\boldsymbol{s}_n)$, at sample locations $\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_n$, the KED estimator at an unobserved location $\boldsymbol{s}_0$ is expressed as a linear combination of the observed values $Z(\boldsymbol{s}_i)$ for $i = 1, \ldots, n$:

$$\widehat{Z}_{KED}(\boldsymbol{s}_0) = \sum_{i=1}^{n} w_i Z(\boldsymbol{s}_i) = \boldsymbol{Z}^T \boldsymbol{\mathcal{W}}, \tag{7}$$

where w_i denotes the KED weight associated with $Z(\boldsymbol{s}_i)$, $\boldsymbol{Z} = [Z(\boldsymbol{s}_1), \ldots, Z(\boldsymbol{s}_n)]^T$ is the vector of the observed values of the target variable, and $\boldsymbol{\mathcal{W}} = [w_1, w_2, \ldots, w_n]^T$ represents the vector of KED weights. The optimal weights are derived by minimizing the estimation error variance under the constraint of unbiasedness, resulting in the following formulation of the KED system:

$$\left[\begin{array}{c|c} \boldsymbol{C} & \boldsymbol{F} \\ \hline \boldsymbol{F}^T & \boldsymbol{0} \end{array}\right] \left[\begin{array}{c} \boldsymbol{\mathcal{W}} \\ \hline \boldsymbol{\mathcal{L}} \end{array}\right] = \left[\begin{array}{c} \boldsymbol{C}_0 \\ \hline \boldsymbol{F}_0 \end{array}\right], \tag{8}$$

where $L = \eta + m$;

$\boldsymbol{0}$ denotes the $(L+1) \times (L+1)$ zero matrix;

$\boldsymbol{\mathcal{L}} = [\lambda_0, \lambda_1, \ldots, \lambda_L]^T$ represents the vector of Lagrange multipliers;

$\boldsymbol{C}_0 = [C(\boldsymbol{s}_1 - \boldsymbol{s}_0), C(\boldsymbol{s}_2 - \boldsymbol{s}_0), \ldots, C(\boldsymbol{s}_n - \boldsymbol{s}_0)]^T$;

$\boldsymbol{F}_0 = [f_0(X(\boldsymbol{s}_0)), f_1(X(\boldsymbol{s}_0)), \ldots, f_L(X(\boldsymbol{s}_0))]^T$;

$$\boldsymbol{C} = \begin{bmatrix} C(\boldsymbol{s}_1 - \boldsymbol{s}_1) & \cdots & C(\boldsymbol{s}_1 - \boldsymbol{s}_n) \\ C(\boldsymbol{s}_2 - \boldsymbol{s}_1) & \cdots & C(\boldsymbol{s}_2 - \boldsymbol{s}_n) \\ \vdots & \ddots & \vdots \\ C(\boldsymbol{s}_n - \boldsymbol{s}_1) & \cdots & C(\boldsymbol{s}_n - \boldsymbol{s}_n) \end{bmatrix};$$

and

$$\boldsymbol{F} = \begin{bmatrix} f_0(X(\boldsymbol{s}_1)) & f_1(X(\boldsymbol{s}_1)) & \cdots & f_L(X(\boldsymbol{s}_1)) \\ f_0(X(\boldsymbol{s}_2)) & f_1(X(\boldsymbol{s}_2)) & \cdots & f_L(X(\boldsymbol{s}_2)) \\ \vdots & \vdots & \ddots & \vdots \\ f_0(X(\boldsymbol{s}_n)) & f_1(X(\boldsymbol{s}_n)) & \cdots & f_L(X(\boldsymbol{s}_n)) \end{bmatrix}.$$

2.2. Dual Kriging

Using the KED system as presented in Equation (8) and assuming an invertible coefficient matrix, the solution can be formalized as [1,2]

$$\begin{bmatrix} \boldsymbol{\mathcal{W}} \\ -- \\ \boldsymbol{\mathcal{L}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{U}} & | & \boldsymbol{\mathcal{V}} \\ -- & + & -- \\ \boldsymbol{\mathcal{V}}^T & | & \boldsymbol{\Omega} \end{bmatrix} \begin{bmatrix} \boldsymbol{C}_0 \\ -- \\ \boldsymbol{F}_0 \end{bmatrix}, \tag{9}$$

where $\boldsymbol{\mathcal{U}}$, $\boldsymbol{\mathcal{V}}$ and $\boldsymbol{\Omega}$ denote matrices of dimensions $n \times n$, $n \times (L+1)$, and $(L+1) \times (L+1)$, respectively. Since the coefficient matrix in Equation (8) is symmetric, its inverse is also symmetric. Consequently, the matrix $\boldsymbol{\mathcal{U}}$ is symmetric.

Substituting the weight vector $\boldsymbol{\mathcal{W}}$, as given in Equation (9), into the estimator formulation in Equation (7) yields the following estimated value:

$$\widehat{Z}_{KED}(\boldsymbol{s}_0) = \boldsymbol{Z}^T \boldsymbol{\mathcal{V}} \boldsymbol{F}_0 + \boldsymbol{Z}^T \boldsymbol{\mathcal{U}} \boldsymbol{C}_0. \tag{10}$$

Let $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ be matrices of dimensions $L \times 1$ and $n \times 1$, respectively; they are defined as follows:

$$\boldsymbol{\alpha}^T = [\alpha_0, \ldots, \alpha_L] = \boldsymbol{Z}^T \boldsymbol{\mathcal{V}} \qquad \text{and} \qquad \boldsymbol{\beta}^T = [\beta_1, \ldots, \beta_n] = \boldsymbol{Z}^T \boldsymbol{\mathcal{U}} = \boldsymbol{Z}^T \boldsymbol{\mathcal{U}}^{\boldsymbol{T}}. \tag{11}$$

As a result, Equation (10) can be reformulated to yield the dual kriging (DK) estimator, which is expressed as follows:

$$\widehat{Z}_{\mathrm{DK}}(\boldsymbol{s}_0) = \boldsymbol{\alpha}^T \boldsymbol{F}_0 + \boldsymbol{\beta}^T \boldsymbol{C}_0. \tag{12}$$

Therefore, the vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ represent the weighting coefficients used in the DK estimator. According to Equation (11), these vectors can be expressed in matrix form as follows:

$$\begin{bmatrix} \boldsymbol{\beta} \\ -- \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{U}} & | & \boldsymbol{\mathcal{A}} \\ -- & + & -- \\ \boldsymbol{\mathcal{V}}^T & | & \boldsymbol{\mathcal{B}} \end{bmatrix} \begin{bmatrix} \boldsymbol{Z} \\ -- \\ \boldsymbol{0}_{L+1} \end{bmatrix}, \tag{13}$$

where $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ denote arbitrary matrices of dimensions $n \times (L+1)$ and $(L+1) \times (L+1)$, respectively. The term $\boldsymbol{0}_{L+1}$ refers to a zero column vector of size $(L+1) \times 1$.

By setting $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{V}}$ and $\boldsymbol{\mathcal{B}} = \boldsymbol{\Omega}$, the coefficient vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are obtained as the solution to the DK system, as shown below.

$$\begin{bmatrix} \boldsymbol{C} & | & \boldsymbol{F} \\ -- & + & -- \\ \boldsymbol{F}^T & | & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta} \\ -- \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} \boldsymbol{Z} \\ -- \\ \boldsymbol{0}_{L+1} \end{bmatrix}. \tag{14}$$

The estimated value of the target variable at the unobserved location $\boldsymbol{s}_0$ can be computed by substituting the coefficient vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ into Equation (12).

The solution of the individual systems used to determine the optimal weights in dual kriging (DK) depends on the invertibility of the coefficient matrices, which ensures a unique solution. Achieving this property requires appropriate selection of variogram or covariance functions and trend models. Myers [33] noted that these coefficient matrices are nonsingular if the covariance function is expressed in terms of linearly independent, real-valued functions of the trend model and is strictly positive definite with respect to the basis functions. In the following section, we introduce the definitions and theorems of

positive definite matrices and completely monotone functions to establish the conditions necessary for the invertibility of the dual kriging system.

2.3. The Relationship Between the Positive Definite Matrix and the Completely Monotone Function

The definitions and theorems presented in this section are adapted from the foundational work [34,35], which provides a comprehensive treatment of positive definite matrices, completely monotone functions, and their applications in approximation theory.

Definition 1. *Let $\mathscr{C}$ be an $n \times n$ symmetric matrix and $u \in \mathbb{R}^n$.*

1. *$\mathscr{C}$ is **positive definite** (denoted by $\mathscr{C} \geq 0$) if*

$$u^T \mathscr{C} u \geq 0 \quad \text{for all } u \neq 0.$$

2. *$\mathscr{C}$ is **strictly positive definite** (denoted by $\mathscr{C} > 0$) if*

$$u^T \mathscr{C} u > 0 \quad \text{for all } u \neq 0.$$

Definition 2. *Let $f : [0, \infty) \to \mathbb{R}$. The function f is called a **completely monotone function** if*

1. *$f \in C[0, \infty) \cap C^\infty(0, \infty)$;*
2. *$(-1)^k f^{(k)}(h) \geq 0$ for all $k \in \mathbb{N}$ and $h \in (0, \infty)$.*

To illustrate the concept of complete monotonicity, we now present a specific function that satisfies the conditions outlined above.

Example 1. *The function $f(h) = e^{-\sqrt{h}/a}$ is completely monotone for every $a > 0$.*

We see that the function $f(h)$ is continuous on $[0, \infty)$. We will verify that $(-1)^k f^{(k)} \geq 0$ for all $k \in \mathbb{N}$ by strong induction. For $h > 0$, it is obvious that

$$(-1) f'(h) = \frac{1}{2a\sqrt{h}} e^{-\sqrt{h}/a} \geq 0.$$

Next, we assume that

$$(-1)^i f^{(i)}(h) \geq 0 \quad \text{for all } i = 1, 2, \ldots, k.$$

Using the Leibniz rule, where

$$(fg)^{(k)} = \sum_{i=0}^{k} \binom{k}{i} f^{(k-i)} g^{(i)},$$

we compute the following:

$$(-1)^{k+1} f^{(k+1)}(h) = \frac{1}{2a} \sum_{i=0}^{k} \binom{k}{i} \frac{(2i+1)!!}{2^i h^{(2i+1)/2}} (-1)^{k-i} f^{(k-i)}(h) \geq 0,$$

where $(2i+1)!! = (2i+1)(2i-1)(2i-3) \cdots 3 \cdot 1$. We then conclude that

$$(-1)^k f^{(k)}(h) \geq 0 \quad \text{for all } k \in \mathbb{N}.$$

The following theorem provides a powerful criterion for ensuring strict positive definiteness of matrices formed from completely monotone functions.

Theorem 1 (Schoenberg's Theorem)**.** *If f is a completely monotone function and not constant on $[0,\infty)$, then for any distinct points $\boldsymbol{s}_1,\ldots,\boldsymbol{s}_n$, the matrix*

$$\mathscr{C} = \left[f(\|\boldsymbol{s}_i - \boldsymbol{s}_j\|^2)\right]$$

is strictly positive definite.

We now turn to a related result concerning the nonsingularity of a specific matrix structure known as the bordered Gramian matrix.

Theorem 2. *Let $\mathscr{C}$ be a positive definite matrix of size $n \times n$, and let $\mathscr{F}$ be an $n \times (L+1)$ matrix. The* ***bordered Gramian matrix***

$$\begin{bmatrix} \mathscr{C} & | & \mathscr{F} \\ -- & + & -- \\ \mathscr{F}^T & | & \mathbf{0} \end{bmatrix}$$

is nonsingular if and only if

$$\operatorname{rank}(\mathscr{F}) = L+1 \quad \textit{and} \quad \mathscr{C} + \mathscr{F}\mathscr{F}^T > 0.$$

3. A New Approach to Nonlinear Trend Modeling in Dual Kriging

This section presents the trend functions employed in the DK approach adopted in this study. These include a second-order polynomial function and the proposed nonlinear trend function, which were constructed by integrating GRBFs with a first-order polynomial. Additionally, the parameter estimates obtained from these two distinct trend models are presented.

Let the dataset at each sample location $\boldsymbol{s}_i \in \mathbb{R}^d$ be denoted by $\{(X(\boldsymbol{s}_i), Z(\boldsymbol{s}_i))\}_{i=1}^n$, where $X(\boldsymbol{s}_i) = [X_1(\boldsymbol{s}_i),\ldots,X_m(\boldsymbol{s}_i)]^T \in \mathbb{R}^m$ is a vector comprising m auxiliary variables and $Z(\boldsymbol{s}_i)$ denotes the observed response at location $\boldsymbol{s}_i$.

The conventional trend functions employed in KED and DK methods are typically first-order (linear) or second-order (quadratic) polynomial models. In this study, we focus on the second-order polynomial model, which is defined as follows [36,37]:

$$\mu(\boldsymbol{s}) = b_0 + \sum_{j=1}^{m} b_j X_j(\boldsymbol{s}) + \sum_{i=1}^{m} b_{ii} X_i(\boldsymbol{s})^2 + \sum_{i=1}^{m-1}\sum_{j=i+1}^{m} b_{ij} X_i(\boldsymbol{s}) X_j(\boldsymbol{s}). \tag{15}$$

Given that $\frac{(m+1)(m+2)}{2} \leq n$, the unknown coefficients in Equation (15) can be estimated using the ordinary least squares (OLS) method [38,39], which is given by

$$\widehat{\mathcal{B}} = (\mathcal{X}^T\mathcal{X})^{-1}\mathcal{X}^T\mathbf{Z}, \tag{16}$$

where $\widehat{\mathcal{B}}$ is an $\frac{(m+1)(m+2)}{2} \times 1$ matrix of estimated coefficients,

$\mathcal{X}$ is an $n \times \frac{(m+1)(m+2)}{2}$ matrix whose rows are the vectors

$\mathcal{X}(\boldsymbol{s}_i) = [1, X_1(\boldsymbol{s}_i),\ldots,X_m(\boldsymbol{s}_i), X_1^2(\boldsymbol{s}_i),\ldots,X_m^2(\boldsymbol{s}_i), X_1(\boldsymbol{s}_i)X_2(\boldsymbol{s}_i),\ldots,X_{m-1}(\boldsymbol{s}_i)X_m(\boldsymbol{s}_i)]$,

and $\mathbf{Z} = [Z(\boldsymbol{s}_1), Z(\boldsymbol{s}_2),\ldots,Z(\boldsymbol{s}_n)]^T$.

In addition, the matrix $\mathcal{X}$ has full column rank.

The proposed nonlinear trend function is created by integrating GRBFs with a first-order polynomial. GRBFs are widely used for modeling and estimating functions in both spatial and temporal contexts. Their formal definition is provided as follows [15]:

$$\varphi_i(X(\boldsymbol{s}), P_i) = \varphi_i(X(\boldsymbol{s}), \Psi_i, \sigma_i) = \exp\left[-\left(\frac{\|X(\boldsymbol{s}) - \Psi_i\|}{\sigma_i}\right)^2\right]. \tag{17}$$

In Equation (17), the symbol $\|\cdot\|$ represents some norm on $\mathbb{R}^m$, typically the Euclidean norm. Each parameter vector P_i is defined as $P_i = [\Psi_i, \sigma_i]$, where Ψ_i and σ_i denote the center (or mean) and the bandwidth (or scale) parameters, respectively.

Given the effectiveness of GRBFs in capturing highly nonlinear relationships, the proposed trend model is augmented with a first-order polynomial to account for the linear component of the trend. According to Equation (1), the resulting trend function is defined as follows:

$$\mu(\boldsymbol{s}) = a_0 + \sum_{i=1}^{m} a_i X_i(\boldsymbol{s}) + \sum_{j=1}^{\eta} b_j \varphi_j(X(\boldsymbol{s}), P_j), \tag{18}$$

where a_i and b_j denote unknown coefficients for $i = 0, 1, \ldots, m$ and $j = 1, \ldots, \eta$, respectively. The vector $X(\boldsymbol{s}) = [X_1(\boldsymbol{s}), \ldots, X_m(\boldsymbol{s})]^T$ represents the m auxiliary variables at location $\boldsymbol{s}$, while $\varphi_j(X(\boldsymbol{s}), P_j)$ is the radial basis function with parameter P_j. Additionally, η denotes the total number of radial basis functions.

In this study, we consider the parameters of the trend model(Model (18)) in two cases. In the first case, Ψ_j and σ_j are estimated using the method proposed in [15]. Subsequently, the parameters a_i and b_j for $i = 0, 1, \ldots, m$ and $j = 1, \ldots, \eta$ are determined using the least squares method.

In the second case, the trend model is treated as a nonlinear model. The parameter Ψ_j is obtained using the same method as in the first case, while the parameters σ_j, a_i, and b_j for $i = 0, 1, \ldots, m$ and $j = 1, \ldots, \eta$ are estimated using the genetic algorithm. The model parameters are estimated using a genetic algorithm implemented via the ga function in MATLAB (Version 2018a). The algorithm is configured with a population size of 30, a crossover probability of 0.8, a mutation probability of 0.2, and a maximum of 20,000 generations. Stochastic uniform selection is employed as the selection method, combined with scattered crossover and Gaussian mutation strategies.

In the first case, for the purpose of estimating the parameters Ψ_j and σ_j, the dataset is partitioned into η clusters, denoted by $\{G_1, \ldots, G_\eta\}$. In this study, k-means clustering is employed to partition the dataset into n distinct clusters. As an unsupervised machine learning algorithm, k-means groups data points based on their similarity by assigning them to the nearest cluster centroid [40]. Following the method proposed in [15], the parameters Ψ_j and σ_j are estimated as follows:

$$\Psi_j = \frac{1}{n_j} \sum_{X_\alpha \in G_j} X_\alpha, \tag{19}$$

and

$$\sigma_j^2 = \frac{1}{n_j} \sum_{X_\alpha \in G_j} ||X_\alpha - \Psi_j||^2, \tag{20}$$

where n_j denotes the number of observations belonging to the j-th cluster G_j. Subsequently, Equation (18) can be expressed in matrix form as

$$\mu(\boldsymbol{s}) = Y^T(\boldsymbol{s})\boldsymbol{\theta}, \tag{21}$$

where $Y(\boldsymbol{s}) = [1, X_1(\boldsymbol{s}), \ldots, X_m(\boldsymbol{s}), \varphi_1(X(\boldsymbol{s})), \varphi_2(X(\boldsymbol{s})), \ldots, \varphi_\eta(X(\boldsymbol{s}))]^T$ and $\boldsymbol{\theta} = [a_0, a_1, \ldots, a_m, b_1, \ldots, b_\eta]^T$.

Assuming that $1 + m + \eta \leq n$, the parameter matrix $\boldsymbol{\theta}$ can be estimated using the least squares method, and it is given by

$$\widehat{\boldsymbol{\theta}} = \left(\mathbf{Y}^T\mathbf{Y}\right)^{-1}\mathbf{Y}^T\mathbf{Z}, \tag{22}$$

where $\mathbf{Y}$ represents an $n \times (1 + m + \eta)$ matrix whose rows are the vectors $Y^T(\boldsymbol{s}_i)$. Furthermore, $\mathbf{Y}$ is of full column rank.

4. Nonsingularity of the Dual Kriging System

The following result combines the preceding theoretical results to establish conditions under which the bordered Gramian matrix, which was constructed from a completely monotone function and a full-rank matrix, is guaranteed to be nonsingular.

Theorem 3. *Let $X(\boldsymbol{s}_1), X(\boldsymbol{s}_2), \ldots, X(\boldsymbol{s}_n)$ be distinct points in $\mathbb{R}^m$, and assume $n \geq 1 + m + \eta$. Define the matrix $\boldsymbol{F} \in \mathbb{R}^{n\times(1+m+\eta)}$ by*

$$\boldsymbol{F} = \begin{bmatrix} 1 & X_1(\boldsymbol{s}_1) & \cdots & X_m(\boldsymbol{s}_1) & \varphi_1(X(\boldsymbol{s}_1)) & \cdots & \varphi_\eta(X(\boldsymbol{s}_1)) \\ 1 & X_1(\boldsymbol{s}_2) & \cdots & X_m(\boldsymbol{s}_2) & \varphi_1(X(\boldsymbol{s}_2)) & \cdots & \varphi_\eta(X(\boldsymbol{s}_2)) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_1(\boldsymbol{s}_n) & \cdots & X_m(\boldsymbol{s}_n) & \varphi_1(X(\boldsymbol{s}_n)) & \cdots & \varphi_\eta(X(\boldsymbol{s}_n)) \end{bmatrix},$$

where for $j = 1, \ldots, \eta$,

$$\varphi_j(X(\boldsymbol{s})) \;=\; \exp\left[-\left(\frac{\| X(\boldsymbol{s}) - \Psi_j \|}{\sigma_j}\right)^2\right], \quad \sigma_j > 0,$$

are Gaussian radial basis functions with distinct centers $\Psi_j \in \mathbb{R}^m$. Then $\operatorname{rank}(\boldsymbol{F}) = 1 + m + \eta$.

Proof. We will show that the columns of $\boldsymbol{F}$ are linearly independent. To establish linear independence, it suffices to show that the homogeneous system

$$\boldsymbol{F}\,\mathbf{c} = 0$$

has only the trivial solution $\mathbf{c} = 0$, where

$$\mathbf{c} = \left[c_0, c_1, \ldots, c_m,\ c_{m+1},\ c_{m+2}, \ldots, c_{m+\eta}\right]^T \in \mathbb{R}^{1+m+\eta}.$$

For each $i = 1, \ldots, n$, the relation $\boldsymbol{F}\mathbf{c} = 0$ can be written as

$$c_0 \;+\; \sum_{k=1}^{m} c_k\, X_k(s_i) + \sum_{j=1}^{\eta} c_{m+j}\, \varphi_j(X(s_i)) \;=\; 0. \tag{23}$$

We decompose the matrix $\boldsymbol{F}$ as

$$\boldsymbol{F} = \left[\ \mathbf{1}\ \middle|\ P\ \middle|\Phi\ \right],$$

where $\mathbf{1} \in \mathbb{R}^n$ is the all-ones column, $P \in \mathbb{R}^{n\times m}$ with $P_{ik} = X_k(s_i)$, and $\Phi \in \mathbb{R}^{n\times\eta}$ with $\Phi_{ij} = \varphi_j(X(s_i))$, .

For distinct sites $X(\boldsymbol{s}_i)$ and GRBFs with distinct centers and $\sigma_j > 0$, Φ has full column rank when $n \geq \eta$. Furthermore, for distinct sites with $n \geq \eta + m + 1 > m + 1$, the polynomial block $[\mathbf{1} \mid P]$ has rank $m + 1$, since it spans the affine-linear functions in $\mathbb{R}^m$.

Case 1: Suppose some $c_j \neq 0$ for a $j \in \{m + 1, \ldots, m + \eta\}$. Then by (23), the nontrivial RBF combination $\sum_{j=1}^{\eta} c_{m+j}\varphi_j$ coincides with an affine-linear function on the finite set $\{X(s_i)\}_{i=1}^{n}$. Since Φ and $[\mathbf{1} \mid P]$ are of full rank and span disjoint subspaces under the stated conditions, this leads to a contradiction. Hence this case is impossible.

Case 2: Assume $c_{m+1} = \cdots = c_{m+\eta} = 0$. Then (23) reduces to

$$c_0 + \sum_{k=1}^{m} c_k \, X_k(s_i) \;=\; 0, \qquad i = 1, \ldots, n.$$

Because $n > m + 1$ and $[\mathbf{1} \mid P]$ has full column rank $1 + m$, it follows that $c_0 = c_1 = \cdots = c_m = 0$.

Since both cases imply $\mathbf{c} = 0$, we conclude that the columns of $\boldsymbol{F}$ are linearly independent. □

Remark 1. *This theorem guarantees uniqueness in interpolation schemes that combine Gaussian radial basis functions with an affine polynomial tail. The assumptions of distinct centers and sufficiently many sample sites are essential for ensuring full rank in both the GRBF and polynomial blocks.*

On the other hand, this study assumes isotropy, implying that the variogram is a function of the distance (i.e., the magnitude of the lag vector $\boldsymbol{h}$) rather than its direction. The exponential variogram is used in its isotropic form without a nugget effect. This model is commonly applied in a wide range of spatial analysis contexts and is defined as follows [31]:

$$\gamma(h) = k_2\Big(1 - e^{-h/k_1}\Big), \tag{24}$$

where $h = \|\boldsymbol{h}\|$ is the separation distance, k_1 is the range parameter, and k_2 represents the sill, corresponding to the variance in the spatial process.

The variogram model is developed using spatial data in conjunction with an empirical variogram estimator, which captures the degree of spatial dependence between observations across different distances. In this study, we adopt the empirical variogram estimator proposed by Matheron [41], a widely recognized and commonly applied approach in geostatistics.

Theorem 4. *The DK system in Equation (14) admits a unique solution.*

Proof. From Equation (6), the covariance function can be derived as follows:

$$C(h) = k_2 - \Big[k_2(1 - e^{-h/k_1})\Big] = k_2 e^{-h/k_1}. \tag{25}$$

According to Example 1, the function

$$f(h) = k_2 e^{-\sqrt{h}/k_1}$$

is a completely monotone function and not constant on $[0, \infty)$. According to Schoenberg's Theorem, for any distinct points $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n$, the matrix

$$\boldsymbol{C} = \Big[f(\|\boldsymbol{s}_i - \boldsymbol{s}_j\|^2)\Big] = \big[C(\boldsymbol{s}_i - \boldsymbol{s}_j)\big]$$

in Equation (14) is strictly positive definite. Since the matrix $\boldsymbol{F}\boldsymbol{F}^T$ is always positive definite, it follows that the sum $\boldsymbol{C} + \boldsymbol{F}\boldsymbol{F}^T$ is strictly positive definite. Furthermore, Theorem 3 assures that $\text{rank}(\boldsymbol{F}) = 1 + m + \eta = L + 1$. Combining these two facts, Theorem 2 applies and ensures that the coefficient matrix in the dual kriging system (Equation (14)) is nonsingular.

Consequently, the DK system admits a unique solution. □

5. Application to $PM_{2.5}$ Spatial Estimation in Northern Thailand

5.1. Study Area and Data Description

In this study, fine particulate matter ($PM_{2.5}$) pollution over northern Thailand is investigated. This region is characterized by its mountainous landscape and persistent air quality issues, particularly during the dry season [42,43]. Northern Thailand consists of 15 provinces and is influenced by a mix of complex topography, intensive agricultural practices, and seasonal burning activities, which together contribute to severe haze events and transboundary air pollution [44,45]. As illustrated in Figure 1, the study area encompasses several provinces in the northern part of the country. The inset map provides a national reference, while the enlarged panel highlights the specific provinces included in the analysis, shown in orange. Fifteen monitoring stations, each corresponding to a province and marked with green circles, were employed to collect data on $PM_{2.5}$ concentrations along with associated meteorological variables.

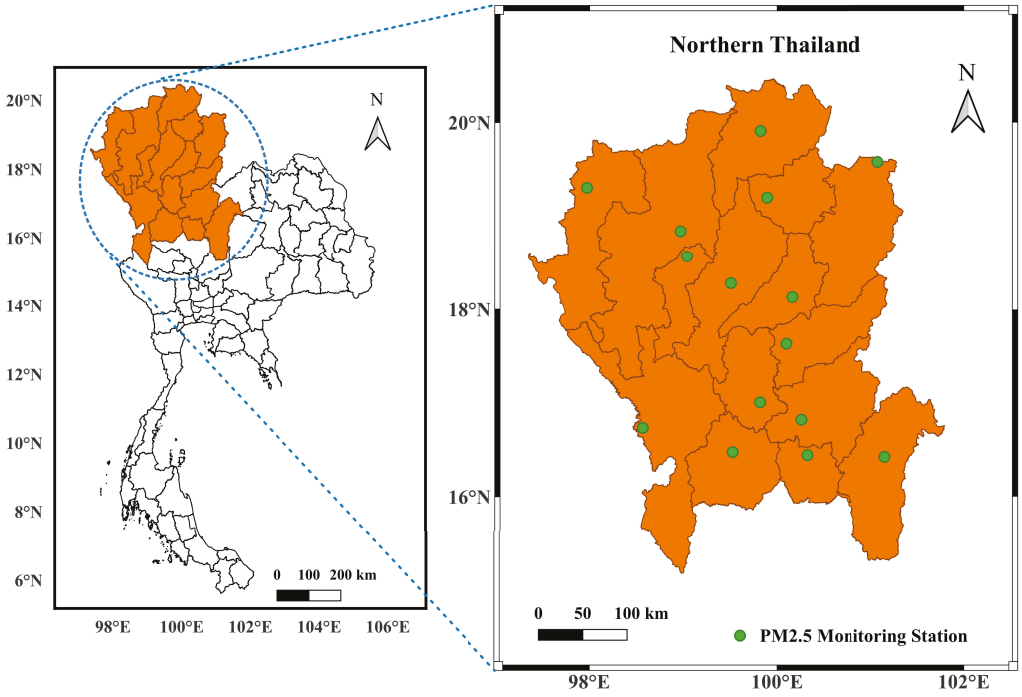


Figure 1. Distribution of $PM_{2.5}$ monitoring stations in northern Thailand.

Model accuracy was assessed using data collected during March and April of both 2023 and 2024. $PM_{2.5}$ concentration data ($\mu g/m^3$) were obtained from the Pollution Control Department (PCD), while meteorological variables, including air pressure (hPa) and relative humidity (%), were retrieved from the Thai Meteorological Department (TMD). All datasets were aggregated weekly to reduce short-term variability and highlight broader spatial and temporal patterns.

The relationship between $PM_{2.5}$ concentrations and meteorological conditions was investigated using the distance correlation coefficient (DCC). This method enabled the

identification of spatial dependencies between air pollution levels and meteorological factors across the study region.

For model validation, the 15 monitoring stations were divided into two subsets. Twelve stations were used for model training, while the remaining three served as an independent test set. This partitioning allowed for an objective evaluation of the model's predictive performance and ensured a reliable basis for spatial interpolation of $PM_{2.5}$ in areas lacking direct observations.

5.2. Performance Metrics and Quantitative Evaluation

This study compares the accuracy of the DK constructed using three different trend functions. The first model (DK–POLY) utilizes a second-order polynomial trend, with its parameters estimated via ordinary least squares (OLS). The second approach (DK–RBFP) incorporates a non-linear trend function based on the GRBF combined with a first-order polynomial function. In this method, the mean and scale parameters of the GRBF are estimated using the methods described in Equations (19) and (20), respectively, while the remaining parameters are estimated using OLS. The third model (DK–RBFPGA) also employs a GRBF-based trend combined with a first-order polynomial function, where the mean parameters are estimated using the approach in Equation (19), and the remaining parameters are optimized using the GA.

For the comparison of the performance of the three models, a k-fold cross-validation technique was used [46]. The dataset was divided into five equal sets randomly (folds). For each of the five iterations, one fold was used as the validation set, while the remaining four were used for model training. This procedure guaranteed that each fold was used as the validation set once. Model accuracy was then computed by averaging over all iterations. Three metrics of performance were used to assess model performance, namely the mean absolute percentage error (MAPE) [10], the mean squared error (MSE) [47], and the root mean square error (RMSE) [10], as described below:

$$\text{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{Z(\boldsymbol{s}_i) - \widehat{Z}(\boldsymbol{s}_i)}{Z(\boldsymbol{s}_i)}\right| , \tag{26}$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Z(\boldsymbol{s}_i) - \widehat{Z}(\boldsymbol{s}_i))^2 , \tag{27}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Z(\boldsymbol{s}_i) - \widehat{Z}(\boldsymbol{s}_i))^2} , \tag{28}$$

where $Z(\boldsymbol{s}_i)$ denotes the observed value at location $\boldsymbol{s}_i$, $\widehat{Z}(\boldsymbol{s}_i)$ is the estimated value at location $\boldsymbol{s}_i$, and n represents the number of observations.

5.3. The Distance Correlation Coefficient

The distance correlation coefficient (DCC), which was introduced by Székely et al. [48], measures the statistical dependence between two random vectors, $U \in \mathbb{R}^p$ and $V \in \mathbb{R}^q$, and is designed to detect both linear and nonlinear relationships. The distance correlation coefficient has proven to be a flexible and effective statistical measure, with recent research highlighting its broad applicability and extension across various fields [49–53]. Its strength lies in its ability to detect both linear and nonlinear relationships, making it particularly well-suited for analyzing complex and high-dimensional data [54].

Given a sample of n paired observations $\{(U_i, V_i)\}_{i=1}^{n}$, the calculation is carried out as follows [48,55]:

First, the empirical distance covariance, $d\mathcal{C}(U, V)$, is computed as:

$$d\mathcal{C}(U, V) = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}B_{ij}} \tag{29}$$

where A_{ij} and B_{ij} are the doubly centered Euclidean distances within the samples of U and V, respectively. The centered distance matrices A_{ij} and B_{ij} are computed by double-centering the pairwise Euclidean distance matrices of the variables U and V, respectively. Specifically, the entries of the centered distance matrix for U are given by

$$A_{ij} = \|U_i - U_j\|_{\mathbb{R}^p} - \frac{1}{n}\sum_{l=1}^{n}\|U_i - U_l\|_{\mathbb{R}^p} - \frac{1}{n}\sum_{k=1}^{n}\|U_k - U_j\|_{\mathbb{R}^p} + \frac{1}{n^2}\sum_{k=1}^{n}\sum_{l=1}^{n}\|U_k - U_l\|_{\mathbb{R}^p} \tag{30}$$

Similarly, the centered distance matrix for V is defined as

$$B_{ij} = \|V_i - V_j\|_{\mathbb{R}^q} - \frac{1}{n}\sum_{l=1}^{n}\|V_i - V_l\|_{\mathbb{R}^q} - \frac{1}{n}\sum_{k=1}^{n}\|V_k - V_j\|_{\mathbb{R}^q} + \frac{1}{n^2}\sum_{k=1}^{n}\sum_{l=1}^{n}\|V_k - V_l\|_{\mathbb{R}^q} \tag{31}$$

In these equations, the notations $\|\cdot\|_{\mathbb{R}^p}$ and $\|\cdot\|_{\mathbb{R}^q}$ denote the Euclidean norm in p-dimensional and q-dimensional spaces, respectively.

Next, the empirical distance variances for U and V are calculated as

$$d\mathcal{V}(U) = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}^2} \tag{32}$$

and

$$d\mathcal{V}(V) = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} B_{ij}^2}\,. \tag{33}$$

Finally, the empirical distance correlation coefficient is given by

$$d\mathcal{R}(U, V) = \begin{cases} \dfrac{d\mathcal{C}(U, V)}{\sqrt{d\mathcal{V}(U)d\mathcal{V}(V)}}, & \text{if } d\mathcal{V}(U) > 0 \text{ and } d\mathcal{V}(V) > 0, \\ 0\,, & \text{otherwise.} \end{cases} \tag{34}$$

The distance correlation coefficient $d\mathcal{R}(U, V)$ takes values between 0 and 1, where a value of 0 indicates complete independence between U and V, while a value of 1 indicates perfect dependence, regardless of whether the relationship is linear or nonlinear.

5.4. Results

The trend function of DK relies on auxiliary variables, which must exhibit a relationship with the target variable, in this case, the $PM_{2.5}$ concentration. As such, it is essential to quantify the relationship between the candidate auxiliary variables and the $PM_{2.5}$ concentration. In this study, the distance correlation coefficient was employed to evaluate this relationship, with air pressure and relative humidity identified as the candidate auxiliary variables.

Tables 1 and 2 present the weekly distance correlations between $PM_{2.5}$ concentrations and two auxiliary meteorological variables, namely air pressure and relative humidity, during March and April for the years 2023 and 2024, respectively.

Table 1. Weekly distance correlation coefficients for $PM_{2.5}$ concentrations with air pressure and relative humidity during March and April 2023, based on the complete dataset.

Auxiliary Variable	Month	Week 1	Week 2	Week 3	Week 4	Average
Air Pressure	March 2023	0.522	0.470	0.449	0.414	0.464
	April 2023	0.458	0.393	0.338	0.737	0.481
Relative Humidity	March 2023	0.424	0.440	0.401	0.345	0.402
	April 2023	0.428	0.490	0.417	0.477	0.453

Table 1 shows the correlation results for 2023. It shows that air pressure had a moderate effect on $PM_{2.5}$ concentrations, with average correlation values of 0.464 in March and 0.481 in April. In Week 4 of April, there was a peak, with the correlation reaching 0.737. This suggests that air pressure may have a short-term effect on $PM_{2.5}$ levels. In contrast, relative humidity demonstrated weaker correlations, with average values of 0.402 in March and 0.453 in April, reflecting a comparatively lower association with $PM_{2.5}$ during this period.

Table 2. Weekly distance correlation coefficients for $PM_{2.5}$ concentrations with air pressure and relative humidity during March and April 2024, based on the complete dataset.

Auxiliary Variable	Month	Week 1	Week 2	Week 3	Week 4	Average
Air Pressure	March 2024	0.446	0.511	0.604	0.574	0.534
	April 2024	0.368	0.542	0.459	0.440	0.452
Relative Humidity	March 2024	0.546	0.587	0.762	0.905	0.700
	April 2024	0.744	0.499	0.424	0.445	0.528

In comparison, Table 2 shows that the 2024 results exhibited generally stronger correlations, particularly for relative humidity. In March 2024, the correlation between relative humidity and $PM_{2.5}$ climbed progressively from Week 1 to Week 4, reaching a high of 0.905 and an overall monthly average of 0.700. Although the average correlation dropped slightly to 0.528 in April, it remained higher than the values observed in 2023. Air pressure also showed stronger correlations in March 2024, averaging 0.534, though it decreased to 0.452 in April. These findings highlight that the influence of meteorological factors on $PM_{2.5}$ concentrations can vary from year to year and suggest that relative humidity may play a more significant role under certain atmospheric conditions.

The results indicate that the correlations between $PM_{2.5}$ concentrations and the auxiliary variables, air pressure and relative humidity, vary across different periods and years. Relative humidity generally shows a stronger and more consistent association with $PM_{2.5}$, while air pressure demonstrates a moderate relationship that fluctuates more noticeably over time.

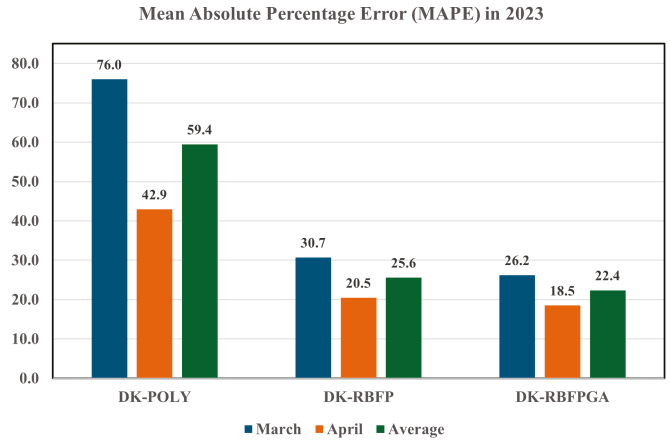
Figures 2–4 present a comparative evaluation of the predictive capabilities of DK with three different trend models: DK–POLY, DK–RBFP, and DK–RBFPGA. The assessment focuses on their performance during March and April of both 2023 and 2024. Each figure highlights a specific error metric, including the MAPE, MSE, and RMSE, to assess the accuracy of each model.

Figure 2 presents a comparison of the MAPE values for the three models across the specified months and years. In 2023, DK–POLY recorded the highest MAPE, indicating

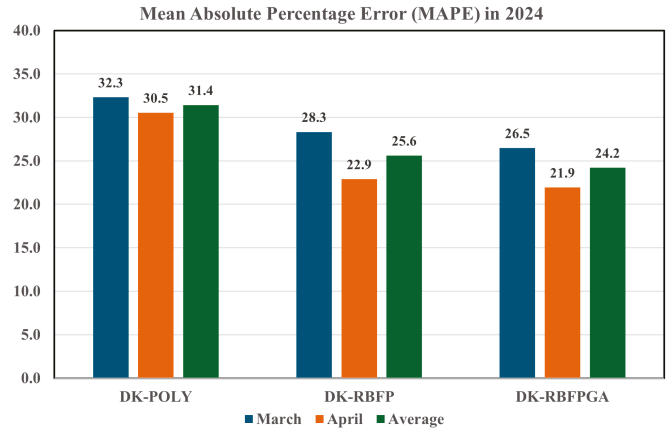
lower predictive accuracy compared to the other models. Conversely, both DK–RBFP and DK–RBFPGA demonstrated superior performance, with DK–RBFPGA showing a marginal advantage. A consistent trend of improvement was observed in 2024, as the MAPE values decreased for all three models. Throughout this period, DK–RBFPGA consistently yielded the lowest MAPE, indicating its enhanced reliability in prediction.

A comparison of the MSE for the same techniques and time periods is shown in Figure 3. DK–POLY in 2023 showed a significantly larger MSE, which was consistent with the MAPE results and confirmed its lower accuracy. DK–RBFPGA, on the other hand, had the lowest MSE, suggesting the most accurate predictions. The DK–POLY approach continued to exhibit the largest error, whereas DK–RBFPGA retained the lowest error, further demonstrating its superior prediction accuracy in 2024, even though the MSE dropped for all methods.

The RMSE for the three DK approaches over the same time periods is also compared in Figure 4. DK–POLY had the greatest RMSE in 2023, while the DK–RBFP and DK–RBFPGA approaches demonstrated reduced errors, with DK–RBFPGA performing slightly better. These results are consistent with the MAPE and MSE studies. The trend of reduced error continued in 2024, with all methods exhibiting lower RMSE values. However, DK–POLY consistently presented the highest RMSE, while DK–RBFPGA again achieved the lowest RMSE, highlighting its improved accuracy.



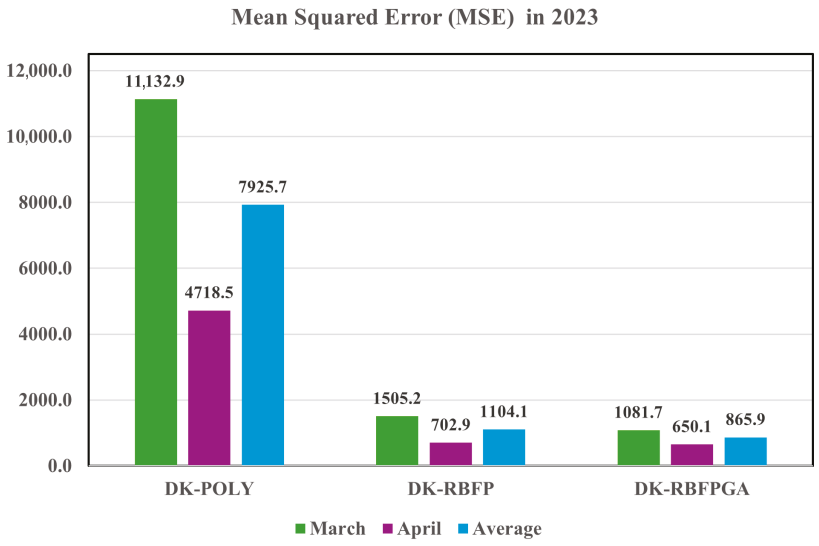
(**a**)



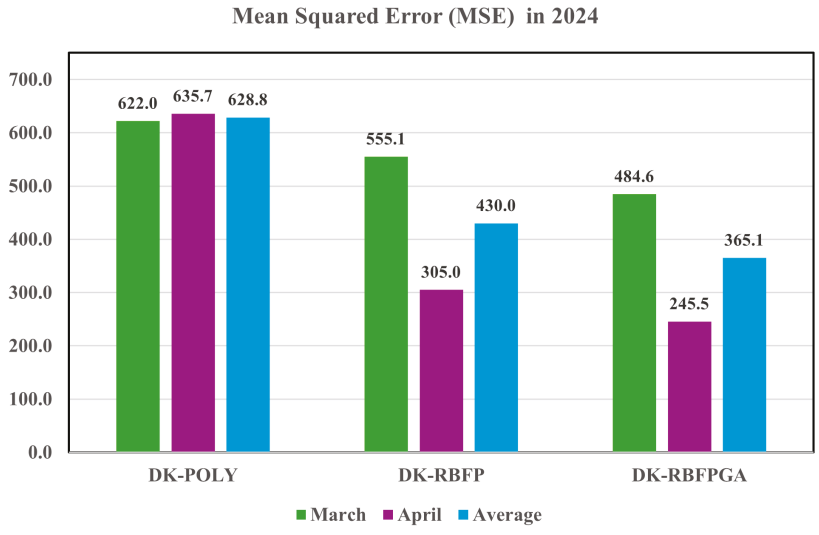
(**b**)

Figure 2. Comparison of the mean absolute percentage error in dual kriging predictions utilizing three different trend functions for the months of March and April in (**a**) the year 2023 and (**b**) the year 2024.

In summary, Figures 2–4 consistently demonstrate that DK–POLY produced the least accurate predictions, with significantly higher MAPE, MSE, and RMSE values, particularly in 2023. In contrast, DK–RBFPGA achieved the highest prediction accuracy, consistently yielding the lowest error metrics for both 2023 and 2024. Furthermore, over the two-year period, the average MAPE, MSE, and RMSE values for DK–RBFPGA improved by 9.0%, 19.8%, and 8.7%, respectively, compared to DK–RBFP. Nevertheless, the MAPE values for both DK–RBFP and DK–RBFPGA, ranging from 18.5% to 30.7%, indicate relatively high prediction errors. This may be attributed to the limited number of data points, which restricts the development of reliable trend and variogram models.



(**a**)



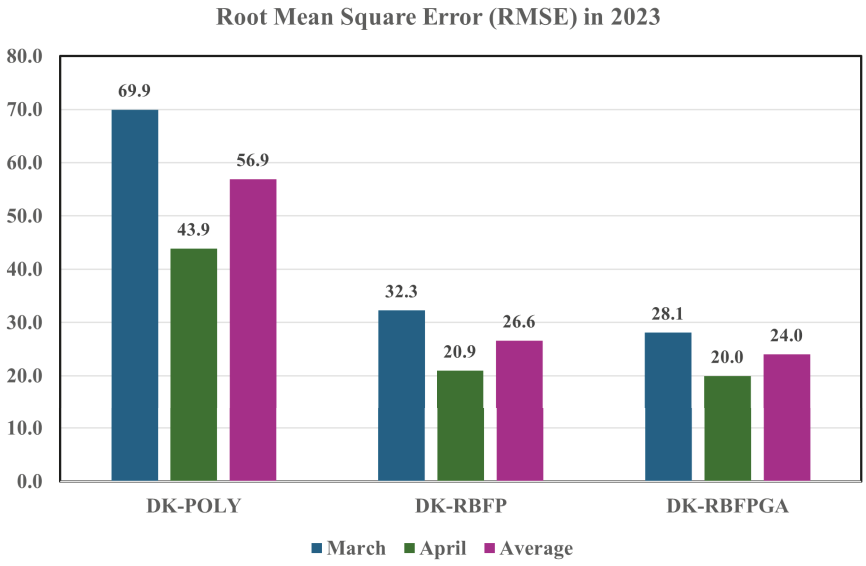
(**b**)

Figure 3. Comparison of the mean squared error in dual kriging predictions using three different trend functions for the months of March and April in (**a**) the year 2023 and (**b**) the year 2024.

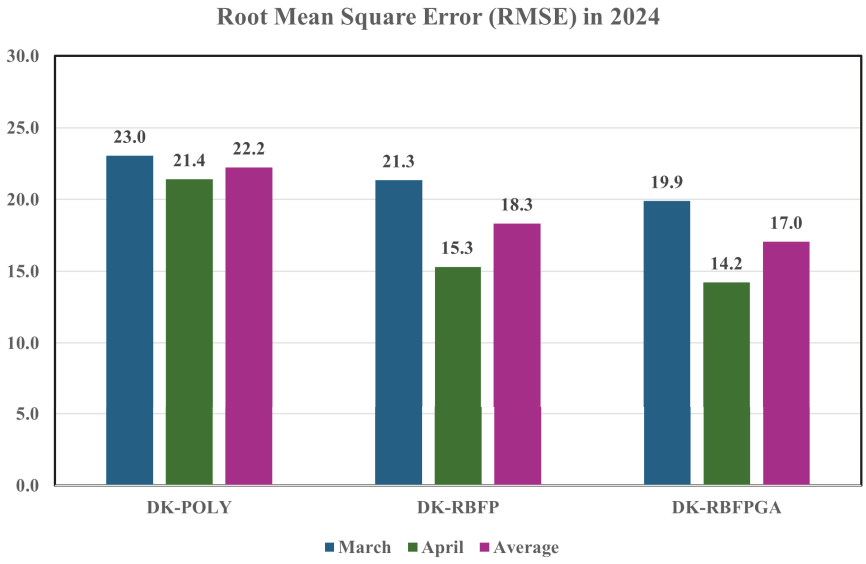
Based on the results indicating that the DK–RBFPGA method provides the highest prediction accuracy, its estimates were subsequently used to generate maps illustrating the spatial distribution of $PM_{2.5}$ for Weeks 1 to 4 of March and April in both 2023 and 2024. The data used for this mapping were taken from fold 2. The maps were produced

using QGIS (Quantum Geographic Information System) software, version 3.40.4, with the study area divided into a grid of square cells, each measuring 0.01 degrees per side. The resulting spatial distribution patterns of weekly mean $PM_{2.5}$ concentrations are presented in Figures 5–8.

Figure 5 illustrates the spatial distribution of weekly mean $PM_{2.5}$ concentrations across northern Thailand for March 2023. During the first two weeks (Figure 5a,b), moderate concentration levels were observed in most areas, with slightly elevated values in the central region during Week 1 and in both the central and northwestern regions during Week 2. In Week 3, there was a significant reduction in $PM_{2.5}$ concentrations (Figure 5c). However, concentrations increased again in Week 4 (Figure 5d), particularly in the northwestern part of the region.



(**a**)



(**b**)

Figure 4. Average root mean squared error of dual kriging predictions for three different trend functions in March and April: (**a**) 2023 and (**b**) 2024.

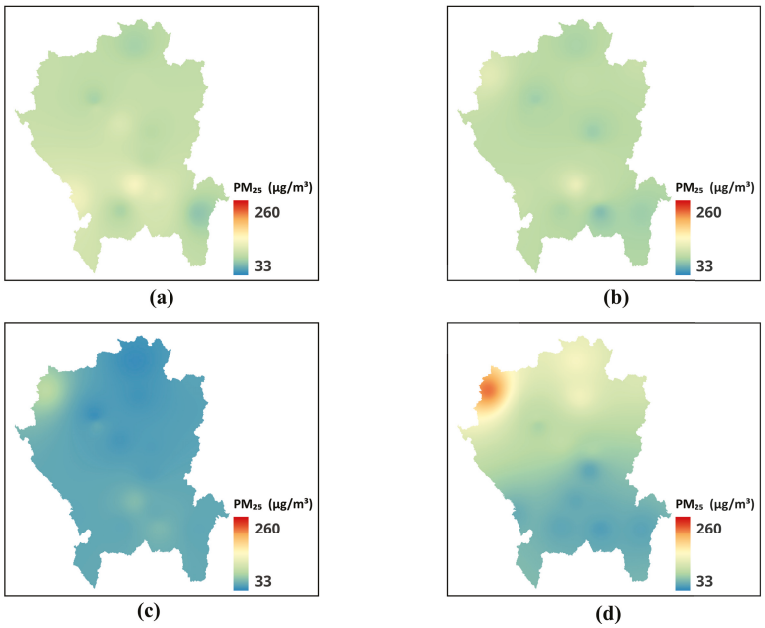


Figure 5. Spatial distribution maps of weekly mean $PM_{2.5}$ concentrations for March 2023: (**a**) Week 1, (**b**) Week 2, (**c**) Week 3, and (**d**) Week 4.

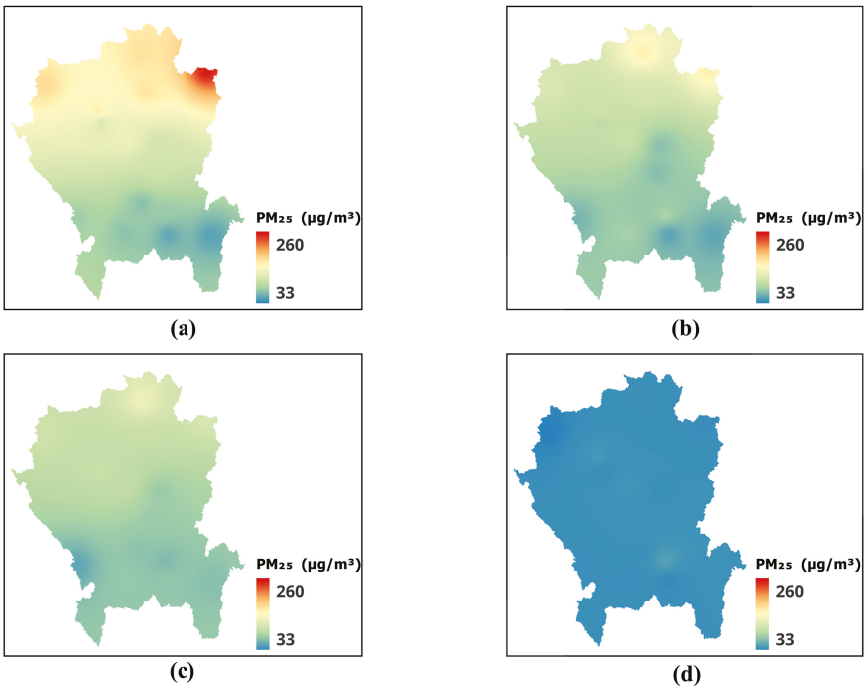


Figure 6. Spatial distribution maps of weekly mean $PM_{2.5}$ concentrations for April 2023: (**a**) Week 1, (**b**) Week 2, (**c**) Week 3, and (**d**) Week 4.

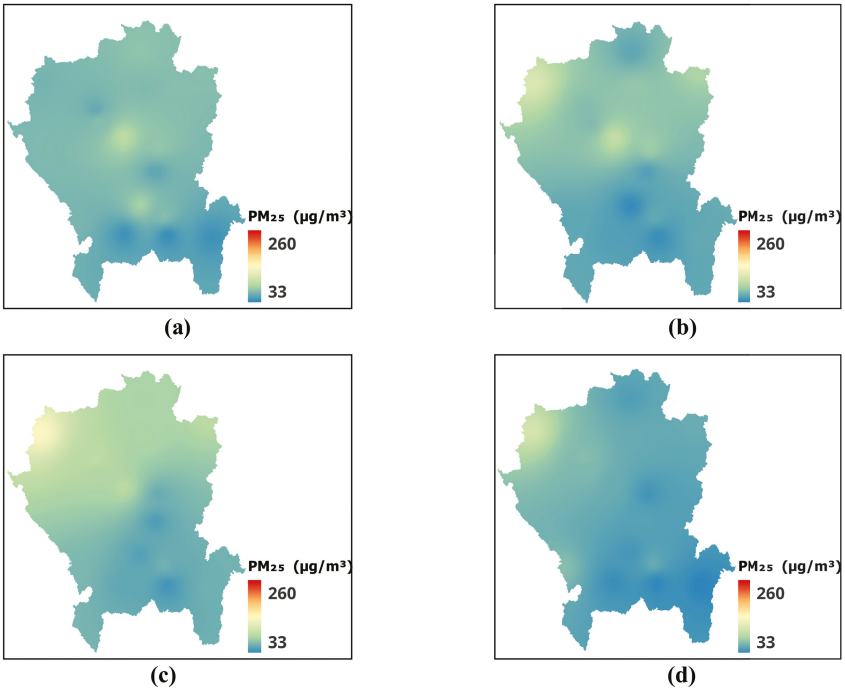


Figure 7. Spatial distribution maps of weekly mean $PM_{2.5}$ concentrations for March 2024: (**a**) Week 1, (**b**) Week 2, (**c**) Week 3, and (**d**) Week 4.

The spatial distribution of weekly mean $PM_{2.5}$ concentrations for April 2023 is depicted in Figure 6. High concentrations were observed in the northern and northeastern areas during Week 1 (Figure 6a). $PM_{2.5}$ levels declined in Week 2 (Figure 6b), and this downward trend continued through Weeks 3 and 4 (Figure 6c,d). By Week 4, concentrations had significantly decreased across the region.

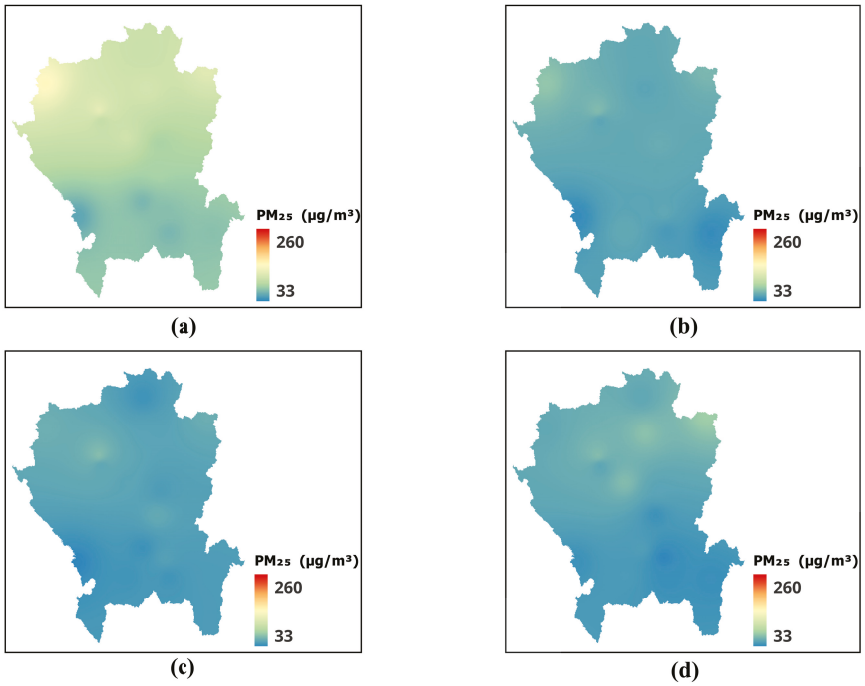


Figure 8. Spatial distribution maps of weekly mean $PM_{2.5}$ concentrations for April 2024: (**a**) Week 1, (**b**) Week 2, (**c**) Week 3, and (**d**) Week 4.

Figure 7 presents the spatial distribution of weekly mean $PM_{2.5}$ concentrations for March 2024. During Week 1 (Figure 7a), concentrations were generally low across the region, although some moderate hotspots appeared in the central part. Week 2 (Figure 7b) shows an increase in $PM_{2.5}$ levels, particularly in the central and northwestern regions. In Week 3 (Figure 7c), concentrations rose slightly compared to Week 2. However, by Week 4 (Figure 7d), levels had returned to lower values.

Figure 8 shows the spatial distribution of weekly mean $PM_{2.5}$ concentrations over northern Thailand for April 2024. In Week 1 (Figure 8a), concentrations were mostly moderate, with elevated values particularly in the northwestern region. A decrease in $PM_{2.5}$ concentrations was observed across most areas in Week 2 (Figure 7b), and this trend persisted through Week 3 (Figure 7c). However, in Week 4 (Figure 7d), concentrations increased slightly compared to the preceding week.

Based on Figures 5 and 6, $PM_{2.5}$ concentrations were significantly elevated during Week 4 of March 2023, as well as Weeks 1 and 2 of April 2023. These increases were likely influenced by forest fires and cross-border agricultural burning. This interpretation aligns with previous studies that identified March and April as peak periods for biomass burning and transboundary haze transport in northern Thailand [56,57]. In contrast, as shown in Figures 7 and 8, air quality in 2024 showed notable improvement compared to 2023. This improvement is primarily attributed to the systematic implementation of the national action plan on combating airborne dust. Key measures included reducing open burning, monitoring black smoke emissions, strengthening the enforcement of environmental laws, and enhancing regional cooperation to address transboundary haze. These coordinated efforts contributed to more tangible progress in controlling air pollution [58].

6. Conclusions and Discussion

This study improves the dual kriging (DK) method by introducing a novel nonlinear trend function that combines the Gaussian radial basis functions (GRBFs) with the first-order polynomial, resulting in two enhanced models: DK–RBFP and DK–RBFPGA. The proposed trend structure enhances the model's ability to capture complex spatial patterns. Theoretical support for the uniqueness of the DK solution is provided by the complete monotonicity of the covariance function and the positive definiteness of the coefficient matrix. To further increase flexibility and accuracy, the DK–RBFPGA model employs k-means clustering to identify GRBF centers and utilizes a genetic algorithm to optimize the bandwidth and other trend parameters. When tested on the spatial interpolation of $PM_{2.5}$ concentrations in northern Thailand, both enhanced models consistently outperformed the standard polynomial-based DK model, achieving significantly lower MAPE, MSE, and RMSE values. These results highlight the advantages of incorporating nonlinear trend structures and data-driven parameter optimization, which together provide a more accurate and flexible framework for environmental data analysis and spatial prediction.

Although the proposed trend model in the DK method has demonstrated strong performance in this application, its effectiveness relies on the appropriate selection of the number of clusters, which determines the number of GRBFs used in the trend function. This selection should be suited to the characteristics and quantity of the data. Moreover, accurate estimation of the GRBF parameters, including the centers and bandwidths, is crucial for achieving effective spatial interpolation.

As we have seen in this case study, DK–RBFPGA has greater predictive accuracy in comparison to DK–RBFP. However, employing the GA to estimate function parameters results in much higher computational expenses in comparison to the least squares method used in DK–RBFP, especially in the presence of large volumes of data. The least squares

method benefits from performing direct matrix calculations, which are efficient and require less memory when compared to the GA's iterative population-based searches with recurring fitness evaluations, where the algorithm assesses how well each solution solves the problem, making it much slower and more costly in terms of resources. As a result, the conflict between the two methods stems from the decision to be made in balancing higher predictive accuracy with computational efficiency.

Furthermore, the proposed models may be adapted for real-time prediction by using a local neighborhood approach that decreases the computational complexity for big datasets. In this method, the prediction for a given case is based on the closest observations, which drastically reduces the amount of data that needs to be processed. This approach helps the models remain responsive even with the large volumes and velocity requirements of data, which are typical in real-time systems.

In future work, we aim to explore alternative variogram models and nonlinear trend functions that guarantee the uniqueness of the DK solution and to extend our approach for broader applicability across diverse data types. In addition, further research will focus on the computational complexity and asymptotic properties of parameter estimation for the trend function.

Author Contributions: Conceptualization, S.U. and S.M.; methodology, S.U. and S.M.; software, S.M.; validation, S.M.; formal analysis, S.U., P.C. and S.M.; investigation, S.U. and S.M.; resources, S.M.; data curation, S.M.; writing—original draft preparation, S.U., P.C. and S.M.; writing—review and editing, S.U. and S.M.; visualization, S.M.; supervision, S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Fundamental Fund 2025, Chiang Mai University.

Data Availability Statement: The data utilized in this study were sourced from the Pollution Control Department (PCD) and the Thai Meteorological Department (TMD).

Acknowledgments: This research was partially supported by Chiang Mai University and the Fundamental Fund 2025, Chiang Mai University.

Conflicts of Interest: The authors declare that they have no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

OK	Ordinary kriging
KED	Kriging with an external drift
DK	Dual kriging
RBF	Radial basis function
GRBFs	Gaussian radial basis functions
GA	Genetic algorithm
DCC	Distance correlation coefficient
MAPE	Mean absolute percentage error
MSE	Mean squared error
RMSE	Root mean square error
OLS	Ordinary least squares
PCD	Pollution Control Department
TMD	Thai Meteorological Department
DK–POLY	Dual kriging with a second-order polynomial trend
DK–RBFP	Dual kriging with a trend function based on the GRBF and a first-order polynomial
DK–RBFPGA	DK–RBFP with parameters estimated using a genetic algorithm

References

1. Wackernagel, H. *Multivariate Geostatistics: An Introduction with Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2003.
2. Chaveesuk, R.; Smith, A.E. Dual Kriging: An exploratory use in economic metamodeling. *Eng. Econ.* **2005**, *50*, 247–271. [CrossRef]
3. Erdogan Erten, G.; Yavuz, M.; Deutsch, C.V. Combination of machine learning and kriging for spatial estimation of geological attributes. *Nat. Resour. Res.* **2022**, *31*, 191–213. [CrossRef]
4. Gravier, V.; Trochu, F.; Aubin, C.-É.; Plouznikoff, A.; Ozell, B. Interpolation by dual kriging of a moving flow front and conservation of the fluid mass. In *Fourth Canada-Japan Workshop on Composites*; CRC Press: Boca Raton, FL, USA, 2020; pp. 255–262.
5. He, Y.; Sun, J.; Song, P.; Wang, X. Dual Kriging assisted efficient global optimization of expensive problems with evaluation failures. *Aerosp. Sci. Technol.* **2020**, *105*, 106006. [CrossRef]
6. Trochu, F.; Vernet, N.; Sun, Y.; Echaabi, J.; Makradi, A.; Belouettar, S. Hybrid twin models of fiber compaction for composite manufacturing based on dual kriging. *Int. J. Mater. Form.* **2022**, *15*, 36. [CrossRef]
7. Kongsanun, C.; Chutsagulprom, N.; Moonchai, S. Spatio-temporal dual kriging with adaptive coefficient drift function. *Mathematics* **2024**, *12*, 400. [CrossRef]
8. Snepvangers, J.J.J.C.; Heuvelink, G.B.M.; Huisman, J.A. Soil water content interpolation using spatio-temporal kriging with external drift. *Geoderma* **2003**, *112*, 253–271. [CrossRef]
9. Freier, L.; Wiechert, W.; von Lieres, E. Kriging with trend functions nonlinear in their parameters: Theory and application in enzyme kinetics. *Eng. Life Sci.* **2017**, *17*, 916–922. [CrossRef]
10. Baisad, K.; Chutsagulprom, N.; Moonchai, S. A non-linear trend function for kriging with external drift using least squares support vector regression. *Mathematics* **2023**, *11*, 4799. [CrossRef]
11. Rusu, C.; Rusu, V. Radial basis functions versus geostatistics in spatial interpolations. In *IFIP International Conference on Artificial Intelligence in Theory and Practice*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 119–128.
12. Majdisova, Z.; Skala, V. Radial basis function approximations: Comparison and applications. *Appl. Math. Model.* **2017**, *51*, 728–743. [CrossRef]
13. Mai-Duy, N.; Tran-Cong, T. Approximation of function and its derivatives using radial basis function networks. *Appl. Math. Model.* **2003**, *27*, 197–220. [CrossRef]
14. Arora, G.; Bala, K.; Emadifar, H.; Khademi, M. A review of radial basis function with applications explored. *J. Egypt. Math. Soc.* **2023**, *31*, 1–14. [CrossRef]
15. Kawano, S.; Konishi, S. Nonlinear regression modeling via regularized Gaussian basis functions. *Bull. Inform. Cybernet.* **2007**, *3*, 83–96. [CrossRef]
16. Karimi, N.; Kazem, S.; Ahmadian, D.; Adibi, H.; Ballestra, L.V. On a generalized Gaussian radial basis function: Analysis and applications. *Eng. Anal. Bound. Elem.* **2020**, *112*, 46–57. [CrossRef]
17. Jasek, K.; Pasternak, M.; Miluski, W.; Bugaj, J.; Grabka, M. Application of Gaussian radial basis functions for fast spatial imaging of ground penetration radar data obtained on an irregular grid. *Electronics* **2021**, *10*, 2965. [CrossRef]
18. Shawe-Taylor, J.; Sun, S. Kernel methods and support vector machines. In *Academic Press Library in Signal Processing*; Elsevier: Amsterdam, The Netherlands, 2014; Volume 1, pp. 857–881.
19. Awad, M.; Khanna, R. Support vector machines for classification. In *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 39–66.
20. Orr, M.J.L. *Introduction to Radial Basis Function Networks*; Technical Report; Center for Cognitive Science, University of Edinburgh: Edinburgh, UK, 1996.
21. Rippa, S. An algorithm for selecting a good value for the parameter c in radial basis function interpolation. *Adv. Comput. Math.* **1999**, *11*, 193–210. [CrossRef]
22. Fasshauer, G.E.; Zhang, J.G. On choosing "optimal" shape parameters for RBF approximation. *Numer. Algorithms* **2007**, *45*, 345–368. [CrossRef]
23. Wang, J.G.; Liu, G.R. On the optimal shape parameters of radial basis functions used for 2-D meshless methods. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 2611–2630. [CrossRef]
24. Schaback, R. Error estimates and condition numbers for radial basis function interpolation. *Adv. Comput. Math.* **1995**, *3*, 251–264. [CrossRef]
25. Goldberg, D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley: Boston, MA, USA, 1989.
26. Holland, J.H. Genetic algorithms. *Sci. Am.* **1992**, *267*, 66–73. [CrossRef]
27. Shi, T.; Meng, S. Optimization of nonlinear functions based on improved genetic algorithm. In *Proceedings of the Third International Conference on Computer Science and Communication Technology (ICCSCT 2022)*; SPIE: Bellingham, WA, USA, 2022; Volume 12506, pp. 272–277.

28. Emmanuel, S.; Okoye, I.; Ezenweke, C.; Shobanke, D.; Adeniyi, I. Estimating nonlinear regression parameters using particle swarm optimization and genetic algorithm. *FUDMA J. Sci.* **2022**, *6*, 202–213. [CrossRef]
29. Onoghojobi, B.; Olewuezi, N.P.; Omojarabi, O. Nonlinear regression parameter estimates using genetic algorithms. *J. Niger. Assoc. Math. Phys.* **2023**, *65*, 173–178.
30. Sang, N.D. The genetic algorithm and its application in calculating the kinetic parameters of the thermoluminescence curve. In *Genetic Algorithms-Theory, Design and Programming*; IntechOpen: London, UK, 2024.
31. Montero, J.-M.; Fernández-Avilés, G.; Mateu, J. *Spatial and Spatio-Temporal Geostatistical Modeling and Kriging*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
32. Zimmerman, D.L.; Stein, M.; Gelfand, A.E.; Diggle, P.J. Classical geostatistical methods. In *Handbook of Spatial Statistics*; CRC Press: Boca Raton, FL, USA, 2010; pp. 29–44.
33. Myers, D.E. Smoothing and interpolation with radial basis functions. *WIT Trans. Model. Simul.* **2024**, *23*.
34. Cheney, W.; Light, W. *A Course in Approximation Theory*; Brooks/Cole Publishing Company: Pacific Grove, CA, USA, 1999.
35. Karim, M.A.; Jan, R.M. *Matrix Algebra*; Cambridge University Press: New York, NY, USA, 2005.
36. Gaspar, B.; Teixeira, A.P.; Soares, C.G. Assessment of the efficiency of kriging surrogate models for structural reliability analysis. *Probabilistic Eng. Mech.* **2014**, *37*, 24–34. [CrossRef]
37. Kleijnen, J.P.C. Regression and Kriging metamodels with their experimental designs in simulation: A review. *Eur. J. Oper. Res.* **2017**, *256*, 1–16. [CrossRef]
38. Lark, R.M.; Webster, R. Geostatistical mapping of geomorphic variables in the presence of trend. *Earth Surf. Process. Landf.* **2006**, *31*, 862–874. [CrossRef]
39. Yoshida, R. *Linear Algebra and Its Applications with R*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2021.
40. Barua, T.; Hiran, K.K.; Jain, R.K.; Doshi, R. *Machine Learning with Python*; Walter de Gruyter GmbH & Co., KG: Berlin, Germany, 2024.
41. Matheron, G. Principles of geostatistics. *Econ. Geol.* **1963**, *58*, 1246–1266. [CrossRef]
42. Junpen, A.; Pansuk, J.; Kamnoet, O.; Cheewaphongphan, P.; Garivait, S. Emission of air pollutants from rice residue open burning in Thailand, 2018. *Atmosphere* **2018**, *9*, 449. [CrossRef]
43. Jainontee, K.; Pongkiatkul, P.; Wang, Y.-L.; Weng, R.J.F.; Lu, Y.-T.; Wang, T.-S.; Chen, W.-K. Strategy design of $PM_{2.5}$ controlling for northern Thailand. *Aerosol Air Qual. Res.* **2023**, *23*, 220432. [CrossRef]
44. Mueller, W.; Loh, M.; Vardoulakis, S.; Johnston, H.J.; Steinle, S.; Precha, N.; Kliengchuay, W.; Tantrakarnapa, K.; Cherrie, J.W. Ambient particulate matter and biomass burning: An ecological time series study of respiratory and cardiovascular hospital visits in northern Thailand. *Environ. Health* **2020**, *19*, 77. [CrossRef]
45. Thongtip, S.; Srivichai, P.; Chaitiang, N.; Tantrakarnapa, K. The influence of air pollution on disease and related health problems in northern Thailand. *Sains Malays.* **2022**, *51*, 1993–2002. [CrossRef]
46. Goud, M.P.; Magadum, A. *Artificial Intelligence and Machine Learning*; RK Publication: Jalandhar, India, 2024.
47. Carter, N. *Data Science for Mathematicians*; CRC Press: Boca Raton, FL, USA, 2020.
48. Szekely, G.J.; Rizzo, M.L.; Bakirov, N.K. Measuring and testing independence by correlation of distances. *Ann. Stat.* **2007**, *35*, 2769–2794. [CrossRef]
49. Hou, J.; Ye, X.; Feng, W.; Zhang, Q.; Han, Y.; Liu, Y.; Li, Y.; Wei, Y. Distance correlation application to gene co-expression network analysis. *BMC Bioinform.* **2022**, *23*, 81. [CrossRef] [PubMed]
50. Ratnasingam, S.; Muñoz-Lopez, J. Distance correlation-based feature selection in random forest. *Entropy* **2023**, *25*, 1250. [CrossRef] [PubMed]
51. Rui, D.; Hui, J. High-accuracy transient fuel consumption model based on distance correlation analysis. *Fuel* **2022**, *321*, 123927. [CrossRef]
52. Park, S.; Moon, J.; Jung, S.; Rho, S.; Hwang, E. Explainable influenza forecasting scheme using DCC-based feature selection. *Data Knowl. Eng.* **2024**, *149*, 102256. [CrossRef]
53. Ren, Z.; Zhang, J.; Zhou, Y.; Ji, X. Prediction of $PM_{2.5}$ with a piecewise affine model considering spatial-temporal correlation. *J. Intell. Fuzzy Syst.* **2024**, *46*, 9525–9542. [CrossRef]
54. Yao, S.; Zhang, X.; Shao, X. Testing mutual independence in high dimension via distance covariance. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2018**, *80*, 455–480. [CrossRef]
55. Martínez-Gómez, E.; Richards, M.T.; Richards, D.S.P. Distance correlation methods for discovering associations in large astrophysical databases. *Astrophys. J.* **2014**, *781*, 39. [CrossRef]
56. Suriyawong, P.; Chuetor, S.; Samae, H.; Piriyakarnsakul, S.; Amin, M.; Furuuchi, M.; Hata, M.; Inerb, M.; Phairuang, W. Airborne particulate matter from biomass burning in Thailand: Recent issues, challenges, and options. *Heliyon* **2023**, *9*, e14261. [CrossRef]

57. Inlaung, K.; Chotamonsak, C.; Macatangay, R.; Surapipith, V. Assessment of transboundary $PM_{2.5}$ from biomass burning in northern Thailand using the WRF-chem model. *Toxics* **2024**, *12*, 462. [CrossRef]
58. Pollution Control Department. *Air and Noise Pollution Situation and Management in Thailand 2024*; Ministry of Natural Resources and Environment: Bangkok, Thailand, 2024. Available online: https://www.pcd.go.th/publication/36289/ (accessed on 5 August 2025). (In Thai)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

A Nonparametric Double Homogeneously Weighted Moving Average Signed-Rank Control Chart for Monitoring Location Parameter

Vasileios Alevizakos

Department of Mathematics, National Technical University of Athens, Zografou, 15773 Athens, Greece; basalebiz@yahoo.gr

Abstract: Nonparametric control charts are widely used in many manufacturing processes when there is a lack of knowledge about the distribution that the quality characteristic of interest follows. If there is evidence that the unknown distribution is symmetric, then the signed-rank statistic is preferred over other nonparametric statistics because it makes control charts more efficient. In this article, a nonparametric double homogeneously weighted moving average control chart based on the signed-rank statistic, namely, the DHWMA-SR chart, is introduced for monitoring the location parameter of an unknown, continuous and symmetric distribution. Monte Carlo simulations are used to study the run-length distribution of the proposed chart. A performance comparison study with the EWMA-SR, DEWMA-SR and HWMA-SR charts indicates that the DHWMA-SR chart is more effective under the zero-state scenario, while its steady-state performance is poor. Finally, two illustrative examples are given to demonstrate the application of the proposed chart.

Keywords: average run-length; exponentially weighted moving average; homogeneously weighted moving average; nonparametric control charts; signed-rank statistic

MSC: 62P30

1. Introduction

Statistical Process Control (SPC) is implemented in many manufacturing and non-manufacturing processes and aims to identify the assignable causes of unnatural variability. Control charts are the most important tool of SPC and are used for on-line process monitoring. There are two types of control charts: the memory-less charts, such as the Shewhart-type charts, introduced by Shewhart [1], and the memory-type charts, such as the cumulative sum (CUSUM) and the exponentially weighted moving average (EWMA) charts proposed by Page [2] and Roberts [3], respectively. The Shewhart-type control charts are easy to implement and effective for detecting large shifts in the process parameters but ineffective for small and moderate shifts. On the other hand, memory-type control charts are more complex to design but efficient for small and moderate shifts.

The design of control charts is usually based on the assumption that the quality characteristic of interest is a random variable that follows a known distribution, commonly, the normal distribution. However, in many cases, this assumption is violated, since little information about the underlying distribution is known. In such cases, the implementation

of parametric control charts may lead to unreliable results about the process. For this reason, nonparametric (or distribution-free) control charts have been developed by many authors. There are two types of nonparametric statistics that are used for monitoring shifts in the location parameter (median) when the target value is known a priori: the sign statistic and the signed-rank statistic. The sign statistic requires only the assumption of continuity for the underlying distribution, while the signed-rank statistic requires the additional assumption of symmetry. As a result, nonparametric control charts based on the signed-rank statistic are usually more efficient than those based on the sign statistic [4].

Several nonparametric control charts based on the signed-rank statistic have been developed by many authors. An overview of them up to 2018 is presented by Chakraborti and Graham [5]. However, many researchers continue to study and propose new schemes. Raza et al. [6] proposed the nonparametric double EWMA chart based on the signed-rank statistic (DEWMA-SR), which is a mixture of two EWMA-SR charts, and they found it more effective than the EWMA-SR and several nonparametric sign charts for small shifts. Alevizakos et al. [7] presented the double generally weighted moving average chart based on the signed-rank statistic (DGWMA-SR) as an effort to improve the performance of the GWMA-SR chart for small shifts. Raza et al. [8] developed nonparametric homogeneously weighted moving average schemes based on the sign (HWMA-SN) and signed-rank (HWMA-SR) statistics. Comparing to the nonparametric EWMA and CUSUM schemes, the nonparametric HWMA schemes were found to be more effective for small and (in some cases) moderate shifts. Perdikis et al. [9] presented the EWMA-SR chart for count data (CEWMA-SR) where the charting statistics are positive integers. The CEWMA-SR chart was found to be more effective than the CUSUM-SR, EWMA-SR and GWMA-SR charts for moderate to large shifts. Perdikis et al. [10] developed a modified nonparametric EWMA-SR chart, namely, the C-WRS EWMA chart, and presented a technique that guarantees steady run-length properties. Alevizakos et al. [11] introduced the nonparametric triple EWMA chart based on the signed-rank statistic (TEWMA-SR) to enhance the detection ability of the EWMA-SR chart. It was shown that the TEWMA-SR chart outperforms the CUSUM-SR, EWMA-SR and DEWMA-SR charts for small shifts. Tang et al. [12] presented a discrete state nonparametric adaptive EWMA chart based on the signed-rank statistic (ADSEWMA-SR). The above works are referred to nonparametric control charts based on the signed-rank statistic. Some recently published articles about nonparametric control charts based on the sign statistic are those of Castagliola et al. [13], Tang et al. [14] and Alevizakos et al. [15]. An overview of nonparametric control charts is provided by Qiu [16] (chapters 8 and 9), Chakraborti and Graham [4,5], Koutras and Triantafyllou [17] and Triantafyllou and Ram [18], while Qiu [19] gives some interesting perspectives on issues related to the nonparametric SPC.

Abid et al. [20] and Riaz et al. [21] proposed the double and the triple HWMA charts, respectively. However, as Alevizakos et al. [22] and Knoth et al. [23] mentioned, these schemes are reparameterizations of the HWMA chart. Similarly, the nonparametric double HWMA chart based on the sign statistic and the triple HWMA chart under ranked set sampling, proposed by Riaz et al. [24] and Zhang et al. [25], respectively, are reparameterizations of the HWMA schemes. Alevizakos et al. [22] developed a double HWMA (DHWMA) chart for normally distributed data to enhance the performance of the HWMA chart [26] for small and moderate shifts based on the DEWMA technique. It should be noted that Adeoti and Koleoso [27] proposed an hybrid HWMA (HHWMA) chart, but unfortunately, the mathematical expression of the variance of the charting statistic was incorrect. Consequently, Malela-Majika et al. [28] provided the correct version of the HHWMA chart. A literature review of the HWMA scheme and its modifications is presented by Letshedi et al. [29]. Alevizakos et al. [30] presented the nonparametric DHWMA sign chart where it was found

more effective than other nonparametric sign control charts, especially for small shifts in the location parameter. Motivated by [22,30], in this article, we propose a nonparametric DHWMA control chart based on the signed-rank statistic (regarded as DHWMA-SR chart) for monitoring shifts in the location parameter of a continuous and symmetric distribution.

The article is organized as follows. In Section 2, we present the structure of nonparametric control charts based on the signed-rank statistic, while Section 2 presents the proposed DHWMA-SR chart. The in-control (IC) and out-of-control (OOC) performances of the DHWMA-SR chart for the zero-state and steady-state scenarios are provided in Sections 4 and 5, respectively. A comparison study of the proposed chart against other nonparametric charts and the parametric DHWMA-$\overline{X}$ chart is given in Section 6. Two examples are discussed in Section 7 to demonstrate the implementation of the DHWMA-SR chart. Finally, conclusions and recommendations are provided in Section 8.

2. Design Structure of Nonparametric Signed-Rank Control Charts

2.1. The Signed-Rank Statistic

Let X be the quality characteristic of interest following an unknown, continuous and symmetric distribution around the median θ. Moreover, suppose that θ_0 denotes the known value of θ and the process is considered to be IC if $\theta = \theta_0$. It should be noted that this value is equal to the IC mean value μ_0 because the distribution is symmetric. Assume that a subgroup $\{X_{1t}, X_{2t}, \ldots, X_{nt}\}$ of size $n > 1$ is collected at at time $t = 1, 2, \ldots$ Let R_{it}^{+}, $i = 1, 2, \ldots, n$, denote the rank of the absolute value of the differences $|X_{it} - \theta_0|$, $i = 1, 2, \ldots, n$, within the t-th sample. The signed-rank (SR) statistic is defined as

$$SR_t = \sum_{i=1}^{n} \text{sign}(X_{it} - \theta_0) R_{it}^{+}, t = 1, 2, \ldots, \tag{1}$$

where $\text{sign}(x) = 1, 0, -1$ if $x > 0, x = 0$ or $x < 0$, respectively. The SR_t statistic is the difference between the sum of the ranks of the absolute differences corresponding to the positive and the negative differences, i.e., $SR_t = T_t^{+} - T_t^{-}$. Furthermore, the SR_t statistic can also be expressed as $SR_t = 2T_t^{+} - n(n+1)/2$. The IC values of mean and variance of the SR_t statistic are zero and $n(n+1)(2n+1)/6$, respectively. More information about the distribution of statistics T_t^{+} and SR_t can be found in [4,5,31].

2.2. The EWMA-SR Control Chart

The charting statistic of the EWMA-SR chart is defined by

$$E_t = \lambda SR_t + (1 - \lambda) E_{t-1},$$

where $E_0 = E(SR_t | IC) = 0$ and $0 < \lambda \leq 1$ is the smoothing parameter. The upper and lower time-varying control limits of the EWMA-SR chart are given by

$$UCL_t / LCL_t = \pm L_E \sqrt{\frac{n(n+1)(2n+1)}{6} \frac{\lambda}{2-\lambda} [1 - (1-\lambda)^{2t}]},$$

where $L_E > 0$ is the coefficient of the control limits, which has to be computed so that the attained IC average run-length (ARL_0) is close to a desired value.

The EWMA-SR chart reduces to the Shewhart-SR chart when $\lambda = 1$. A process is considered to be OOC if E_t plots beyond the control limits. Otherwise, the process is considered to be IC and no shift has been occurred.

2.3. The DEWMA-SR Control Chart

The DEWMA-SR chart is a mixture of two EWMA-SR charts and its charting statistic is defined via the system of equations

$$\begin{cases} E_t = \lambda_1 SR_t + (1-\lambda_1)E_{t-1}, \\ D_t = \lambda_2 E_t + (1-\lambda_2)D_{t-1}, \end{cases}$$

where $E_0 = D_0 = E(SR_t|IC) = 0$ and $0 < \lambda_1, \lambda_2 \leq 1$ are the smoothing parameters. For the case where $\lambda_1 = \lambda_2 = \lambda$, the IC expected value and variance of the statistic D_t are given by

$$E(D_t|IC) = 0$$

and

$$Var(D_t|IC) = \frac{\lambda^4[1+k-(t+1)^2k^t+(2t^2+2t-1)k^{t+1}-t^2k^{t+2}]}{[1-k]^3}\frac{n(n+1)(2n+1)}{6},$$

where $k = (1-\lambda)^2$. The time-varying control limits of the DEWMA-SR chart are given by

$$UCL_t/LCL_t = \pm L_D\sqrt{Var(D_t|IC)},$$

where $L_D > 0$ is the coefficient of the control limits and, similar, to the EWMA-SR chart, it has to be computed so that the attained ARL_0 is close to a desired value.

The DEWMA-SR chart reduces to the EWMA-SR chart when $\lambda_1 = 1$ or $\lambda_2 = 1$ and to the Shewhart-SR chart when $\lambda_1 = \lambda_2 = 1$. A process is considered to be OOC if $D_t \leq LCL_t$ or $D_t \geq UCL_t$. Otherwise, the process is said to be IC.

2.4. The HWMA-SR Control Chart

The charting statistic of the HWMA-SR chart gives a specific weight to the most current sample and the remaining is equally distributed among the past samples. More specifically, the charting statistic of the HWMA-SR chart is defined as

$$H_t = \lambda SR_t + (1-\lambda)\overline{SR}_{t-1},$$

where $\overline{SR}_0 = 0$, $\overline{SR}_{t-1} = \frac{\sum_{k=1}^{t-1} SR_k}{t-1}$ and $0 < \lambda \leq 1$ is the smoothing parameter.

The control limits of the HWMA-SR chart are given by

$$UCL_t/LCL_t = \begin{cases} \pm L_H\lambda\sqrt{\frac{n(n+1)(2n+1)}{6}} & \text{, for } t = 1, \\ \pm L_H\sqrt{\left(\lambda^2 + \frac{(1-\lambda)^2}{t-1}\right)\frac{n(n+1)(2n+1)}{6}} & \text{, for } t > 1, \end{cases}$$

where $L_H > 0$ is the coefficient of the control limits. The HWMA-SR chart reduces to the Shewhart-SR chart when $\lambda = 1$. The process is considered to be OOC if any H_t plots on or outside the control limits; otherwise, the process is considered to be IC.

3. The DHWMA-SR Control Chart

The DHWMA-SR chart is a mixture of two HWMA-SR charts and its charting statistic DH_t is defined via the systems of equations

$$\begin{cases} H_t = \lambda SR_t + (1-\lambda)\overline{SR}_{t-1}, \\ DH_t = \lambda H_t + (1-\lambda)\overline{H}_{t-1}, \end{cases} \tag{2}$$

where $0 < \lambda \leq 1$ is the smoothing parameter, $\overline{SR}_{t-1} = \frac{\sum_{k=1}^{t-1} SR_k}{t-1}$, $\overline{H}_{t-1} = \frac{\sum_{k=1}^{t-1} H_k}{t-1}$ and $\overline{SR}_0 = \overline{H}_0 = E(SR_t|IC) = 0$.

For $t = 1$, we have

$$DH_1 = \lambda^2 SR_1,$$

while for $t = 2$, it is

$$DH_2 = \lambda^2 SR_2 + 2\lambda(1-\lambda)SR_1.$$

For $t > 2$, the charting statistic DH_t simplifies to

$$DH_t = \lambda^2 SR_t + \frac{2\lambda(1-\lambda)}{t-1} SR_{t-1} + \frac{(1-\lambda)}{t-1} \sum_{u=1}^{t-2} \left(2\lambda + (1-\lambda) \sum_{k=u}^{t-2} \frac{1}{k} \right) SR_u.$$

The IC expected value of the charting statistic DH_t is equal to 0 and its IC variance is given by

$$Var(DH_t) = \begin{cases} \frac{\lambda^4 n(n+1)(2n+1)}{6} & \text{, for } t = 1, \\ \lambda^2\left(\lambda^2 + 4(1-\lambda)^2\right)\frac{n(n+1)(2n+1)}{6} & \text{, for } t = 2, \\ \left[\lambda^4 + \frac{4\lambda^2(1-\lambda)^2}{(t-1)^2} + \right. & \\ \left. \frac{(1-\lambda)^2}{(t-1)^2} \sum_{u=1}^{t-2} \left(2\lambda + (1-\lambda) \sum_{k=u}^{t-2} \frac{1}{k} \right)^2 \right] \frac{n(n+1)(2n+1)}{6} & \text{, for } t > 2. \end{cases} \tag{3}$$

For more details about the algebraic derivation for the mean and variance of DHWMA statistic, readers can consult [22]. The control limits of the DHWMA-SR chart are given by

$$UCL_t / LCL_t = \pm L_{DH} \sqrt{Var(DH_t)}, \tag{4}$$

where $L_{DH} > 0$ is the coefficient of the control limits which has to be calculated so that the attained ARL_0 is close to a desired ARL_0 and $Var(DH_t)$ is given by Equation (3). The DHWMA-SR chart is constructed by plotting the statistic DH_t versus t. If any of DH_t plots on or outside the control limits, a signal is given and the process is considered to be OOC. Otherwise, if $LCL_t < DH_t < UCL_t$, the process is said to be IC and the charting procedure continuous. Note that the proposed chart reduces to the Shewhart-SR chart when $\lambda = 1$.

4. Performance Analysis for the Zero-State Scenario

4.1. Design and Implementation

The run-length distribution and its associated characteristics are usually used in order to measure the performance of a control chart. The run-length is a discrete random variable and is defined as the number of charting statistics that must be plotted on a control chart

in order for the chart to give an OOC signal. The most popular metric of a control chart's performance is the ARL which is the expected number of charting statistics that must be plotted on a chart before the first OOC signal [32]. For an efficient control chart, the ARL_0 should be large to avoid any false alarm, while the OOC ARL (ARL_1) should be small to detect the shift quickly. Other characteristics of the run-length distribution, such as the standard deviation (SDRL) and the median (MDRL) of the run-length, are recommended to use to obtain more information about the run-length distribution.

The run-length characteristics of a control chart can be evaluated via Markov chains, integral equations and Monte Carlo simulations. In this study, we perform the latter method because the first two methods cannot be easily applied to the proposed scheme due to the complexity of the charting statistic DH_t and the form of control limits [22,30]. The simulation algorithm includes the following steps:

Step 1: For a specified value of $n > 1$ and a shift δ, generate 10,000 random variables from any continuous and symmetric distribution.

Step 2: For a desired value of ARL_0, specify the design parameters λ and L_{DH}.

Step 3: Compute the SR_t and DH_t statistics using Equations (1) and (2), respectively.

Step 4: Compute the control limits of the DHWMA-SR chart using Equation (4).

Step 5: Compare each charting statistic with the corresponding control limits. The DHWMA-SR chart gives a signal if any DH_t plots beyond the control limits.

Step 6: Repeat Steps 1 to 5 for 50,000 times and compute the run-length characteristics from the above values.

In order to evaluate the L_{DH} values of the proposed chart, so that the ARL_0 be equal to a desired one, we perform the above algorithm using $\delta = 0$. Table 1 presents the L_{DH} values of the DHWMA-SR chart for $n = 5, 10, 15, 20$ and $\lambda = 0.10, 0.14, 0.15, 0.17, 0.20, 0.25, 0.30$, so that the desired ARL_0 be equal to 200, 300, 370 and 500. These values were computed using the standard normal distribution; however, any other continuous and symmetric distribution can be used. From Table 1, we observe that for a specified value of n (λ), the value of L_{DH} increases as the value of λ (n) increases to obtain the desired value of ARL_0. For example, when $n = 5$ and $ARL_0 \approx 370$, the L_{DH} values are equal to 1.000, 1.280, 1.356, 1.492, 1.688, 1.915 and 2.027 for $\lambda = 0.10, 0.14, 0.15, 0.17, 0.20, 0.25$ and 0.30, respectively. Moreover, when $\lambda = 0.15$ and $ARL_0 \approx 500$, the L_{DH} values are equal to 1.448, 1.598, 1.631 and 1.650 for $n = 5, 10, 15$ and 20, respectively.

Table 1. Values of L_{DH} for the DHWMA-SR chart for different (n, λ) combinations to achieve a desired ARL_0.

		ARL_0						**ARL_0**			
n	***λ***	**200**	**300**	**370**	**500**	***n***	***λ***	**200**	**300**	**370**	**500**
5	0.10	0.897	0.955	1.000	1.067	10	0.10	0.931	1.023	1.071	1.153
	0.14	1.135	1.221	1.280	1.368		0.14	1.195	1.324	1.395	1.506
	0.15	1.204	1.293	1.356	1.448		0.15	1.269	1.403	1.479	1.598
	0.17	1.310	1.437	1.492	1.583		0.17	1.404	1.567	1.650	1.782
	0.20	1.491	1.622	1.688	1.766		0.20	1.617	1.795	1.890	2.033
	0.25	1.756	1.867	1.915	1.971		0.25	1.944	2.134	2.229	2.363
	0.30	1.939	2.007	2.027	2.052		0.30	2.199	2.369	2.451	2.555
15	0.10	0.941	1.035	1.085	1.173	20	0.10	0.948	1.043	1.094	1.178
	0.14	1.207	1.342	1.420	1.534		0.14	1.216	1.352	1.432	1.553
	0.15	1.279	1.422	1.504	1.631		0.15	1.285	1.435	1.521	1.650
	0.17	1.419	1.590	1.679	1.820		0.17	1.431	1.603	1.700	1.843
	0.20	1.641	1.831	1.936	2.090		0.20	2.652	1.850	1.958	2.118
	0.25	1.980	2.185	2.296	2.442		0.25	1.998	2.212	2.325	2.481
	0.30	2.243	2.438	2.531	2.654		0.30	2.269	2.470	2.570	2.699

4.2. IC Robustness

As the DHWMA-SR chart is nonparametric, its IC run-length characteristics should be approximately the same for any continuous and symmetric distribution. To study the IC robustness of the proposed chart, we use the following distributions:

- The standard normal distribution, $N(0,1)$;
- The scaled Student's t-distribution, $t(\nu)/\sqrt{\nu/(\nu-2)}$ with degrees of freedom $\nu = 4$ and 8;
- The logistic distribution, $LG(0,\sqrt{3}/\pi)$;
- The double exponential or Laplace distribution, $L(0,1/\sqrt{2})$;
- The contaminated normal (CN) distribution, which is a mixture of $N(0,\sigma_1^2)$ and $N(0,\sigma_2^2)$, represented by $(1-\alpha)N(0,\sigma_1^2)+\alpha N(0,\sigma_2^2)$ with $\sigma_1/\sigma_2 = 2$ and level of contamination $\alpha = 0.05$;
- The uniform distribution, $U(-\sqrt{3},\sqrt{3})$.

The scaled Student's t and the logistic distributions have heavier tails than the normal one, while the Laplace distribution has fatter tails than the normal. On the other hand, the CN distribution is used to study the effects of outliers. Note that all distributions have mean/median equal to 0 and standard deviation of 1.

Figures A1–A14 in the Appendix B show the IC run-length characteristics ARL_0, $MDRL_0$ and $SDRL_0$ of the DHWMA-SR chart under the above distributions for $n = 5$ and 10 respectively, given an $ARL_0 \approx 370$. From these figures, we observe that for a specified value of λ, the IC run-length characteristics remain approximately stable across all distributions. Comparing the Figures A1–A14 it is seen that for a specified value of λ, the $MDRL_0$ ($SDRL_0$) values for $n = 5$ are smaller (larger) than the corresponding $MDRL_0$ ($SDRL_0$) values for $n = 10$. For example, when $\lambda = 0.15$, the $MDRL_0$ and $SDRL_0$ values of the DHWMA-SR chart for $n = 5$ are 90 and 537, while the corresponding values for $n = 10$ are 183 and 427. It is also important to mention that for a specified value of n, the $MDRL_0$ ($SDRL_0$) value increases up (decreases down) to the value of $\lambda = 0.25$ and subsequently, it decreases (increases). Moreover, for $n = 5$, the $SDRL_0$ value ranges from 859 (when $\lambda = 0.10$) to 430 (when $\lambda = 0.25$), while for $n = 10$, the $SDRL_0$ value ranges from 698 (when $\lambda = 0.10$) to 284 (when $\lambda = 0.25$). From these findings, we observe that the IC run-length distribution is more positively-skewed for small values of n and λ. According to Chan and Zhang [33], a large value of $SDRL_0$ may lead to a high probability of an OOC signal. For this reason, we recommend practitioners to use values λ = 0.20–0.25 for the proposed scheme when $n = 5$ and $\lambda \geq 0.15$ when $n \in [10, 20]$.

4.3. OOC Performance

To study the OOC performance of the DHWMA-SR chart, we use the combinations of design parameters (λ, L_{DH}) shown in Table 1. We also consider the positive shifts $\delta = 0.05, 0.10, 0.25, 0.50, 0.75, 1.00, 1.25, 1.50$ in terms of population standard deviation. The results are similar for negative shifts as the underlying distribution is symmetric around the mean/median. Tables 2 and 3 show the zero-state run-length characteristics (ARL, MDRL and SDRL) of the DHWMA-SR chart for $n = 5$ and 10, respectively, given an $ARL_0 \approx 370$. From these tables, we observe the following:

Table 2. Zero-state run-length characteristics for the DHWMA-SR chart under various distributions when $ARL_0 \approx 370$ and $n = 5$.

				δ								
λ	L_{DH}	**Distribution**	**Characteristic**	**0**	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
0.10	1.000	$N(0,1)$	ARL	370.56	91.48	35.62	9.13	3.37	1.95	1.36	1.12	1.03
			MDRL	9	8	7	4	1	1	1	1	1
			SDRL	850.96	179.61	60.58	12.16	3.53	1.78	1.02	0.55	0.27
		t_4	ARL	376.39	69.10	26.26	6.81	2.67	1.66	1.28	1.13	1.06
			MDRL	10	8	7	4	1	1	1	1	1
			SDRL	856.20	129.31	42.47	8.44	2.63	1.41	0.85	0.54	0.36
		t_8	ARL	371.82	82.46	32.05	8.28	3.14	1.85	1.34	1.13	1.05
			MDRL	9	8	7	4	1	1	1	1	1
			SDRL	852.12	159.24	53.73	10.81	3.21	1.65	0.95	0.56	0.33
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	371.14	80.66	31.58	8.14	3.08	1.84	1.34	1.13	1.05
			MDRL	9	8	7	4	1	1	1	1	1
			SDRL	848.03	155.58	52.77	10.55	3.17	1.64	0.96	0.56	0.33
		$L(0, \frac{1}{\sqrt{2}})$	ARL	374.27	60.82	23.18	6.40	2.69	1.73	1.33	1.15	1.07
			MDRL	10	8	6	4	1	1	1	1	1
			SDRL	858.95	110.50	36.85	7.71	2.61	1.47	0.93	0.59	0.39
		$CN(a = 0.05)$	ARL	371.33	88.43	34.20	8.76	3.25	1.88	1.32	1.10	1.03
			MDRL	9	9	7	4	1	1	1	1	1
			SDRL	852.31	170.91	58.14	11.57	3.36	1.70	0.94	0.50	0.25
		$U(-\sqrt{3}, \sqrt{3})$	ARL	371.14	97.62	39.40	10.44	3.85	2.21	1.48	1.14	1.01
			MDRL	9	9	7	4	1	1	1	1	1
			SDRL	848.03	193.14	68.72	14.51	4.31	2.16	1.23	0.62	0.19
0.14	1.280	$N(0,1)$	ARL	370.53	127.75	52.40	12.87	4.35	2.39	1.60	1.25	1.09
			MDRL	65	41	23	7	3	1	1	1	1
			SDRL	571.41	187.26	69.99	14.73	4.16	2.05	1.21	0.73	0.41
		t_4	ARL	372.31	98.82	38.19	9.21	3.31	1.96	1.45	1.23	1.12
			MDRL	65	36	18	6	2	1	1	1	1
			SDRL	572.09	140.50	49.38	10.00	3.03	1.63	1.04	0.72	0.52
		t_8	ARL	372.60	116.32	46.46	11.32	3.91	2.22	1.55	1.25	1.11
			MDRL	66	39	21	7	3	1	1	1	1
			SDRL	572.98	169.02	61.60	12.69	3.66	1.90	1.17	0.75	0.49
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	372.32	117.18	46.21	11.32	3.92	2.22	1.56	1.25	1.11
			MDRL	67	40	21	7	3	1	1	1	1
			SDRL	572.96	169.10	60.91	12.59	3.66	1.88	1.16	0.75	0.48
		$L(0, \frac{1}{\sqrt{2}})$	ARL	372.32	87.54	33.71	8.59	3.32	2.05	1.52	1.26	1.14
			MDRL	67	33	16	5	2	1	1	1	1
			SDRL	572.96	121.85	42.89	9.15	3.02	1.71	1.13	0.78	0.55
		$CN(a = 0.05)$	ARL	366.05	124.46	50.14	12.14	4.12	2.28	1.55	1.22	1.08
			MDRL	64	41	22	7	3	1	1	1	1
			SDRL	565.39	180.90	66.80	13.72	3.89	1.95	1.15	0.67	0.37
		$U(-\sqrt{3}, \sqrt{3})$	ARL	372.32	139.37	58.13	14.85	5.11	2.77	1.78	1.28	1.05
			MDRL	67	44	24	8	4	2	1	1	1
			SDRL	572.96	205.38	78.34	17.30	5.06	2.47	1.43	0.78	0.30
0.15	1.356	$N(0,1)$	ARL	369.85	134.33	55.66	13.64	4.56	2.49	1.65	1.27	1.10
			MDRL	88	51	27	8	3	1	1	1	1
			SDRL	535.34	187.12	71.12	15.18	4.29	2.13	1.27	0.77	0.43
		t_4	ARL	372.55	104.13	40.59	9.73	3.45	2.02	1.48	1.24	1.12
			MDRL	89	44	21	6	2	1	1	1	1
			SDRL	536.54	141.08	50.33	10.32	3.12	1.69	1.08	0.74	0.53
		t_8	ARL	371.81	122.31	49.36	11.97	4.09	2.30	1.59	1.26	1.12
			MDRL	90	48	24	7	3	1	1	1	1
			SDRL	536.39	169.13	62.79	13.10	3.78	1.96	1.22	0.78	0.50

Table 2. *Cont.*

λ	L_{DH}	Distribution	Characteristic	δ								
				0	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	372.09	123.46	49.16	11.97	4.10	2.30	1.59	1.26	1.11
			MDRL	90	49	25	7	3	1	1	1	1
			SDRL	537.41	169.68	62.09	12.99	3.78	1.95	1.21	0.78	0.49
		$L(0, \frac{1}{\sqrt{2}})$	ARL	372.09	92.54	35.83	9.05	3.46	2.11	1.55	1.28	1.14
			MDRL	90	41	19	6	2	1	1	1	1
			SDRL	537.41	123.01	43.83	9.45	3.11	1.77	1.17	0.81	0.56
		$CN(a = 0.05)$	ARL	367.09	130.51	53.21	12.86	4.32	2.37	1.59	1.23	1.08
			MDRL	87	50	26	8	3	1	1	1	1
			SDRL	532.25	181.00	67.80	14.13	4.02	2.02	1.20	0.70	0.39
		$U(-\sqrt{3}, \sqrt{3})$	ARL	372.09	146.55	61.67	15.75	5.39	2.92	1.86	1.31	1.06
			MDRL	90	55	29	9	4	2	1	1	1
			SDRL	537.41	205.34	79.47	23.00	5.22	2.56	1.50	0.83	0.31
0.17	1.492	$N(0,1)$	ARL	370.26	146.89	62.38	15.44	5.10	2.79	1.87	1.42	1.18
			MDRL	146	76	37	10	4	2	1	1	1
			SDRL	471.94	184.05	71.93	15.83	4.50	2.21	1.36	0.89	0.57
		t_4	ARL	372.64	114.50	45.69	10.98	3.83	2.24	1.63	1.34	1.18
			MDRL	148	63	28	8	3	1	1	1	1
			SDRL	473.84	139.61	51.29	10.84	3.26	1.76	1.17	0.84	0.62
		t_8	ARL	373.53	134.57	55.39	13.53	4.57	2.57	1.78	1.39	1.20
			MDRL	150	71	33	9	4	2	1	1	1
			SDRL	474.77	166.77	63.69	13.69	3.96	2.04	1.31	0.89	0.62
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	372.67	135.14	55.40	13.50	4.58	2.57	1.78	1.40	1.20
			MDRL	148	72	34	9	4	2	1	1	1
			SDRL	475.41	166.43	63.20	13.57	3.95	2.03	1.31	0.89	0.61
		$L(0, \frac{1}{\sqrt{2}})$	ARL	372.67	102.46	40.36	10.15	3.82	2.33	1.71	1.39	1.21
			MDRL	148	58	26	7	3	1	1	1	1
			SDRL	475.41	122.54	44.82	9.89	3.25	1.85	1.26	0.91	0.66
		$CN(a = 0.05)$	ARL	367.92	143.13	59.55	14.56	4.82	2.66	1.80	1.37	1.16
			MDRL	144	75	35	10	4	2	1	1	1
			SDRL	470.68	178.06	68.68	14.76	4.20	2.10	1.30	0.84	0.53
		$U(-\sqrt{3}, \sqrt{3})$	ARL	372.67	159.37	69.20	17.88	6.08	3.31	2.13	1.49	1.12
			MDRL	148	81	41	12	5	3	1	1	1
			SDRL	475.41	199.96	80.27	18.55	5.46	2.65	1.58	0.97	0.46
0.20	1.688	$N(0,1)$	ARL	369.35	154.45	67.33	17.12	5.60	3.01	1.97	1.46	1.20
			MDRL	192	95	46	12	4	3	1	1	1
			SDRL	431.92	177.01	71.30	16.39	4.78	2.38	1.48	0.97	0.61
		t_4	ARL	368.68	121.75	49.80	12.16	4.17	2.38	1.69	1.36	1.19
			MDRL	190	78	35	9	4	1	1	1	1
			SDRL	431.47	135.78	51.46	11.33	3.47	1.89	1.26	0.89	0.65
		t_8	ARL	371.03	142.12	60.06	14.99	5.00	2.76	1.86	1.43	1.21
			MDRL	193	89	41	11	4	2	1	1	1
			SDRL	433.32	160.94	63.39	14.20	4.21	2.19	1.41	0.96	0.65
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	370.26	142.47	60.19	14.98	5.00	2.75	1.86	1.43	1.21
			MDRL	192	89	42	11	4	2	1	1	1
			SDRL	433.44	160.74	63.01	14.10	4.20	2.18	1.41	0.96	0.65
		$L(0, \frac{1}{\sqrt{2}})$	ARL	370.26	109.03	44.07	11.23	4.15	2.46	1.77	1.42	1.23
			MDRL	192	71	31	8	4	1	1	1	1
			SDRL	433.44	119.58	44.98	10.35	3.47	1.99	1.36	0.96	0.69
		$CN(a = 0.05)$	ARL	367.29	150.32	64.60	16.17	5.28	2.86	1.89	1.41	1.17
			MDRL	188	93	44	12	4	2	1	1	1
			SDRL	430.40	171.00	68.38	15.35	4.45	2.56	1.41	0.90	0.56

Table 2. *Cont.*

λ	L_{DH}	Distribution	Characteristic	δ								
				0	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
		$U(-\sqrt{3},\sqrt{3})$	ARL	370.26	167.51	74.95	19.95	6.75	3.62	2.27	1.53	1.12
			MDRL	192	102	51	15	5	3	1	1	1
			SDRL	433.44	192.48	79.75	19.24	5.78	2.84	1.71	1.04	0.48
0.25	1.915	$N(0,1)$	ARL	370.45	155.41	70.30	18.92	6.36	3.49	2.35	1.77	1.43
			MDRL	207	106	53	15	5	3	2	1	1
			SDRL	429.72	162.85	66.38	16.00	4.78	2.37	1.51	1.08	0.78
		t_4	ARL	369.94	123.56	52.52	13.53	4.75	2.77	2.00	1.60	1.37
			MDRL	204	87	40	11	4	2	1	1	1
			SDRL	429.27	125.04	48.19	11.14	3.48	1.91	1.33	1.01	0.79
		t_8	ARL	370.68	143.49	63.25	16.65	5.68	3.19	2.21	1.71	1.42
			MDRL	206	99	48	13	5	3	2	1	1
			SDRL	430.15	148.82	59.33	13.94	4.22	2.19	1.46	1.07	0.80
		$LG(0,\frac{\sqrt{3}}{\pi})$	ARL	370.80	144.03	63.25	16.59	5.68	3.20	2.22	1.71	1.42
			MDRL	204	100	48	13	5	3	2	1	1
			SDRL	431.87	148.19	58.98	13.84	4.21	2.18	1.46	1.07	0.80
		$L(0,\frac{1}{\sqrt{2}})$	ARL	370.80	111.17	46.92	12.49	4.70	2.85	2.09	1.68	1.43
			MDRL	204	80	37	10	4	2	1	1	1
			SDRL	431.87	110.18	42.52	10.23	3.48	2.00	1.42	1.07	0.84
		$CN(a=0.05)$	ARL	368.62	151.42	67.56	17.86	6.02	3.33	2.26	1.71	1.39
			MDRL	203	104	51	14	5	3	2	1	1
			SDRL	428.99	158.06	63.71	15.01	4.46	2.25	1.45	1.02	0.73
		$U(-\sqrt{3},\sqrt{3})$	ARL	370.80	167.77	78.23	22.05	7.70	4.20	2.75	1.93	1.38
			MDRL	204	113	58	18	6	4	2	2	1
			SDRL	431.87	177.42	74.56	18.74	5.81	2.82	1.70	1.11	0.66
0.30	2.027	$N(0,1)$	ARL	368.72	141.80	66.17	18.88	6.65	3.84	2.82	2.37	2.15
			MDRL	170	96	50	15	6	3	2	2	2
			SDRL	549.86	148.36	59.39	14.70	4.39	2.03	1.14	0.69	0.42
		t_4	ARL	368.17	113.50	50.20	13.66	5.08	3.21	2.56	2.28	2.14
			MDRL	170	80	39	11	4	3	2	2	2
			SDRL	543.16	112.47	43.47	10.24	3.12	1.53	0.94	0.63	0.44
		t_8	ARL	371.08	131.24	59.93	16.68	5.99	3.59	2.72	2.34	2.16
			MDRL	171	90	46	14	5	3	2	2	2
			SDRL	548.95	134.54	53.25	12.79	3.85	1.83	1.07	0.68	0.45
		$LG(0,\frac{\sqrt{3}}{\pi})$	ARL	368.96	131.62	59.84	16.62	5.98	3.58	2.72	2.34	2.16
			MDRL	168	90	46	14	5	3	2	2	2
			SDRL	550.61	134.20	53.08	12.67	3.82	1.83	1.07	0.68	0.45
		$L(0,\frac{1}{\sqrt{2}})$	ARL	368.96	102.57	44.87	12.65	5.05	3.29	2.63	2.33	2.17
			MDRL	168	74	36	10	4	3	2	2	2
			SDRL	550.61	99.24	38.22	9.37	3.11	1.62	1.02	0.69	0.48
		$CN(a=0.05)$	ARL	364.59	138.32	63.72	17.86	6.32	3.69	2.74	2.32	2.13
			MDRL	169	94	48	15	5	3	2	2	2
			SDRL	537.41	143.54	57.04	13.77	4.10	1.91	1.07	0.65	0.39
		$U(-\sqrt{3},\sqrt{3})$	ARL	368.96	152.86	73.40	21.97	7.98	4.51	3.13	2.45	2.10
			MDRL	168	101	56	18	7	4	3	2	2
			SDRL	550.61	162.83	66.92	17.23	5.40	2.52	1.37	0.77	0.33

Table 3. Zero-state run-length characteristics for the DHWMA-SR chart under various distributions when $ARL_0 \approx 370$ and $n = 10$.

				δ								
λ	L_{DH}	**Distribution**	**Characteristic**	**0**	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
0.10	1.071	$N(0,1)$	ARL	371.65	66.86	23.90	5.86	2.16	1.30	1.06	1.01	1.00
			MDRL	17	13	8	4	1	1	1	1	1
			SDRL	696.43	112.83	35.33	6.64	1.97	0.89	0.37	0.13	0.04
		t_4	ARL	371.40	49.72	17.67	4.41	1.75	1.18	1.04	1.01	1.00
			MDRL	17	11	7	3	1	1	1	1	1
			SDRL	695.73	79.93	24.84	4.68	1.48	0.65	0.28	0.13	0.07
		t_8	ARL	366.02	59.79	21.38	5.27	2.01	1.25	1.05	1.01	1.00
			MDRL	16	12	8	3	1	1	1	1	1
			SDRL	690.48	99.41	31.17	5.82	1.80	0.81	0.34	0.13	0.05
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	366.63	60.15	21.44	5.31	2.01	1.25	1.05	1.01	1.00
			MDRL	17	12	8	4	1	1	1	1	1
			SDRL	693.01	99.38	31.17	5.85	1.78	0.79	0.33	0.13	0.05
		$L(0, \frac{1}{\sqrt{2}})$	ARL	366.63	45.56	16.11	4.26	1.77	1.20	1.05	1.01	1.00
			MDRL	17	11	7	3	1	1	1	1	1
			SDRL	693.01	72.66	22.37	4.47	1.50	0.70	0.33	0.15	0.07
		$CN(a = 0.05)$	ARL	374.55	64.28	22.91	5.61	2.06	1.26	1.05	1.01	1.00
			MDRL	17	12	8	4	1	1	1	1	1
			SDRL	698.97	107.41	33.52	6.29	1.86	0.82	0.33	0.11	0.03
		$U(-\sqrt{3}, \sqrt{3})$	ARL	369.00	69.28	25.22	6.38	2.41	1.42	1.09	1.01	1.00
			MDRL	17	13	8	4	1	1	1	1	1
			SDRL	693.71	116.62	37.43	7.47	2.29	1.09	0.48	0.14	0.01
0.14	1.395	$N(0,1)$	ARL	370.16	96.72	35.26	7.94	2.68	1.50	1.12	1.02	1.00
			MDRL	139	48	20	5	1	1	1	1	1
			SDRL	462.14	122.89	41.38	7.94	2.29	1.09	0.51	0.20	0.07
		t_4	ARL	369.52	73.00	25.68	5.84	2.11	1.31	1.08	1.02	1.01
			MDRL	139	39	15	4	1	1	1	1	1
			SDRL	462.17	90.24	29.35	5.55	1.73	0.83	0.40	0.20	0.10
		t_8	ARL	368.50	87.93	31.49	7.08	2.46	1.43	1.11	1.02	1.00
			MDRL	137	44	18	5	1	1	1	1	1
			SDRL	461.36	110.47	36.85	6.97	2.08	1.00	0.47	0.20	0.09
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	369.43	88.01	31.51	7.12	2.47	1.43	1.11	1.02	1.00
			MDRL	137	45	18	5	1	1	1	1	1
			SDRL	462.20	109.98	36.64	6.97	2.06	0.99	0.47	0.20	0.08
		$L(0, \frac{1}{\sqrt{2}})$	ARL	369.43	66.98	23.45	5.59	2.14	1.35	1.10	1.03	1.01
			MDRL	137	36	14	4	1	1	1	1	1
			SDRL	462.20	82.26	26.55	5.27	1.75	0.89	0.45	0.23	0.11
		$CN(a = 0.05)$	ARL	371.38	93.37	33.81	7.59	2.56	1.44	1.10	1.02	1.00
			MDRL	140	47	20	5	1	1	1	1	1
			SDRL	464.13	118.11	39.48	7.52	2.16	1.01	0.45	0.17	0.05
		$U(-\sqrt{3}, \sqrt{3})$	ARL	369.43	101.30	37.80	8.82	3.05	1.69	1.18	1.02	1.00
			MDRL	137	50	21	6	2	1	1	1	1
			SDRL	462.20	128.06	44.70	8.96	2.67	1.31	0.61	0.21	0.03
0.15	1.479	$N(0,1)$	ARL	370.63	103.71	38.14	8.53	2.85	1.58	1.15	1.03	1.00
			MDRL	183	59	24	6	2	1	1	1	1
			SDRL	427.32	123.66	42.36	8.20	2.36	1.15	0.55	0.22	0.08
		t_4	ARL	369.30	78.42	27.77	6.24	2.23	1.36	1.10	1.03	1.01
			MDRL	182	47	18	5	1	1	1	1	1
			SDRL	427.05	91.21	30.16	5.74	1.79	0.88	0.44	0.22	0.12
		t_8	ARL	369.79	94.36	34.03	7.59	2.61	1.49	1.13	1.03	1.01
			MDRL	183	54	21	5	1	1	1	1	1
			SDRL	425.88	111.40	37.71	7.21	2.15	1.05	0.51	0.23	0.10

Table 3. *Cont.*

λ	L_{DH}	Distribution	Characteristic	δ								
				0	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	370.47	94.37	34.05	7.62	2.61	1.50	1.13	1.03	1.01
			MDRL	183	55	22	6	1	1	1	1	1
			SDRL	427.02	110.77	37.45	7.20	2.13	1.04	0.51	0.23	0.09
		$L(0, \frac{1}{\sqrt{2}})$	ARL	370.47	72.05	25.32	5.96	2.26	1.40	1.12	1.04	1.01
			MDRL	183	43	16	5	1	1	1	1	1
			SDRL	427.02	83.16	27.22	5.44	1.80	0.94	0.49	0.26	0.13
		$CN(a = 0.05)$	ARL	370.63	100.05	36.54	8.12	2.71	1.51	1.12	1.02	1.00
			MDRL	184	57	23	6	2	1	1	1	1
			SDRL	427.00	118.69	40.40	7.78	2.22	1.07	0.49	0.19	0.06
		$U(-\sqrt{3}, \sqrt{3})$	ARL	370.47	108.54	40.87	9.49	3.25	1.78	1.21	1.03	1.00
			MDRL	183	62	25	7	2	1	1	1	1
			SDRL	427.02	128.51	45.66	9.27	2.75	1.36	0.66	0.23	0.04
0.17	1.650	$N(0,1)$	ARL	369.54	114.66	43.04	9.60	3.15	1.71	1.20	1.04	1.01
			MDRL	251	77	30	7	3	1	1	1	1
			SDRL	373.04	123.00	43.68	8.69	2.51	1.25	0.63	0.28	0.10
		t_4	ARL	368.93	87.50	31.40	7.00	2.45	1.44	1.13	1.04	1.01
			MDRL	249	60	22	5	1	1	1	1	1
			SDRL	372.71	91.98	31.25	6.10	1.91	0.97	0.51	0.26	0.14
		t_8	ARL	368.52	104.78	38.46	8.58	2.87	1.61	1.18	1.04	1.01
			MDRL	249	71	27	6	2	1	1	1	1
			SDRL	372.20	111.52	38.94	7.67	2.28	1.15	0.59	0.28	0.12
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	369.23	104.70	38.57	8.57	2.87	1.61	1.18	1.04	1.01
			MDRL	249	71	27	7	2	1	1	1	1
			SDRL	372.14	110.84	38.78	7.64	2.26	1.14	0.59	0.28	0.13
		$L(0, \frac{1}{\sqrt{2}})$	ARL	369.23	80.53	28.74	6.65	2.47	1.50	1.17	1.05	1.01
			MDRL	249	56	21	5	1	1	1	1	1
			SDRL	372.14	84.02	28.38	5.77	1.93	1.03	0.57	0.31	0.16
		$CN(a = 0.05)$	ARL	368.64	110.57	41.24	9.15	2.99	1.63	1.16	1.03	1.00
			MDRL	250	74	29	7	2	1	1	1	1
			SDRL	372.45	118.14	41.61	8.23	2.36	1.17	0.57	0.24	0.08
		$U(-\sqrt{3}, \sqrt{3})$	ARL	369.23	120.34	46.21	10.74	3.61	1.94	1.28	1.04	1.00
			MDRL	249	81	33	8	3	1	1	1	1
			SDRL	372.14	128.22	46.93	9.82	2.92	1.47	0.76	0.29	0.05
0.20	1.890	$N(0,1)$	ARL	370.28	127.97	49.69	11.26	3.69	2.01	1.35	1.10	1.02
			MDRL	305	98	39	9	3	1	1	1	1
			SDRL	317.69	119.61	44.14	9.15	2.62	1.36	0.76	0.38	0.17
		t_4	ARL	367.73	98.61	36.48	8.19	2.87	1.69	1.26	1.11	1.05
			MDRL	302	76	29	7	3	1	1	1	1
			SDRL	317.57	90.73	31.98	6.43	1.99	1.09	0.64	0.39	0.25
		t_8	ARL	368.20	117.56	44.58	10.03	3.36	1.88	1.32	1.11	1.03
			MDRL	304	90	35	8	3	1	1	1	1
			SDRL	316.61	109.37	39.60	8.08	2.37	1.26	0.72	0.40	0.21
		$LG(0, \frac{\sqrt{3}}{\pi})$	ARL	369.19	117.27	44.55	10.05	3.36	1.89	1.33	1.11	1.03
			MDRL	305	90	35	8	3	1	1	1	1
			SDRL	317.09	108.22	39.29	8.03	2.35	1.25	0.72	0.39	0.21
		$L(0, \frac{1}{\sqrt{2}})$	ARL	369.19	91.00	33.39	7.78	2.90	1.76	1.32	1.13	1.05
			MDRL	305	71	26	6	3	1	1	1	1
			SDRL	317.09	82.82	29.08	6.09	2.01	1.14	0.71	0.43	0.26
		$CN(a = 0.05)$	ARL	368.84	123.39	47.65	10.74	3.51	1.91	1.30	1.08	1.02
			MDRL	302	94	37	9	3	1	1	1	1
			SDRL	317.49	115.01	42.21	8.66	2.46	1.28	0.70	0.34	0.14

Table 3. *Cont.*

λ	L_{DH}	Distribution	Characteristic	δ								
				0	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
		$U(-\sqrt{3},\sqrt{3})$	ARL	369.19	133.80	53.34	12.65	4.24	2.29	1.47	1.11	1.01
			MDRL	305	102	42	10	4	2	1	1	1
			SDRL	317.09	124.28	47.58	10.33	3.05	1.56	0.89	0.40	0.10
0.25	2.229	$N(0,1)$	ARL	369.93	138.24	55.72	13.25	4.40	2.45	1.63	1.24	1.07
			MDRL	321	112	46	11	4	2	1	1	1
			SDRL	284.58	114.78	43.47	9.40	2.70	1.43	0.92	0.56	0.31
		t_4	ARL	367.49	107.89	41.34	9.66	3.42	2.04	1.48	1.23	1.12
			MDRL	318	88	35	8	3	2	1	1	1
			SDRL	284.09	87.80	31.74	6.66	2.06	1.21	0.82	0.58	0.41
		t_8	ARL	367.34	127.27	50.25	11.80	4.00	2.28	1.58	1.25	1.10
			MDRL	319	103	42	10	4	2	1	1	1
			SDRL	283.47	105.02	39.10	8.33	2.45	1.35	0.89	0.58	0.36
		$LG(0,\frac{\sqrt{3}}{\pi})$	ARL	368.50	127.15	50.21	11.83	4.01	2.29	1.58	1.25	1.10
			MDRL	320	103	42	10	4	2	1	1	1
			SDRL	284.04	104.56	38.76	8.29	2.43	1.35	0.89	0.58	0.36
		$L(0,\frac{1}{\sqrt{2}})$	ARL	368.50	99.62	37.85	9.18	3.45	2.12	1.55	1.27	1.13
			MDRL	320	82	32	8	3	2	1	1	1
			SDRL	284.04	80.45	28.89	6.30	2.09	1.26	0.88	0.62	0.43
		$CN(a=0.05)$	ARL	368.11	133.51	53.50	12.61	4.19	2.33	1.56	1.20	1.06
			MDRL	319	108	45	11	4	2	1	1	1
			SDRL	284.11	110.70	41.69	8.90	2.54	1.36	0.87	0.52	0.27
		$U(-\sqrt{3},\sqrt{3})$	ARL	368.50	144.72	59.94	14.90	5.07	2.80	1.81	1.26	1.03
			MDRL	320	116	50	13	5	3	1	1	1
			SDRL	284.04	119.68	46.90	10.61	3.14	1.61	1.03	0.58	0.20
0.30	2.451	$N(0,1)$	ARL	370.61	140.16	57.19	14.02	4.76	2.70	1.85	1.39	1.15
			MDRL	309	113	48	12	4	3	2	1	1
			SDRL	291.90	114.52	42.65	9.20	2.65	1.39	0.95	0.66	0.41
		t_4	ARL	368.00	109.39	42.56	10.29	3.71	2.25	1.64	1.33	1.18
			MDRL	308	89	36	9	3	2	1	1	1
			SDRL	290.76	86.65	30.87	6.56	2.01	1.21	0.88	0.65	0.49
		t_8	ARL	368.29	129.17	51.57	12.52	4.33	2.52	1.77	1.38	1.17
			MDRL	310	104	43	11	4	2	1	1	1
			SDRL	289.68	104.20	38.20	8.17	2.40	1.33	0.93	0.66	0.46
		$LG(0,\frac{\sqrt{3}}{\pi})$	ARL	368.43	128.73	51.46	12.53	4.34	2.53	1.78	1.38	1.17
			MDRL	311	104	43	11	4	3	1	1	1
			SDRL	289.31	103.59	37.91	8.11	2.39	1.32	0.93	0.66	0.45
		$L(0,\frac{1}{\sqrt{2}})$	ARL	368.43	101.05	39.00	9.76	3.73	2.32	1.72	1.39	1.20
			MDRL	311	83	33	9	4	2	1	1	1
			SDRL	289.31	79.36	28.11	6.20	2.06	1.26	0.93	0.69	0.51
		$CN(a=0.05)$	ARL	368.93	135.37	54.84	13.35	4.54	2.59	1.77	1.34	1.13
			MDRL	308	108	46	12	4	3	1	1	1
			SDRL	290.93	110.26	40.79	8.73	2.49	1.33	0.91	0.61	0.37
		$U(-\sqrt{3},\sqrt{3})$	ARL	368.43	147.00	61.58	15.78	5.48	3.10	2.05	1.42	1.07
			MDRL	311	118	51	14	5	3	2	1	1
			SDRL	289.31	119.83	46.17	10.43	3.08	1.55	1.03	0.67	0.28

1. The DHWMA-SR chart performs better as the value of n increases. For example, when $n = 5$ and $\lambda = 0.15$, the ARL_1 value of the proposed chart is equal to 40.59 at $\delta = 0.10$ under the t_4 distribution, while the corresponding value is equal to 27.77 for $n = 10$.
2. The detection ability of the proposed chart is better as the value of λ decreases. For example, when $n = 10$ and $\delta = 0.05$, the ARL_1 values of the DHWMA-SR chart under the $N(0,1)$ distribution are equal to 66.86, 96.72, 103.71, 114.66, 127.97, 138.24

and 140.16 for $\lambda = 0.10, 0.14, 0.15, 0.17, 0.20, 0.25$ and 0.30, respectively. However, we remind that this superiority of the DHWMA-SR chart is accompanied by a larger value of $SDRL_0$.

3. For small to moderate shifts ($\delta \leq 0.50$), the proposed chart performs better when the underlying distribution is the $L(0, 1/\sqrt{2})$, while for larger shifts ($\delta > 0.50$), it performs better when the underlying distribution is the t_4. On the other hand, the worst performance is observed under the $U(-\sqrt{3}, \sqrt{3})$ for $\delta \leq 1.25$ and under the $L(0, 1/\sqrt{2})$ for $\delta = 1.50$.
4. Similar to the IC run-length distribution, the OOC run-length distribution is also positively skewed as $ARL_1 > MDRL_1$.

5. Performance Analysis for the Steady-State Scenario

The results reported in Tables 2 and 3 represent the zero-state performance of the DHWMA-SR chart, as it is assumed that the shift in the location parameter occurs in the beginning. In this subsection, we investigate the OOC performance of the proposed chart when the process is IC for a long time period and the shift occurs later; this case is referred to as steady-state. In other words, under the steady-state scenario, $\theta = \theta_0$ for $t < \tau$ and $\theta \neq \theta_0$ for $t \geq \tau$, where τ denotes the unknown change point.

Before we present the steady-state behavior of the DHWMA-SR chart, we discuss the steady-state performance of the HWMA scheme. Knoth et al. [23] pointed out that the HWMA scheme does not perform well in both the zero- and steady-state cases compared to the EWMA scheme, while Riaz et al. [34] showed that an HWMA scheme with suitable design parameters may be superior to an EWMA chart under the zero-state case, while it is more effective than the EWMA chart for large shifts under the steady-state case. However, both of these works do not provide a fair comparison study, as the ARL_0 values for the steady-state case are not computed. At this point, we note that the ARL_0 value of an EWMA chart with time-varying control limits converges quickly to a specific value as the value of τ increases, while this is not true for the HWMA chart. More specifically, the ARL_0 values of the HWMA chart change significantly over the time, especially for $\lambda \leq 0.10$. In Appendix A, Table A1 presents the IC run-length characteristics of the HWMA-SR chart for several values of λ, n and τ. It is clearly seen that for $n = 5$, the ARL_0 values increase as the value of τ also increases, while the corresponding $SDRL_0$ values are too large. On the other side, for $n = 10$, the ARL_0 values decrease significantly as the value of τ increases for $\lambda = 0.05$ and 0.10, while the steady-state ARL_0 values of the HWMA-SR chart with $\lambda = 0.25$ are very close to the corresponding zero-state one for all of the considered values of τ.

Table 4 shows the IC steady-state run-length characteristics of the DHWMA-SR chart given a zero-state $ARL_0 \approx 370$, when the shift occurs at $\tau = 100, 200, 500$ and 1000. Comparing to the results of Tables 2 and 3 for $\delta = 0$, we observe that the IC steady-state run-length characteristics are not close to the corresponding zero-state values. More specifically, the run-length characteristics decrease as the value of τ increases, while for $\lambda \leq 0.20$ when $n = 5$ or $\lambda \leq 0.17$ when $n = 10$ and values of τ are small, the ARL_0 and $MDRL_0$ values are larger than the corresponding zero-state values.

The steady-state run-length characteristics of the DHWMA-SR chart under various distributions for $\tau = 100$ and 500 when $n = 5$ and 10 are presented in Tables 5 and 6, respectively. From these tables, it is seen that for $\tau = 100$ and $\lambda = 0.20$ and 0.25 (when $n = 5$) and $\lambda = 0.17$ (when $n = 10$), the ARL_0 values are close to the corresponding zero-state values. However, this is not true for $\tau = 500$ where the IC run-length characteristics are significantly smaller than the zero-state ones. Moreover, the OOC steady-state performance

is very poor over the entire range of shifts. For example, when $n = 5$ and $\lambda = 0.25$, the zero-state ARL values of the DHWMA-SR chart under the $N(0,1)$ distribution are 370.45, 155.41, 70.30, 18.92, 6.36, 3.49, 2.35, 1.77 and 1.43 for $\delta = 0, 0.05, 0.10, 0.25, 0.50, 0.75, 1.00, 1.25$ and 1.50, respectively, while the corresponding steady-state ARL values for $\tau = 500$ are 211.25, 129.77, 88.21, 56.22, 42.41, 34.75, 29.66, 26.70 and 25.05. Thus, although the steady-state ARL_0 value is very small, the ARL_1 values are too large.

Table 4. IC steady-state run-length characteristics of the DHWMA-SR chart for different valious of τ.

		$n = 5$				$n = 10$			
λ	**Characteristic**	$\tau = 100$	**200**	**500**	**1000**	**100**	**200**	**500**	**1000**
0.10	ARL	696.63	718.10	702.15	575.00	608.95	612.22	535.18	370.55
	MDRL	138	136	43	9	204	198	77	16
	SDRL	1087.35	1095.80	1079.97	963.18	812.63	806.86	746.51	592.30
0.14	ARL	491.08	464.19	362.49	206.29	434.97	394.73	261.18	108.59
	MDRL	214	177	36	10	278	224	61	16
	SDRL	612.50	594.38	521.11	368.23	458.43	436.92	344.31	181.08
0.15	ARL	468.92	435.10	324.74	170.17	408.85	363.23	224.43	84.49
	MDRL	228	182	36	10	282	221	58	17
	SDRL	563.04	542.32	463.70	308.99	413.40	390.67	294.05	138.83
0.17	ARL	419.44	375.76	258.77	113.29	372.14	316.69	177.11	61.27
	MDRL	230	172	30	11	287	212	55	18
	SDRL	478.37	454.45	373.11	213.99	350.24	325.63	228.50	94.47
0.20	ARL	385.18	335.76	217.66	88.44	330.64	270.76	137.14	54.29
	MDRL	225	159	30	14	271	191	54	24
	SDRL	427.03	402.52	320.24	168.68	292.24	268.31	172.06	74.53
0.25	ARL	355.27	306.42	211.25	112.13	306.18	245.66	134.48	78.97
	MDRL	184	115	32	20	253	175	71	45
	SDRL	428.16	407.40	338.18	217.54	263.78	239.66	157.76	93.98
0.30	ARL	343.74	319.97	282.32	243.93	303.14	253.60	171.75	131.41
	MDRL	122	73	36	29	237	175	104	82
	SDRL	565.77	577.88	577.41	568.61	274.19	252.52	188.66	144.19

Table 5. Steady-state run-length characteristics for the DHWMA-SR chart under various distributions when $n = 5$.

					δ								
τ	λ	L_{DH}	**Distribution**	**Characteristic**	**0**	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
100	0.20	1.688	$N(0,1)$	ARL	385.18	206.60	122.62	59.92	35.90	27.59	23.67	21.59	20.44
				MDRL	225	159	108	58	35	27	23	21	20
				SDRL	427.03	205.09	106.66	45.21	25.25	19.00	16.22	14.81	14.05
			t_8	ARL	386.61	195.09	115.36	56.28	34.03	26.50	23.05	21.24	20.26
				MDRL	226	151	103	55	34	26	23	21	20
				SDRL	427.71	190.66	99.13	42.24	23.91	18.32	15.87	14.63	13.97
			$L(0, \frac{1}{\sqrt{2}})$	ARL	387.88	165.17	96.62	48.77	31.19	25.39	22.65	21.21	20.39
				MDRL	230	136	89	48	31	25	22	21	20
				SDRL	426.07	153.90	79.69	35.79	21.79	17.50	15.56	14.57	14.03
			$U(-\sqrt{3}, \sqrt{3})$	ARL	387.88	218.37	131.74	65.05	39.43	30.08	25.18	22.19	20.24
				MDRL	230	166	115	63	39	30	25	22	20
				SDRL	426.07	217.89	115.91	49.48	27.86	20.68	17.17	15.15	13.89
500	0.20	1.688	$N(0,1)$	ARL	217.66	145.33	99.39	60.85	45.49	39.09	35.01	32.30	30.65
				MDRL	30	30	31	32	33	31	28	26	24
				SDRL	320.24	197.17	121.58	64.68	44.05	36.89	33.11	30.77	29.38
			t_8	ARL	213.02	137.67	93.89	58.92	44.72	38.53	34.75	32.40	31.00
				MDRL	29	29	30	32	33	31	28	26	25
				SDRL	316.04	186.04	114.41	62.12	42.96	36.25	32.77	30.79	29.60
			$L(0, \frac{1}{\sqrt{2}})$	ARL	215.38	123.02	84.63	54.65	42.38	36.96	33.82	31.96	30.76
				MDRL	30	30	31	34	33	29	27	26	25
				SDRL	318.99	160.11	98.86	56.00	40.50	34.94	32.10	30.49	29.46
			$U(-\sqrt{3}, \sqrt{3})$	ARL	215.38	149.78	104.30	63.44	47.47	41.24	36.83	33.29	30.54
				MDRL	30	30	30	31	33	32	30	27	24
				SDRL	318.99	205.81	129.72	68.64	46.52	38.88	34.51	31.53	29.26

Table 5. *Cont.*

					δ								
τ	λ	L_{DH}	Distribution	Characteristic	0	0.05	0.10	0.25	0.50	0.75	1.00	1.25	1.50
100	0.25	1.915	$N(0,1)$	ARL	355.27	184.12	110.52	54.84	32.78	24.76	20.84	18.75	17.58
				MDRL	184	136	96	52	32	24	20	18	17
				SDRL	428.16	186.80	96.12	40.75	22.76	17.05	14.44	13.08	12.33
			t_8	ARL	352.16	173.58	103.95	51.62	30.98	23.69	20.25	18.46	17.49
				MDRL	180	129	90	49	30	23	19	18	17
				SDRL	428.65	172.91	89.18	38.15	21.56	16.41	14.12	12.95	12.32
			$L(0, \frac{1}{\sqrt{2}})$	ARL	354.27	147.44	87.60	44.57	28.19	22.47	19.81	18.37	17.56
				MDRL	183	118	79	43	27	22	19	18	17
				SDRL	428.13	138.96	71.88	32.21	19.59	15.60	13.80	12.84	12.32
			$U(-\sqrt{3}, \sqrt{3})$	ARL	354.27	193.19	118.61	59.62	36.18	27.25	22.38	19.35	17.40
				MDRL	183	140	102	56	35	26	22	19	17
				SDRL	428.13	198.09	104.55	44.71	25.15	18.61	15.34	13.40	12.19
500	0.25	1.915	$N(0,1)$	ARL	211.25	129.77	88.21	56.22	42.41	34.75	29.66	26.70	25.05
				MDRL	32	32	32	33	32	27	23	21	19
				SDRL	338.18	179.93	106.37	57.54	40.41	33.19	28.67	26.08	24.64
			t_8	ARL	206.49	122.34	83.88	54.93	41.52	34.04	29.33	26.74	25.34
				MDRL	31	31	32	33	31	26	23	21	19
				SDRL	333.48	168.13	100.46	55.81	39.44	32.48	28.36	26.09	24.83
			$L(0, \frac{1}{\sqrt{2}})$	ARL	209.15	109.24	76.40	51.46	38.69	32.14	28.42	26.34	25.16
				MDRL	32	32	33	34	30	25	22	20	19
				SDRL	337.25	142.41	87.08	51.12	36.97	31.02	27.69	25.80	24.73
			$U(-\sqrt{3}, \sqrt{3})$	ARL	209.15	134.36	92.53	58.24	44.32	37.26	31.79	27.67	24.88
				MDRL	32	32	32	32	32	29	25	21	19
				SDRL	337.25	189.30	113.83	60.59	42.32	35.04	30.32	26.82	24.43

Table 6. Steady-state run-length characteristics for the DHWMA-SR chart under various distributions when $n = 10$.

					δ								
τ	λ	L_{DH}	Distribution	Characteristic	0	0.05	0.10	0.25	0.50	0.75	1.00	1.25	1.50
100	0.17	1.650	$N(0,1)$	ARL	372.14	183.27	106.81	50.60	29.47	22.11	18.69	16.93	16.02
				MDRL	287	160	102	51	30	22	18	17	16
				SDRL	350.24	159.46	83.38	35.64	20.02	15.05	12.84	11.73	11.17
			t_8	ARL	374.03	174.79	101.28	48.17	28.25	21.51	18.47	16.93	16.13
				MDRL	289	155	97	48	28	21	18	17	16
				SDRL	351.43	149.79	77.83	33.55	19.09	14.60	12.64	11.67	11.18
			$L(0, \frac{1}{\sqrt{2}})$	ARL	373.27	149.84	86.33	42.18	25.88	20.52	18.12	16.87	16.19
				MDRL	286	136	84	42	26	20	18	17	16
				SDRL	352.01	124.61	64.91	29.15	17.56	14.02	12.47	11.67	11.24
			$U(-\sqrt{3}, \sqrt{3})$	ARL	373.27	189.05	111.53	54.18	32.01	23.89	19.77	17.35	15.92
				MDRL	286	165	106	54	32	24	20	17	16
				SDRL	352.01	164.62	87.25	38.01	21.57	16.07	13.41	11.91	11.06
500	0.17	1.650	$N(0,1)$	ARL	177.11	132.74	98.34	59.78	38.85	29.64	24.92	22.37	21.02
				MDRL	55	54	50	41	28	21	17	15	14
				SDRL	228.50	158.41	107.90	59.27	37.79	29.69	25.78	23.62	22.46
			t_8	ARL	175.05	130.34	95.74	57.50	37.23	28.69	24.47	22.27	21.09
				MDRL	55	54	51	40	27	20	17	15	14
				SDRL	225.72	153.88	103.65	56.54	36.23	28.84	25.31	23.44	22.43
			$L(0, \frac{1}{\sqrt{2}})$	ARL	177.17	120.06	86.13	51.33	33.65	27.01	23.75	21.99	21.01
				MDRL	55	53	48	36	24	19	16	15	14
				SDRL	227.99	138.17	91.17	50.22	33.28	27.54	24.76	23.24	22.37
			$U(-\sqrt{3}, \sqrt{3})$	ARL	177.17	135.97	101.99	63.48	42.31	32.51	26.65	22.97	20.72
				MDRL	55	54	52	44	32	25	19	16	14
				SDRL	227.99	161.90	111.54	62.48	40.28	31.67	26.90	23.93	22.07

Table 6. *Cont.*

τ	λ	L_{DH}	Distribution	Characteristic	δ								
					0	**0.05**	**0.10**	**0.25**	**0.50**	**0.75**	**1.00**	**1.25**	**1.50**
100	0.25	2.229	$N(0,1)$	ARL	306.18	178.71	108.49	50.10	26.77	18.75	15.12	13.29	12.35
				MDRL	253	159	102	49	26	18	15	13	12
				SDRL	263.78	141.28	76.42	31.94	17.10	12.44	10.30	9.29	8.75
			t_8	ARL	307.10	171.26	102.57	47.28	25.39	18.03	14.80	13.22	12.42
				MDRL	253	153	97	46	25	17	14	13	12
				SDRL	265.17	133.64	71.14	29.81	16.25	11.99	10.12	9.21	8.74
			$L(0, \frac{1}{\sqrt{2}})$	ARL	307.22	148.97	87.28	40.58	22.65	16.96	14.45	13.17	12.48
				MDRL	251	135	83	39	22	16	14	13	12
				SDRL	266.83	112.47	59.30	25.68	14.79	11.44	9.97	9.22	8.82
			$U(-\sqrt{3}, \sqrt{3})$	ARL	307.22	184.71	113.80	54.21	29.87	20.87	16.32	13.75	12.24
				MDRL	251	164	107	53	29	20	16	13	12
				SDRL	266.83	146.00	80.21	34.16	18.61	13.46	10.91	9.47	8.65
500	0.25	2.229	$N(0,1)$	ARL	134.48	117.05	94.31	56.14	29.64	19.18	14.62	12.47	11.41
				MDRL	71	68	61	41	19	10	6	4	3
				SDRL	157.76	126.70	94.04	52.29	30.01	21.25	17.23	15.27	14.28
			t_8	ARL	135.00	115.73	92.36	53.12	27.54	18.07	14.10	12.31	11.43
				MDRL	71	68	61	38	17	9	6	4	3
				SDRL	158.04	124.42	90.92	49.55	28.21	20.22	16.71	15.05	14.23
			$L(0, \frac{1}{\sqrt{2}})$	ARL	135.19	109.50	83.35	44.59	23.25	16.33	13.47	12.10	11.37
				MDRL	71	66	56	31	13	7	5	4	3
				SDRL	157.69	114.83	80.61	42.89	24.99	18.87	16.22	14.92	14.21
			$U(-\sqrt{3}, \sqrt{3})$	ARL	135.19	119.48	98.06	63.47	34.78	22.57	16.32	13.02	11.2
				MDRL	71	69	64	44	25	14	8	5	3
				SDRL	157.69	129.02	97.48	62.48	33.32	23.60	18.44	15.61	13.99

6. Comparison Study

In this section, we compare the performance of the proposed chart with the control charts presented in Section 2 for $n = 10$. It is to be noted that the different control charts are not compared under a fixed value of λ, but under an ARL_0 approximately equal to 370 and $SDRL_0 \leq ARL_0$. The last constraint was proposed by Chan and Zhang [33], who suggested to take into account in the design of a control chart not only the ARL_0, but also the $SDRL_0$ in order to avoid a large variation in the run-length. Moreover, the proposed DHWMA-SR chart is compared to the nonparametric DHWMA sign and the parametric DHWMA-$\overline{X}$ charts under several symmetric and asymmetric distributions. The control chart with the smallest ARL_1 value in a specific shift is considered to be the most effective.

6.1. DHWMA-SR Versus EWMA-SR, DEWMA-SR and HWMA-SR

Table 7 shows the ARL and SDRL values of the DHWMA-SR ($\lambda = 0.173, L_{DH} = 1.678$), HWMA-SR ($\lambda = 0.05, L_H = 2.308$), EWMA-SR ($\lambda = 0.19, L_E = 2.807$) and DEWMA-SR ($\lambda = 0.30, L_D = 2.681$) charts under the $N(0,1)$ distribution for both the zero- and steady-state ($\tau = 100$ and 500) cases. The design parameters of the competing charts were chosen so as to satisfy the above conditions. We note that the $SDRL_0$ values of the EWMA-SR and DEWMA-SR charts with time-varying control limits are large for very small values of λ and decrease as the value of λ increases. The values of 0.19 (for the EWMA-SR chart) and 0.30 (for the DEWMA-SR chart) are the smallest, so that $SDRL_0 \leq ARL_0$. From Table 7, we observe that for the zero-state scenario, the HWMA-SR chart is the most effective for a very small shift ($\delta = 0.05$), while the DHWMA-SR chart is superior to its competitors for the rest of the range of shifts. It should be pointed out that both the HWMA-SR and DHWMA-SR charts have significantly better detection ability than the EWMA-SR and DEWMA-SR charts for small shifts ($\delta \leq 0.25$). Moreover, the DEWMA-SR chart is more sensitive than the EWMA-SR chart over the entire range of shifts.

Table 7. Comparative study between DHWMA-SR, HWMA-SR, EWMA-SR and DEWMA-SR charts.

	Zero-State				**Steady-State ($\tau = 100$)**				**Steady-State ($\tau = 500$)**			
δ	**DHWMA-SR** $\lambda = 0.173$ $L_{DH} = 1.678$	**HWMA-SR** 0.05 $L_H = 2.308$	**EWMA-SR** 0.19 $L_D = 2.807$	**DEWMA-SR** 0.30 $L_E = 2.681$	**DHWMA-SR** 0.173 $L_{DH} = 1.678$	**HWMA-SR** 0.05 $L_H = 2.308$	**EWMA-SR** 0.19 $L_E = 2.807$	**DEWMA-SR** 0.30 $L_D = 2.681$	**DHWMA-SR** 0.173 $L_{DH} = 1.678$	**HWMA-SR** 0.05 $L_H = 2.308$	**EWMA-SR** 0.19 $L_E = 2.807$	**DEWMA-SR** 0.30 $L_D = 2.681$
0.00	370.50	370.14	370.67	370.66	369.80	345.11	367.54	366.50	172.74	202.37	370.52	371.43
	(369.34)	(322.32)	(368.89)	(370.37)	(344.82)	(303.07)	(367.90)	(365.20)	(222.37)	(221.42)	(371.27)	(371.65)
0.05	115.74	112.74	212.49	208.63	184.03	138.33	210.97	208.94	131.35	124.27	213.28	211.59
	(122.24)	(97.21)	(209.50)	(206.89)	(158.70)	(106.64)	(207.43)	(206.75)	(156.20)	(115.74)	(209.78)	(209.32)
0.10	43.91	44.22	88.07	85.47	107.47	70.55	88.66	86.37	97.99	78.93	88.54	86.66
	(44.06)	(35.24)	(84.10)	(82.13)	(83.14)	(47.06)	(84.36)	(82.37)	(107.06)	(64.44)	(84.23)	(83.00)
0.25	9.88	10.81	15.42	14.61	50.94	27.72	16.07	15.59	59.83	36.34	16.11	15.59
	(8.88)	(7.34)	(11.51)	(11.19)	(35.54)	(15.90)	(11.48)	(11.06)	(59.00)	(27.14)	(11.56)	(11.07)
0.50	3.21	4.09	4.76	4.47	29.58	13.85	5.53	5.73	38.70	17.48	5.53	5.73
	(2.54)	(2.06)	(2.39)	(2.36)	(19.94)	(7.60)	(2.48)	(2.22)	(37.49)	(13.75)	(2.48)	(2.21)
0.75	1.73	2.56	2.83	2.58	22.11	9.53	3.60	4.08	29.51	11.09	3.60	4.09
	(1.27)	(1.24)	(1.01)	(1.11)	(14.97)	(5.25)	(1.25)	(1.18)	(29.54)	(9.33)	(1.25)	(1.17)
1.00	1.21	1.83	2.22	1.88	18.67	7.62	2.91	3.50	24.69	8.30	2.91	3.50
	(0.64)	(0.98)	(0.48)	(0.71)	(12.76)	(4.25)	(0.88)	(0.91)	(25.53)	(7.40)	(0.87)	(0.91)
1.25	1.04	1.38	2.04	1.53	16.89	6.68	2.60	3.23	21.92	6.99	2.61	3.24
	(0.28)	(0.72)	(0.20)	(0.55)	(11.66)	(3.77)	(0.72)	(0.81)	(23.77)	(6.46)	(0.73)	(0.81)
1.50	1.01	1.14	2.00	1.31	15.97	6.19	2.46	3.11	20.57	6.31	2.46	3.11
	(0.11)	(0.45)	(0.06)	(0.47)	(11.09)	(3.52)	(0.67)	(0.76)	(22.13)	(5.99)	(0.67)	(0.76)

The results are different for the steady-state scenario. First of all, both of the EWMA-SR and DEWMA-SR charts have steady-state ARL_0 and $SDRL_0$ values approximately equal to the corresponding zero-state ones. This is not true for the HWMA-SR and DHWMA-SR charts when $\tau = 500$. We note that the DHWMA-SR chart has $ARL_0 \approx 370$ when $\tau = 100$, but its $SDRL_0$ value is smaller than the corresponding zero-state one (344.82 versus 369.34). Despite the fact that the DHWMA-SR and HWMA-SR charts have good OOC steady-state performance for $\delta = 0.05$ and $\delta \leq 0.10$, respectively, when $\tau = 100$, they perform very poorly for larger shifts. Similar results are observed when $\tau = 500$. Regarding the EWMA-SR and DEWMA-SR charts, it is seen that their steady-state performances are the same regardless of the value of τ. Furthermore, the DEWMA-SR chart is more effective than the EWMA-SR chart for $\delta \leq 0.25$ and vice versa for the rest of the range of shifts.

6.2. DHWMA-SR Versus DHWMA Sign and DHWMA-$\overline{X}$

Table 8 shows the ARL values of the parametric DHWMA-$\overline{X}$ and nonparametric DHWMA sign charts for $\lambda = 0.15$ and $n = 10$ under several symmetric distributions. From this table, we observe the following:

1. The nonparametric DHWMA charts are IC robust for all considered distributions, while the ARL_0 values of the DHWMA-$\overline{X}$ chart change when the underlying distribution also changes because it is designed for the normal distribution. For example, the ARL_0 of the DHWMA-$\overline{X}$ chart is 370.51 under the $N(0,1)$ distribution and ranges from 295 to 415 for the non-normal distributions.
2. The DHWMA-SR chart outperforms the DHWMA sign chart for small to moderate shifts ($\delta \leq 0.50$) and performs a slightly better for larger shifts under all considered distributions except for the Laplace distribution where, in this case, the DHWMA sign chart is more effective only for $\delta \leq 0.10$.
3. In the case of normal distribution, it is seen that the DHWMA-SR chart performs slightly better than the DHWMA-$\overline{X}$ chart for $\delta \leq 0.10$ and vice versa for the rest of the range of shifts.
4. The ARL_1 values of the DHWMA-$\overline{X}$ chart are approximately similar for all distributions. For example, when $\delta = 0.10$, the ARL_1 values of the DHWMA-$\overline{X}$ chart are 38.33, 38.64, 37.94, 38.27, 38.41, 38.70 and 38.55 under the normal, t_4, t_8, logistic, Laplace, CN and uniform distributions, respectively.
5. Although the ARL_0 values of the DHWMA-$\overline{X}$ chart are smaller than 370 under the Student's t, logistic and Laplace distributions, the ARL_1 values of the DHWMA-SR chart are smaller than those of the DHWMA-$\overline{X}$ chart, especially for $\delta \leq 0.50$, while their performance is similar for larger shifts.
6. In the case of CN distribution where the ARL_0 of the DHWMA-$\overline{X}$ chart is equal to 414.71, the DHWMA-SR chart is more effective than the DHWMA-$\overline{X}$ chart for $\delta \leq 0.25$, while the latter chart performs slightly better for larger shifts.
7. In the case of uniform distribution where the ARL_0 of the DHWMA-$\overline{X}$ chart is equal to 386.46, the ARL_1 values of the DHWMA-$\overline{X}$ chart are smaller than those of the DHWMA-SR chart over the entire range of shifts.

In the following lines, we present the performance of the above competing control charts under several asymmetric distributions. Those are as follows:

- The gamma distribution, $Gam(\alpha, \beta)$ with parameters $(\alpha, \beta) = (1,1), (3,1)$ and $(5,1)$;
- The lognormal distribution $LN(\mu, \sigma)$ with parameters $(\mu, \sigma) = (0, 0.25), (0, 0.5)$ and $(0,1)$;

- The Weibull distribution $W(\alpha, \beta)$ with parameters $(\alpha, \beta) = (0.5, 1), (1.5, 1)$ and $(5, 1)$.

All of these distributions, except for $W(5, 1)$, are positively skewed. Furthermore, the distributions have been shifted and scaled, so that they have a median and standard deviation equal to 0 and 1, respectively.

Table 8. ARL values of the DHWMA-SR, DHWMA sign and DHWMA-$\bar{X}$ charts for $\lambda = 0.15$ and $n = 10$ under various symmetric distributions when $ARL_0 \approx 370$.

δ	Chart	L_{DH}	Normal	t_4	t_8	Logistic	Laplace	CN	Uniform
0	DHWMA-SR	1.479	370.63	369.30	369.79	370.47	370.47	370.63	370.47
	DHWMA sign	1.504	372.69	368.86	371.16	370.89	370.89	369.90	370.89
	DHWMA-$\bar{X}$	1.551	370.51	295.79	347.46	353.25	335.31	414.71	386.46
0.05	DHWMA-SR	1.479	103.71	78.42	94.36	94.37	72.05	100.05	108.54
	DHWMA sign	1.504	212.55	93.78	116.86	115.81	66.19	129.55	190.18
	DHWMA-$\bar{X}$	1.551	104.61	103.08	104.99	105.28	104.71	107.83	105.94
0.10	DHWMA-SR	1.479	38.14	27.77	34.03	34.05	25.32	36.54	40.87
	DHWMA sign	1.504	52.12	33.66	44.10	43.43	23.78	50.21	84.02
	DHWMA-$\bar{X}$	1.551	38.33	38.64	37.94	38.27	38.41	38.70	38.55
0.25	DHWMA-SR	1.479	8.53	6.24	7.59	7.62	5.96	8.12	9.49
	DHWMA sign	1.504	11.90	7.55	9.83	9.82	5.98	11.46	20.50
	DHWMA-$\bar{X}$	1.551	8.38	8.27	8.19	8.29	8.30	8.25	8.31
0.50	DHWMA-SR	1.479	2.85	2.23	2.61	2.61	2.26	2.71	3.25
	DHWMA sign	1.504	3.93	2.70	3.40	3.31	2.47	3.74	6.32
	DHWMA-$\bar{X}$	1.551	2.70	2.65	2.71	2.66	2.67	2.65	2.69
0.75	DHWMA-SR	1.479	1.58	1.36	1.49	1.50	1.40	1.51	1.78
	DHWMA sign	1.504	2.17	1.59	1.92	1.90	1.58	2.05	3.20
	DHWMA-$\bar{X}$	1.551	1.46	1.42	1.46	1.45	1.45	1.43	1.46
1.00	DHWMA-SR	1.479	1.15	1.10	1.13	1.13	1.12	1.12	1.21
	DHWMA sign	1.504	1.46	1.20	1.35	1.34	1.23	1.39	1.98
	DHWMA-$\bar{X}$	1.551	1.09	1.09	1.09	1.09	1.09	1.08	1.09
1.25	DHWMA-SR	1.479	1.03	1.03	1.03	1.03	1.04	1.02	1.03
	DHWMA sign	1.504	1.16	1.06	1.11	1.11	1.08	1.12	1.33
	DHWMA-$\bar{X}$	1.551	1.01	1.02	1.01	1.01	1.01	1.01	1.01
1.50	DHWMA-SR	1.479	1.00	1.01	1.01	1.01	1.01	1.00	1.00
	DHWMA sign	1.504	1.04	1.02	1.03	1.03	1.02	1.03	1.03
	DHWMA-$\bar{X}$	1.551	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Table 9 shows the ARL values of the DHWMA-SR, DHWMA sign and DHWMA-$\bar{X}$ charts under the above asymmetric distributions. It can be seen that the proposed DHWMA-SR chart is not IC robust because the assumption of symmetry is violated. On the other hand, as it was expected, the DHWMA sign chart is IC robust because the assumption of symmetry is not required. Moreover, we conclude that the ARL values of the DHWMA-$\bar{X}$ chart are comparable to those under the normal distribution when the underlying distribution approaches the normal. Finally, we note that the DHWMA sign chart is superior to the DHWMA-$\bar{X}$ chart for the $Gam(1, 1), LN(0, 0.5), LN(0, 1)$ and $W(0.5, 1)$ distributions.

Table 9. ARL values of the DHWMA-SR, DHWMA sign and DHWMA-$\bar{X}$ charts for $\lambda = 0.15$ and $n = 10$ under various asymmetric distributions when $ARL_0 \approx 370$.

δ	Chart	L_{DH}	Gam(1,1)	Gam(3,1)	Gam(5,1)	LN(0,0.25)	LN(0,0.5)	LN(0,1)	W(0.5,1)	W(1.5,1)	W(5,1)
0	DHWMA-SR	1.479	19.64	48.91	71.93	97.52	39.12	16.77	8.66	43.94	217.71
	DHWMA sign	1.504	372.19	370.25	372.72	372.69	372.69	372.69	370.84	370.89	370.89
	DHWMA-$\bar{X}$	1.551	364.61	364.77	367.82	366.84	353.42	292.64	296.52	371.71	370.50
0.05	DHWMA-SR	1.479	10.82	22.33	29.11	35.18	17.98	7.23	2.88	21.10	222.38
	DHWMA sign	1.504	97.33	121.52	125.86	124.87	103.67	42.50	11.24	130.05	138.33
	DHWMA-$\bar{X}$	1.551	103.45	103.24	103.45	103.81	102.80	102.42	100.03	102.26	104.85
0.10	DHWMA-SR	1.479	6.88	12.84	15.74	17.99	10.39	3.99	1.05	12.43	61.27
	DHWMA sign	1.504	33.71	45.64	47.80	47.38	37.18	12.43	1.52	49.26	54.28
	DHWMA-$\bar{X}$	1.551	40.05	39.28	39.25	38.73	39.58	42.84	43.61	38.66	37.77
0.25	DHWMA-SR	1.479	2.68	4.50	5.20	5.59	3.68	1.24	1.00	4.53	10.68
	DHWMA sign	1.504	6.40	9.70	10.39	10.38	7.63	2.06	1.00	10.52	12.63
	DHWMA-$\bar{X}$	1.551	59.61	8.77	8.63	8.62	9.03	10.21	11.46	8.93	8.35
0.50	DHWMA-SR	1.479	1.08	1.73	1.98	2.08	1.42	1.00	1.00	1.76	3.29
	DHWMA sign	1.504	1.60	2.93	3.25	3.29	2.31	1.00	1.00	3.16	4.23
	DHWMA-$\bar{X}$	1.551	3.05	2.92	2.89	2.88	3.08	3.62	4.43	3.17	2.83
0.75	DHWMA-SR	1.479	1.00	1.09	1.17	1.22	1.02	1.00	1.00	1.09	1.77
	DHWMA sign	1.504	1.00	1.47	1.68	1.74	1.19	1.00	1.00	1.54	2.36
	DHWMA-$\bar{X}$	1.551	1.71	1.62	1.58	1.60	1.76	2.14	2.79	1.87	1.61
1.00	DHWMA-SR	1.479	1.00	1.00	1.01	1.02	1.00	1.00	1.00	1.00	1.23
	DHWMA sign	1.504	1.00	1.05	1.14	1.18	1.00	1.00	1.00	1.04	1.60
	DHWMA-$\bar{X}$	1.551	1.23	1.17	1.15	1.17	1.28	1.56	2.10	1.39	1.19
1.25	DHWMA-SR	1.479	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.05
	DHWMA sign	1.504	1.00	1.00	1.01	1.02	1.00	1.00	1.00	1.00	1.24
	DHWMA-$\bar{X}$	1.551	1.06	1.04	1.03	1.04	1.09	1.28	1.73	1.17	1.06
1.50	DHWMA-SR	1.479	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.01
	DHWMA sign	1.504	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.08
	DHWMA-$\bar{X}$	1.551	1.01	1.01	1.00	1.01	1.03	1.14	1.51	1.08	1.02

7. Illustrative Examples

7.1. Smartphone Accelerometer Dataset

In this example, we use the smartphone accelerometer dataset, provided by Riaz et al. [35] and also applied by Tang et al. [12]. The accelerometer is widely used in many manufacturing and non-manufacturing processes to measure the vibrations in a car or airplane engine or in an earthquake. The dataset consists of 15 samples, each of size $n = 10$, and is presented in Table 10. According to [12,35], the underlying distribution is symmetric and the IC process median is equal to -7.437. Thus, nonparametric control charts based on the signed-rank statistic can be applied. Setting an $ARL_0 = 370$ and $\text{SDRL}_0 \leq \text{ARL}_0$, we construct the EWMA-SR ($\lambda = 0.19, L_E = 2.807$), the DEWMA-SR ($\lambda = 0.30, L_D = 2.681$), the HWMA-SR ($\lambda = 0.05, L_H = 2.308$) and the DHWMA-SR ($\lambda = 0.173, L_{DH} = 1.678$) charts. The charting statistics are also given in Table 10, while the control charts are shown in Figures 1–4. The red dots indicate the charting statistics that lie over the UCL_t. We observe that the proposed chart detects the shift after three samples, the HWMA-SR chart after four samples and the EWMA-SR and DEWMA-SR charts after 13 samples. This example indicates the superiority of the proposed chart versus its competitors.

7.2. Piston Rings Dataset

In this example, we use the dataset from a piston ring process, provided by Montgomery [32]. At each sampling point t, 15 samples, each of size $n = 5$, are collected in order to detect a shift in the mean/median of the inside diameter (in mm) of piston rings. The dataset is shown in Table 11. The IC process median θ_0 computed from data known to be IC (see Table 6.3 of [19]) is equal to 74, while the Pearson's coefficient of skewness is equal to 0.052, which is very close to 0. Thus, the distribution of the dataset is symmetric. Setting an $ARL_0 = 200$, we construct the EWMA-SR ($\lambda = 0.05, L_E = 2.267$), DEWMA-SR ($\lambda = 0.05, L_D = 1.726$), the HWMA-SR ($\lambda = 0.05, L_H = 1.924$) and the DHWMA-SR ($\lambda = 0.20, L_{DH} = 1.491$). These values of design parameters give the desired value of ARL_0. The charting statistics are also provided in Table 11 and the control charts are displayed in Figures 5–8. From these figures, it is seen that the DEWMA-SR chart gives an OOC signal at the 13th sample and the other charts at the 12th sample. This example indicates that the proposed chart performs as well as or better than the other charts.

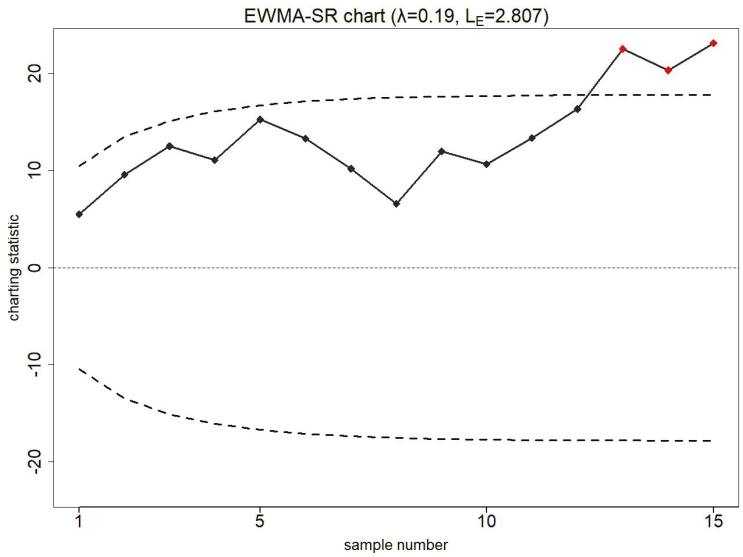


Figure 1. The EWMA-SR chart for piston rings dataset.

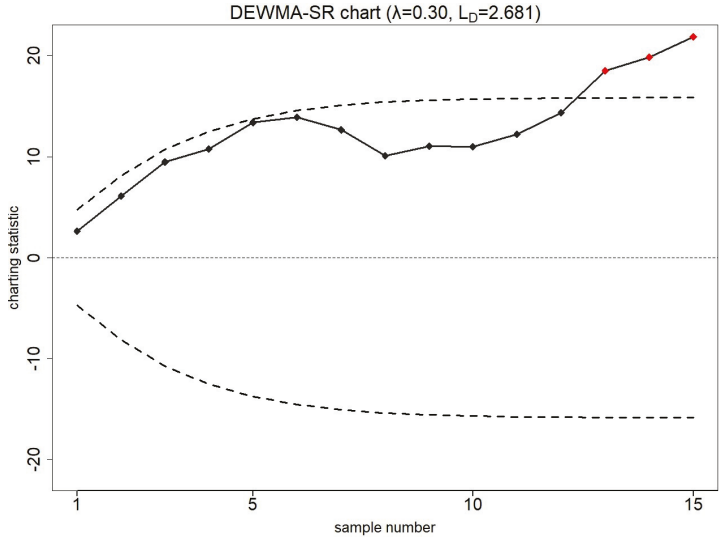


Figure 2. The DEWMA-SR chart for smart phone accelerometer rings dataset.

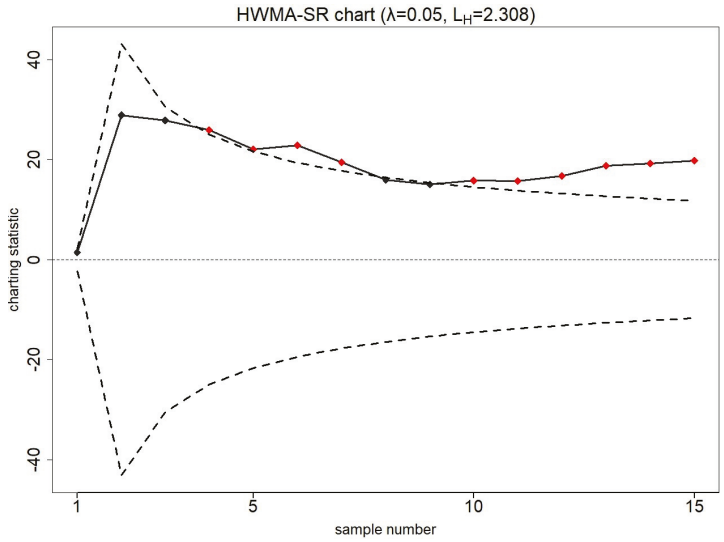


Figure 3. The HWMA-SR chart for smart phone accelerometer rings dataset.

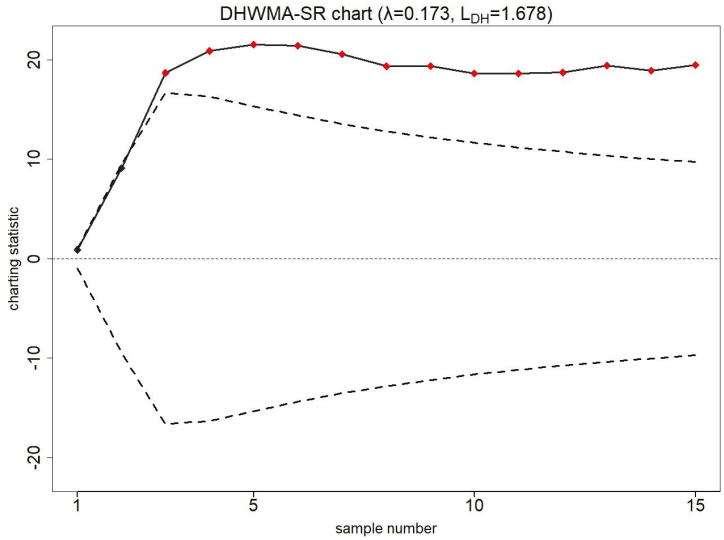


Figure 4. The DHWMA-SR chart for smart phone accelerometer rings dataset.

Table 10. Smartphone accelerometer dataset.

t	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	SR_t	E_t	D_t	H_t	DH_t
1	−6.6033	−8.6885	−5.8915	−5.3095	−8.4897	−6.5009	−5.7023	−7.0817	−7.7137	−6.2593	29	5.510	2.610	1.450	0.868
2	−7.2912	−7.0770	−6.6033	−6.7937	−5.5964	−6.6854	−7.0175	−8.1553	−7.9577	−7.7423	27	9.593	6.084	28.900	9.106
3	−7.1472	−4.8192	−7.9577	−6.9568	−6.9925	−5.8320	−7.7173	−4.2848	−8.1303	−7.1817	25	12.520	9.489	27.850	18.677
4	−6.2307	−7.3138	−6.5914	−9.2455	−8.1826	−8.6801	−6.8044	−4.8763	−7.8196	−7.2769	5	11.092	10.753	25.900	20.870
5	−8.1826	5.8701	−8.5778	−7.1448	−4.6407	−7.5007	−6.0010	−6.4176	−6.7901	−4.9751	33	15.254	13.375	22.075	21.502
6	−6.4545	−7.5507	−7.4162	−6.0986	−9.1538	−4.6109	−8.9110	−6.7925	−6.9401	−8.6302	5	13.306	13.906	22.860	21.391
7	−5.4226	−8.6837	−7.5947	−8.6432	−7.4781	−7.3578	−7.0317	−7.3531	−6.6366	−8.4278	−3	10.208	12.644	19.483	20.562
8	−7.3281	−8.3886	−5.7880	−7.1924	−7.9934	−7.6661	−6.9056	−8.3743	−8.9670	−7.1353	−9	6.558	10.078	15.971	19.329
9	−5.0822	−4.8846	−7.1758	−8.4540	−5.4048	−7.0163	−6.4581	−8.3100	−6.6187	−5.2988	35	11.962	11.064	15.050	19.352
10	−6.0010	−8.7111	−9.1169	−8.4957	−6.7568	−7.6554	−5.9820	−6.7592	−7.2603	−6.8532	5	10.639	11.001	15.767	18.597
11	−8.0029	−8.5016	−5.9796	−6.4640	−8.7111	−6.7878	−4.6990	−7.0175	−6.5759	−4.6728	25	13.368	12.230	15.690	18.611
12	−8.4314	−4.5347	−5.5547	−6.0641	−4.1598	−5.2453	−8.4350	−8.5826	−8.1826	−6.6068	29	16.338	14.342	16.736	18.702
13	−7.4959	−7.4400	−5.7975	−5.3500	−4.8989	−7.2162	−5.8011	−6.9865	−5.8904	−4.1277	49	22.544	18.496	18.758	19.423
14	−6.9270	−8.2433	−7.0175	−8.0220	−6.9639	−6.1308	−8.2957	−6.4259	−6.0034	−8.3897	11	20.350	19.856	19.185	18.886
15	−8.4040	−7.5173	−8.5457	−6.0379	−4.5264	−5.9844	−4.4597	−5.3976	−5.7523	−8.3040	35	23.134	21.886	19.800	19.462

Table 11. Piston rings dataset.

t	x_1	x_2	x_3	x_4	x_5	SR_t	E_t	D_t	H_t	DH_t
1	74.012	74.015	74.030	73.986	74.000	8	0.400	0.020	0.400	0.320
2	73.995	74.010	73.990	74.015	74.001	4	0.580	0.048	7.800	2.720
3	73.987	73.999	73.985	74.000	73.990	−14	−0.149	0.038	5.000	3.920
4	74.008	74.010	74.003	73.991	74.006	7	0.208	0.047	−0.283	3.053
5	74.003	74.000	74.001	73.986	73.997	−3	0.048	0.047	1.038	2.413
6	73.994	74.003	74.015	74.020	74.004	9	0.496	0.069	0.830	2.355
7	74.008	74.002	74.018	73.995	74.005	10	0.971	0.114	2.242	2.585
8	74.001	74.004	73.990	73.996	73.998	−6	0.622	0.140	2.550	2.258
9	74.015	74.000	74.016	74.025	74.000	12	1.191	0.192	2.381	2.665
10	74.030	74.005	74.000	74.016	74.012	14	1.832	0.274	3.550	3.063
11	74.001	73.990	73.995	74.010	74.024	4	1.940	0.358	4.095	3.052
12	74.015	74.020	74.024	74.005	74.019	15	2.593	0.469	4.636	3.584
13	74.035	74.010	74.012	74.015	74.026	15	3.213	0.606	5.500	3.954
14	74.017	74.013	74.036	74.025	74.026	15	3.803	0.766	6.231	4.311
15	74.010	74.005	74.029	74.000	74.020	14	4.313	0.944	6.807	4.613

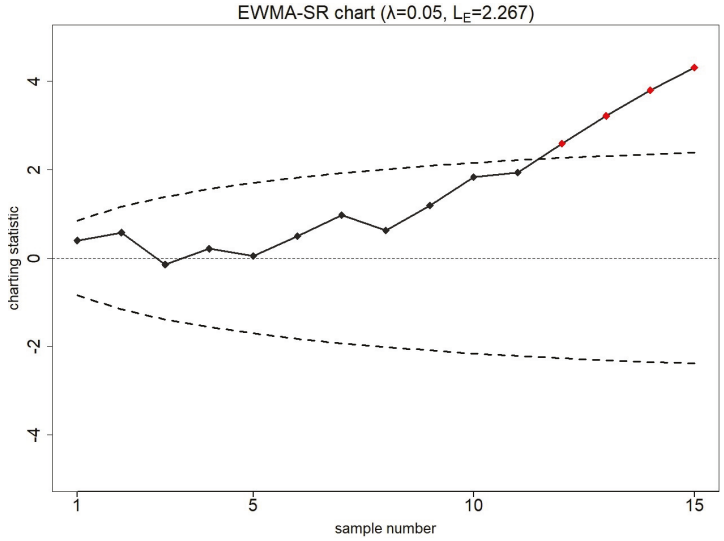


Figure 5. The EWMA-SR chart for piston rings dataset.

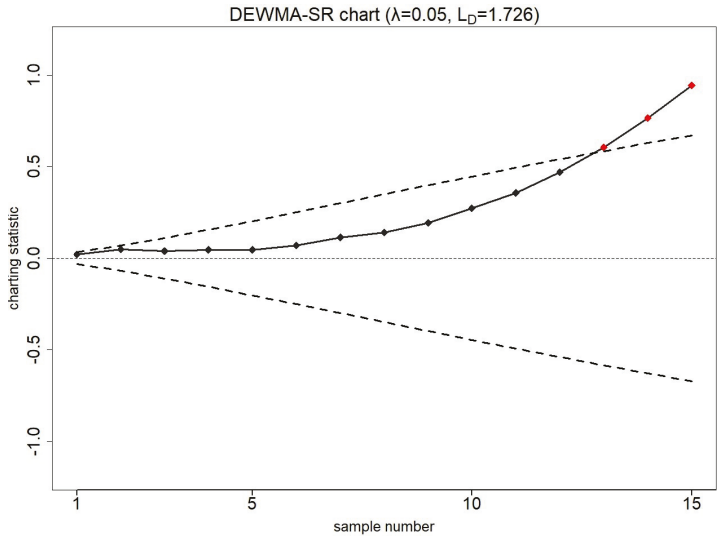


Figure 6. The DEWMA-SR chart for piston rings dataset.

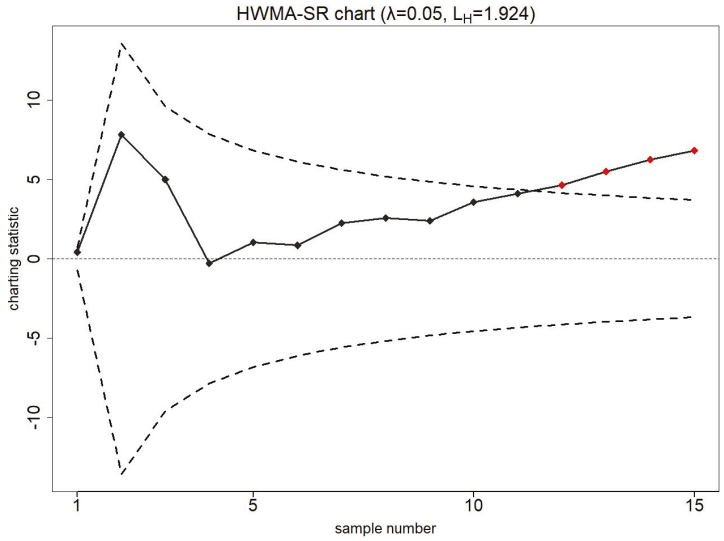


Figure 7. The HWMA-SR chart for piston rings dataset.

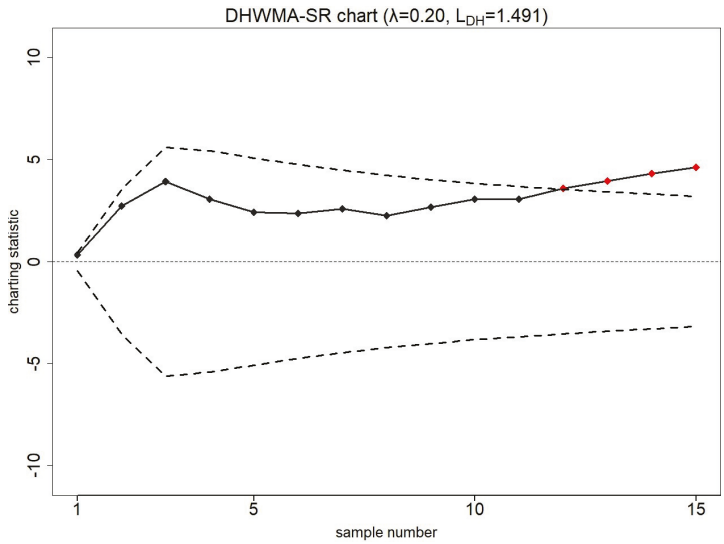


Figure 8. The DHWMA-SR chart for piston rings dataset.

8. Conclusions and Recommendations

In this article, we proposed a new nonparametric DHWMA control chart based on the signed-rank statistic (DHWMA-SR) for monitoring shifts in the location parameter of an unknown but continuous and symmetric distribution. The run-length distribution and its associated characteristics were evaluated by performing Monte Carlo simulations for both the zero- and steady-state cases. It was shown that the performance of the proposed chart improves as the value of sample size n increases and as the value of smoothing parameter λ decreases. However, the superiority of the DHWMA-SR chart for small values of λ is accompanied with a large value of $SDRL_0$. For this reason, we recommend quality practitioners to design and implement the proposed chart with λ = 0.20–0.25 for small values of n regardless of the amount of shift (which is usually unknown) and $\lambda \geq 0.15$ for n = 10–20 only for the zero-state case or if there is an evidence that the shift occurs very early in the process. Moreover, the performance of the DHWMA-SR chart was compared with that of the EWMA-SR, DEWMA-SR and HWMA-SR charts. The results indicated that the proposed chart has better detection ability than the EWMA-SR and DEWMA-SR charts, especially for small and moderate shifts, and it is superior to the HWMA-SR chart, especially for moderate and large shifts. We also compared the DHWMA-SR chart with the nonparametric DHWMA sign and the parametric DHWMA-$\overline{X}$ charts. The comparison study indicated that the proposed chart is more effective than the DHWMA sign chart under many symmetric distributions except for Laplace and, in many cases, it is more efficient than the DHWMA-$\overline{X}$ chart.

Knoth et al. [23] advised practitioners against the implementation of the HWMA chart and its extensions due to their poor steady-state performance. More specifically, they compared the HWMA and EWMA charts with design parameters that have arisen by setting as equal the IC asymptotic variances in their charting statistics. This criterion should be considered as arbitrary because the competing schemes have very different IC run-length characteristics. Alevizakos et al. [36] studied this topic and they concluded that an appropriately designed HWMA scheme is more effective than the EWMA chart for small and moderate shifts in the zero-state case. In this article, we showed that the steady-state IC run-length characteristics of the DHWMA-SR chart change as the shift occurs later and do not converge to specific values, like with the EWMA-SR and DEWMA-SR charts. Moreover,

the steady-state OOC performance of the proposed chart, as well as that of the HWMA-SR chart, is very poor, especially for moderate and large shifts.

As future work, it would be of interest to investigate in more detail the steady-state performances of the HWMA and DHWMA schemes and develop modifications to enhance them. Additionally, the proposed scheme could be applied to joint monitoring of the unknown process parameters using the Lepage or Cucconi statistics.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The author declares no conflicts of interest.

Appendix A

Table A1. IC steady-state run-length characteristics for the HWMA-SR chart for different valious of τ.

			$n = 5$						$n = 10$				
λ	L_H	Characteristic	$\tau = 1$	100	200	500	1000	L_H	$\tau = 1$	100	200	500	1000
0.05	2.070	ARL	370.48	393.87	382.08	363.12	345.08	2.308	370.74	345.11	300.33	202.37	124.89
		MDRL	188	205	181	122	66		295	274	223	121	68
		SDRL	525.06	542.16	557.24	590.75	629.56		322.32	303.07	285.06	221.42	148.02
0.10	2.165	ARL	368.91	553.66	678.70	1119.37	1915.95	2.588	370.18	334.65	303.40	252.76	222.43
		MDRL	127	116	111	119	182		282	246	211	165	145
		SDRL	809.00	1628.41	1897.00	2593.39	3435.38		329.79	320.81	304.82	264.76	234.81
0.25	2.113	ARL	370.09	1387.79	2380.22	5626.89	9952.47	2.701	372.13	374.90	370.40	366.23	362.50
		MDRL	50	57	70	186	2603		256	261	101	254	253
		SDRL	1588.98	4798.64	6219.96	8771.89	9829.65		377.96	376.66	372.54	364.23	359.17

Appendix B. Robustness of the DHWMA-SR Chart when $n = 5$ and 10 for an $ARL_0 \approx 370$ Under Different Symmetric Distributions

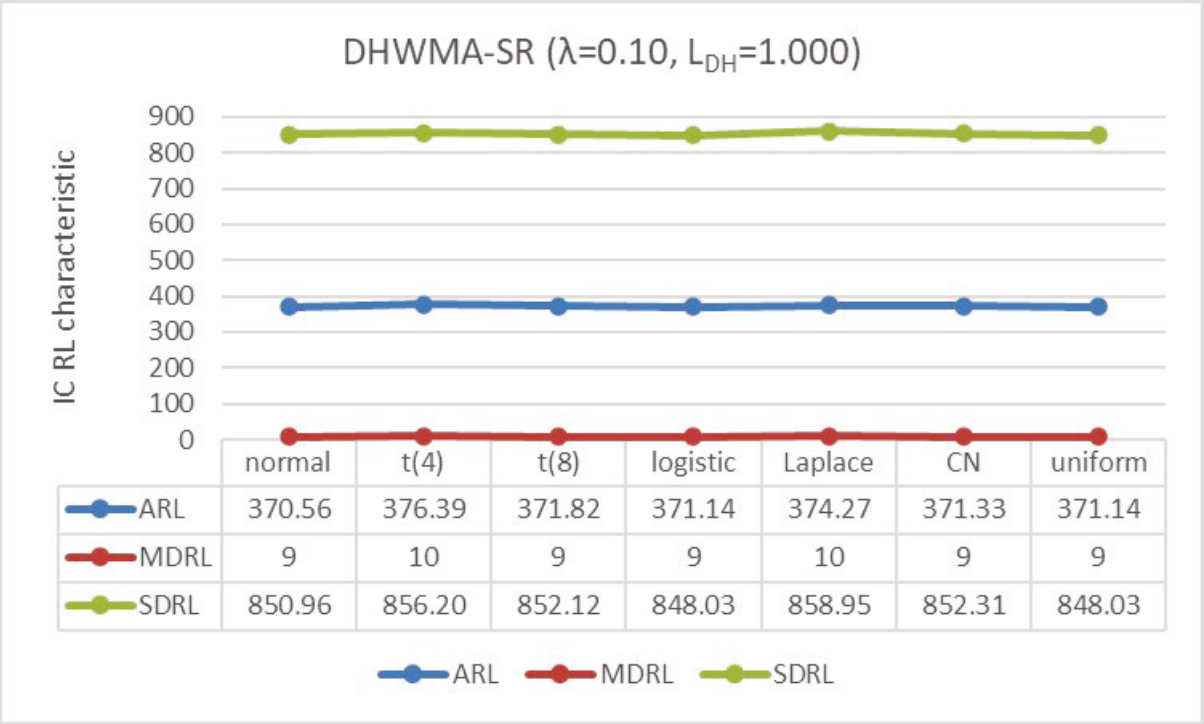


Figure A1. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.10$ and $L = 1.000$ when $n = 5$.

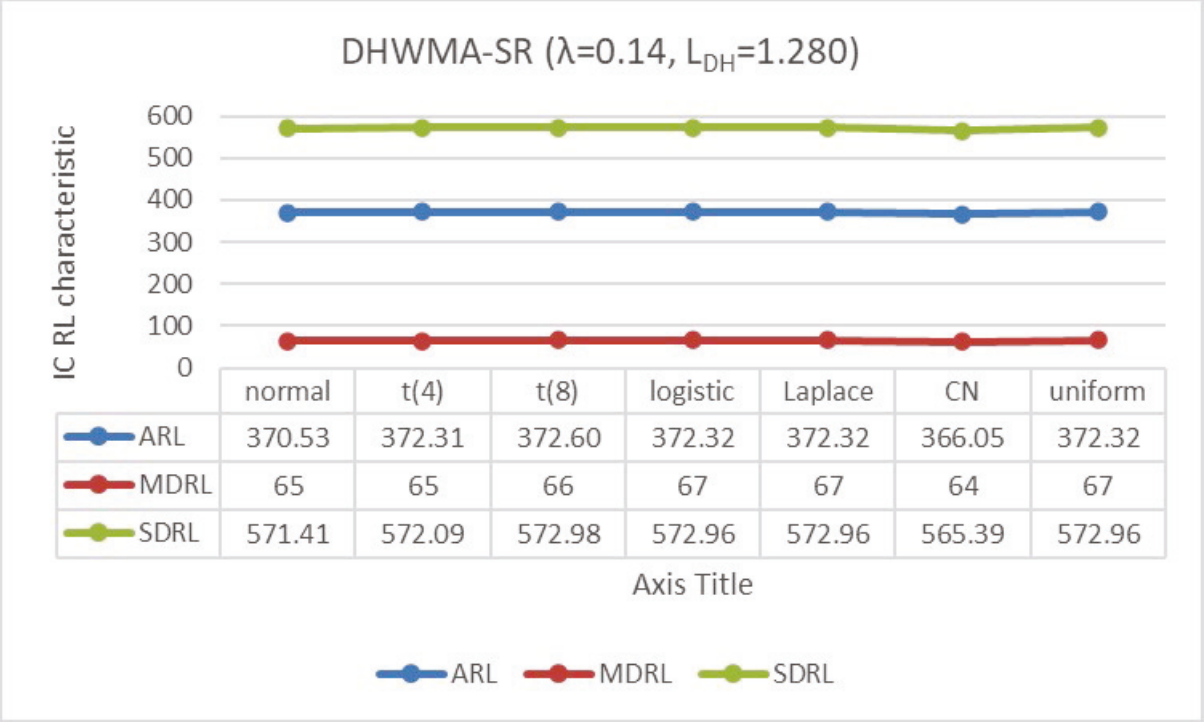


Figure A2. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.14$ and $L = 1.280$ when $n = 5$.

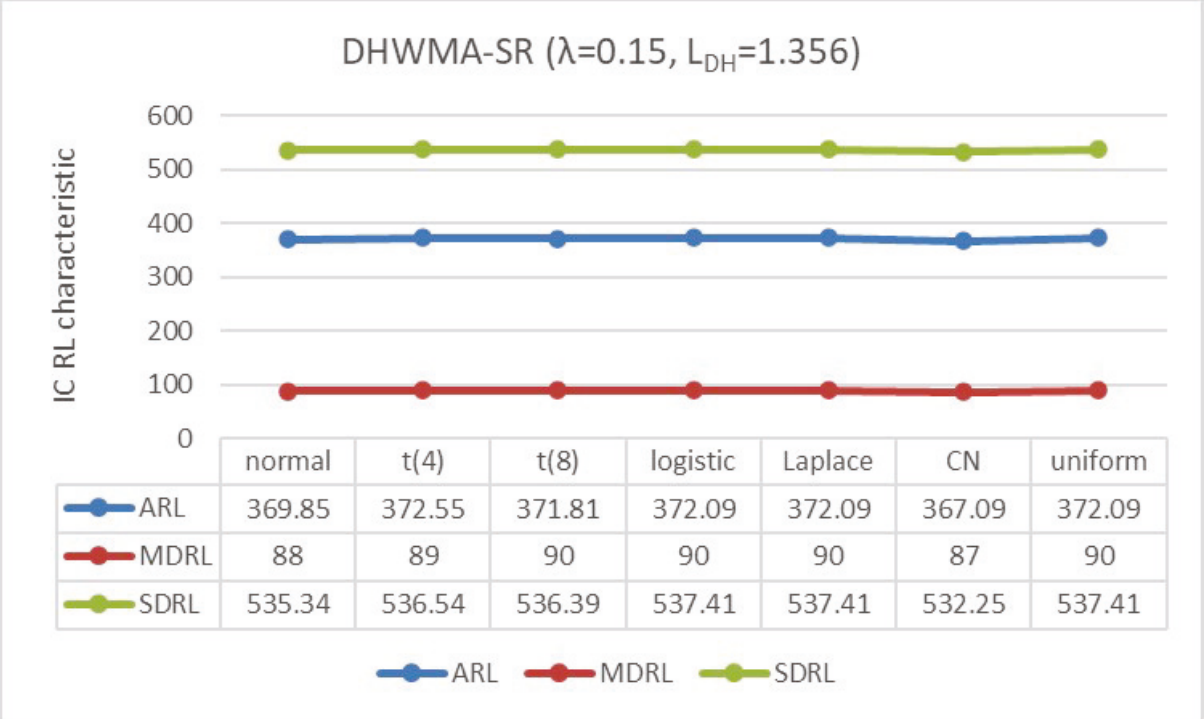


Figure A3. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.15$ and $L = 1.356$ when $n = 5$.

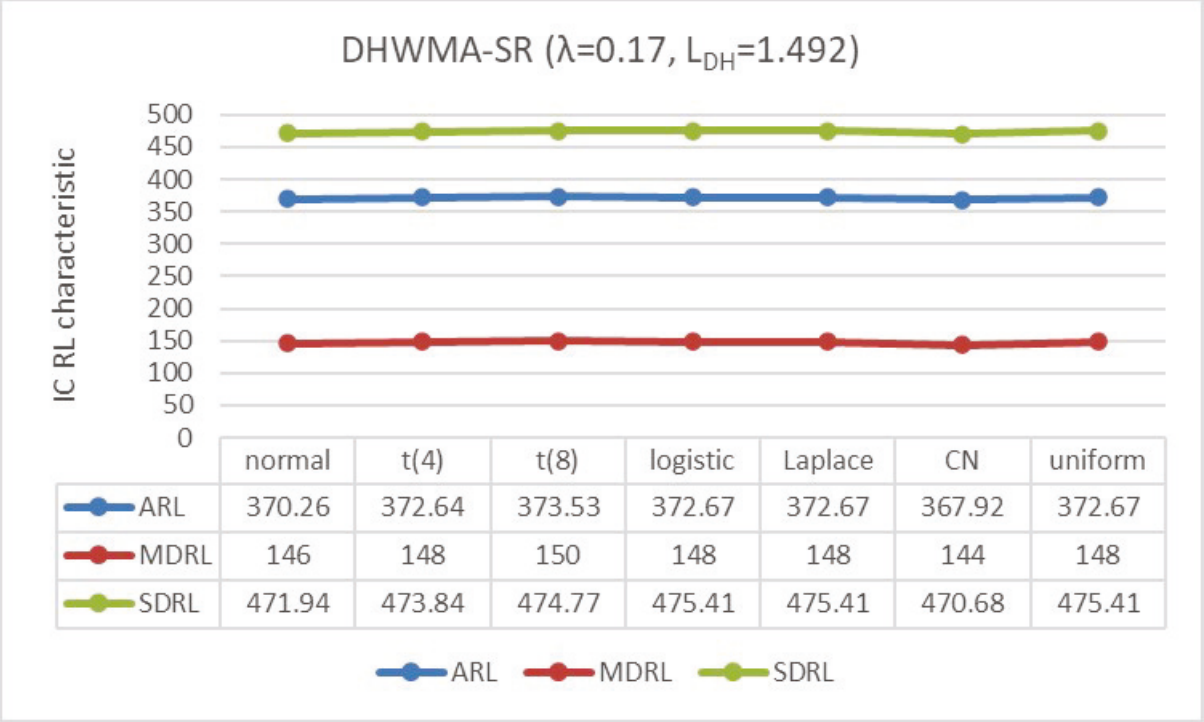


Figure A4. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.17$ and $L = 1.492$ when $n = 5$.

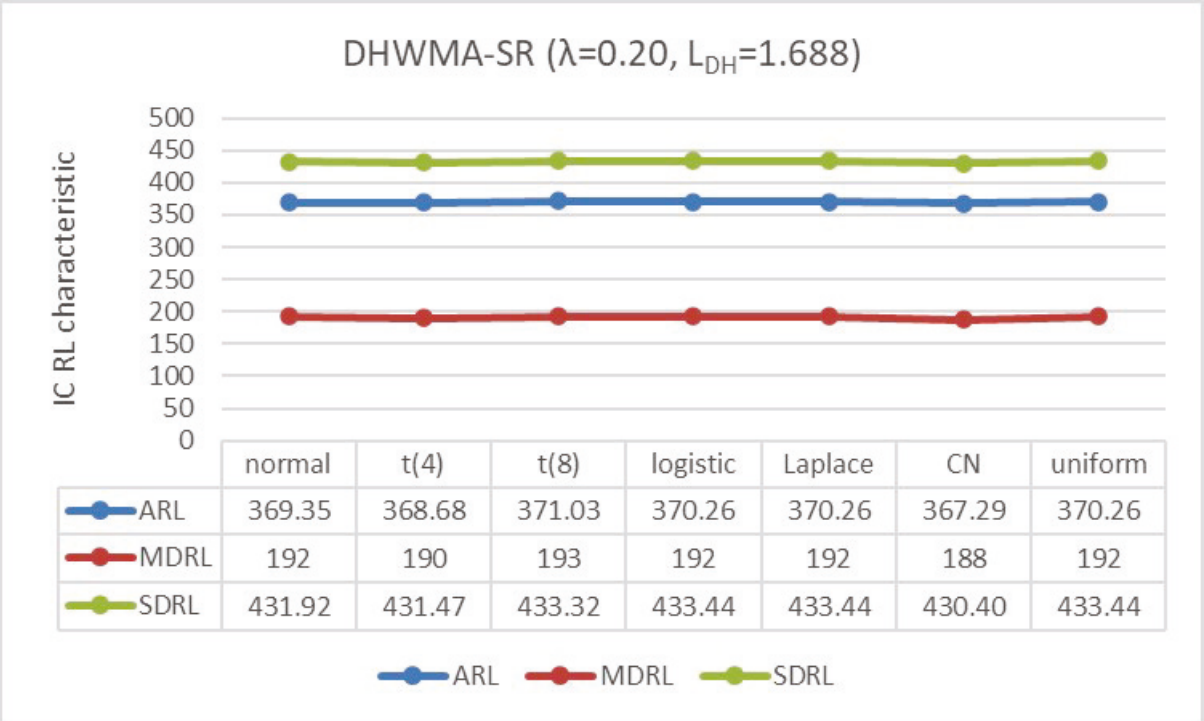


Figure A5. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.20$ and $L = 1.688$ when $n = 5$.

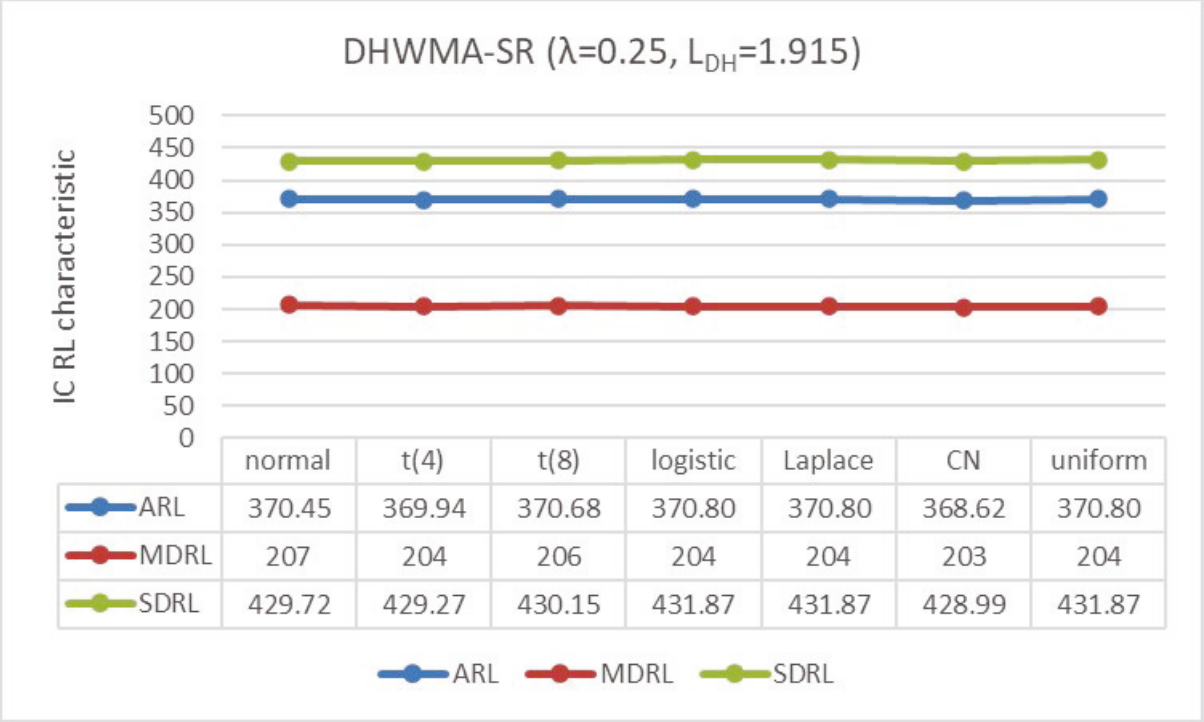


Figure A6. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.25$ and $L = 1.915$ when $n = 5$.

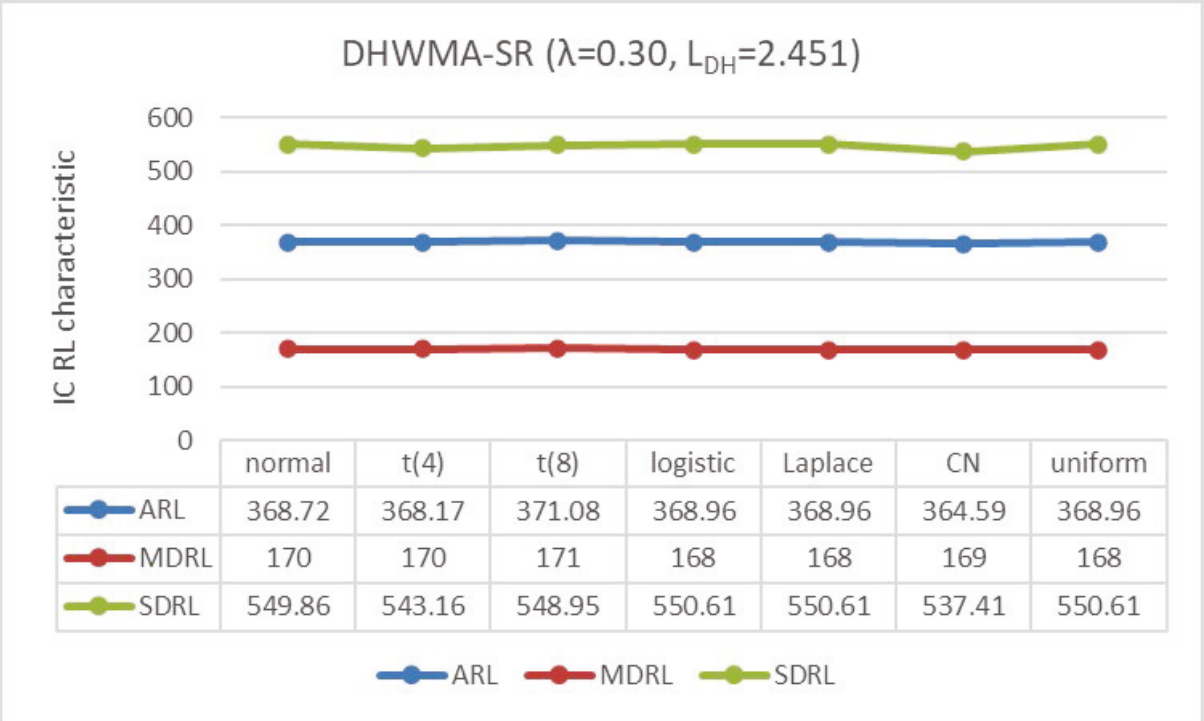
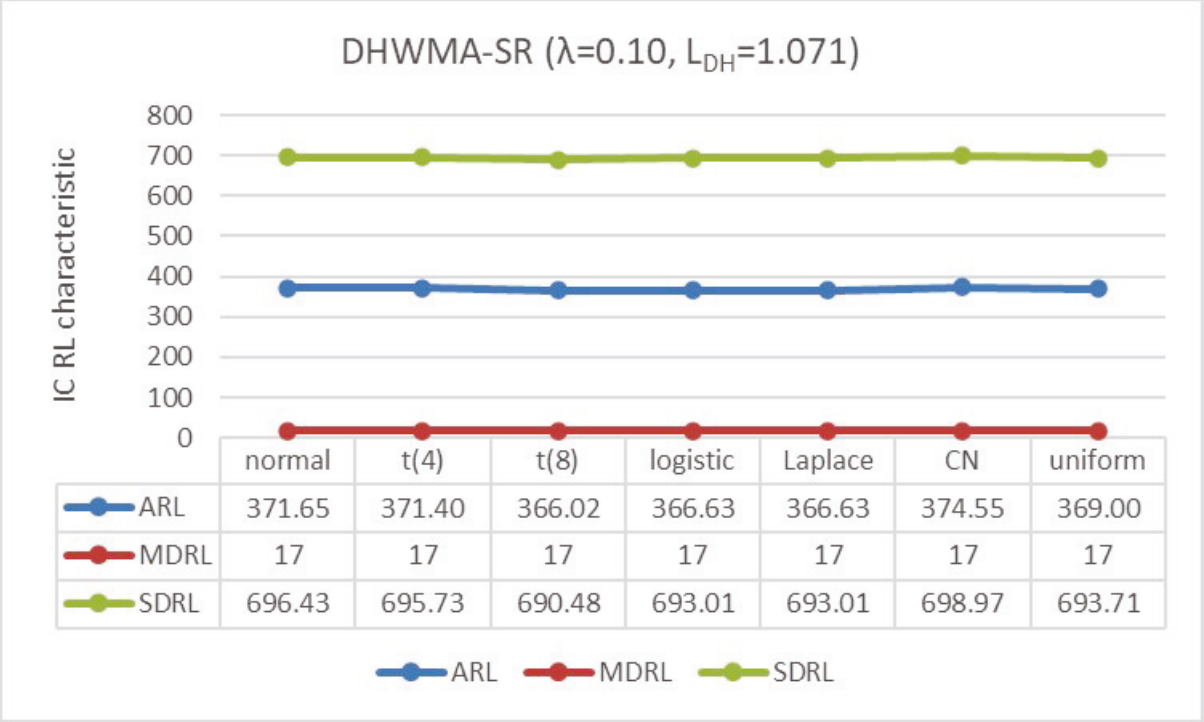
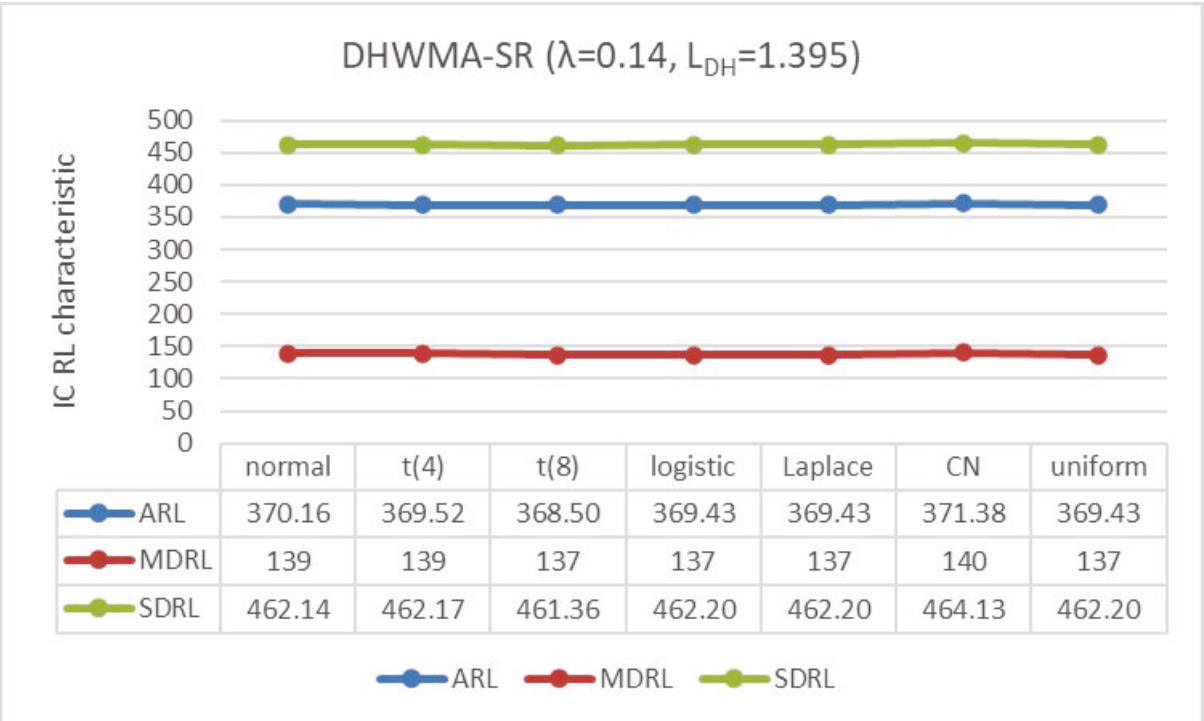


Figure A7. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.30$ and $L = 2.451$ when $n = 5$.



	normal	t(4)	t(8)	logistic	Laplace	CN	uniform
ARL	371.65	371.40	366.02	366.63	366.63	374.55	369.00
MDRL	17	17	17	17	17	17	17
SDRL	696.43	695.73	690.48	693.01	693.01	698.97	693.71

Figure A8. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.10$ and $L = 1.071$ when $n = 10$.



	normal	t(4)	t(8)	logistic	Laplace	CN	uniform
ARL	370.16	369.52	368.50	369.43	369.43	371.38	369.43
MDRL	139	139	137	137	137	140	137
SDRL	462.14	462.17	461.36	462.20	462.20	464.13	462.20

Figure A9. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.14$ and $L = 1.395$ when $n = 10$.

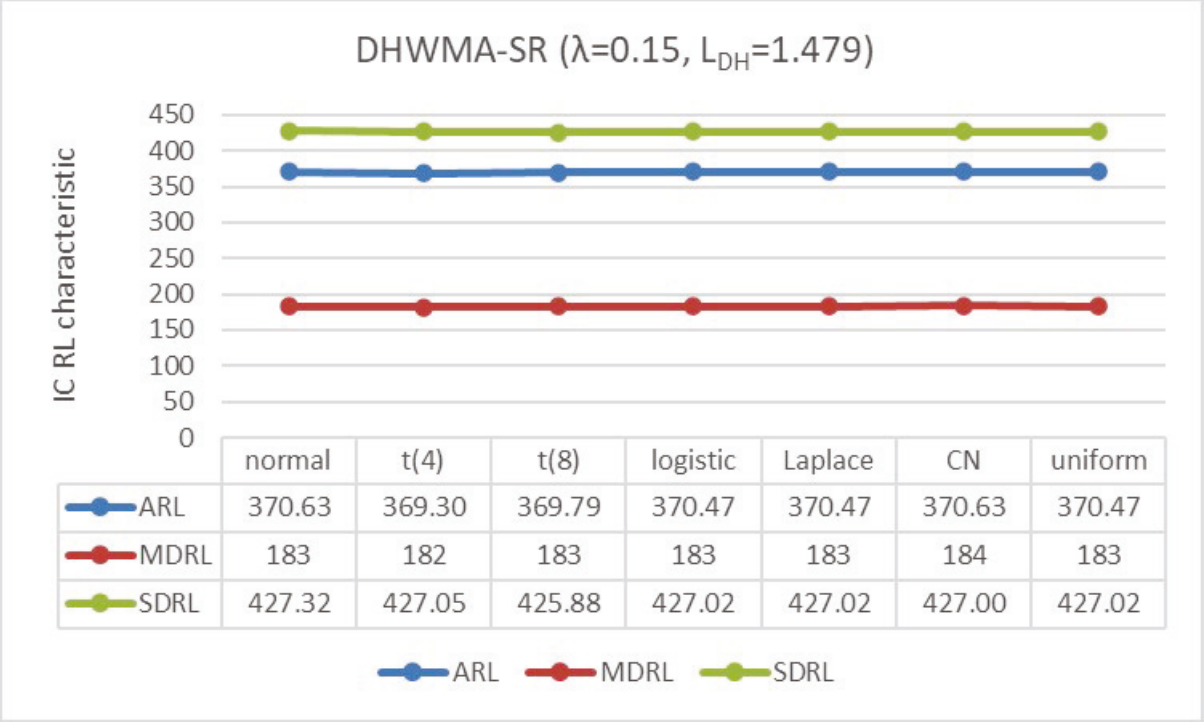


Figure A10. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.15$ and $L = 1.479$ when $n = 10$.

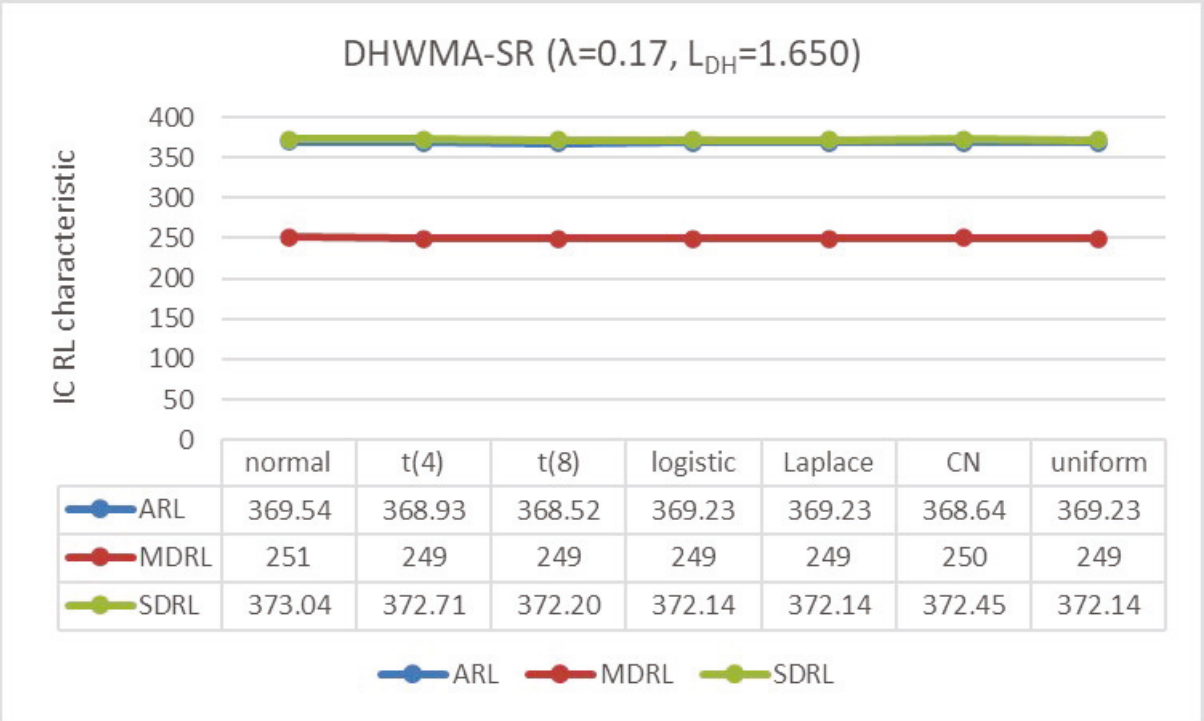


Figure A11. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.17$ and $L = 1.650$ when $n = 10$.

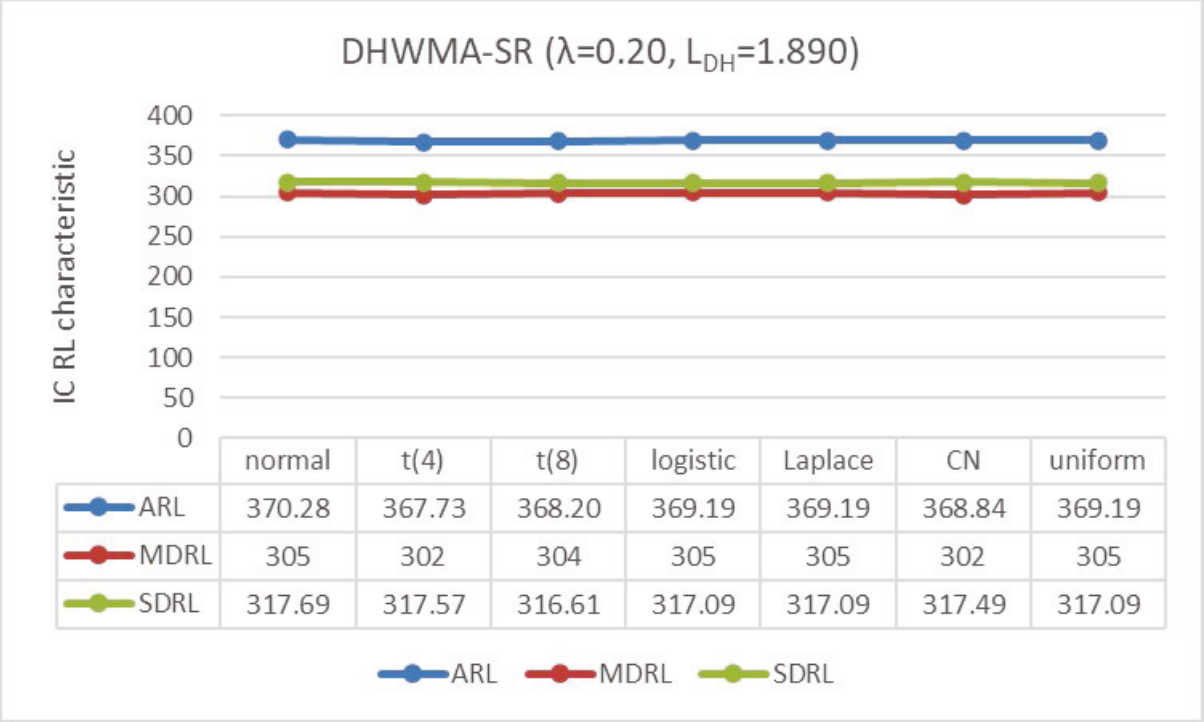


Figure A12. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.20$ and $L = 1.890$ when $n = 10$.

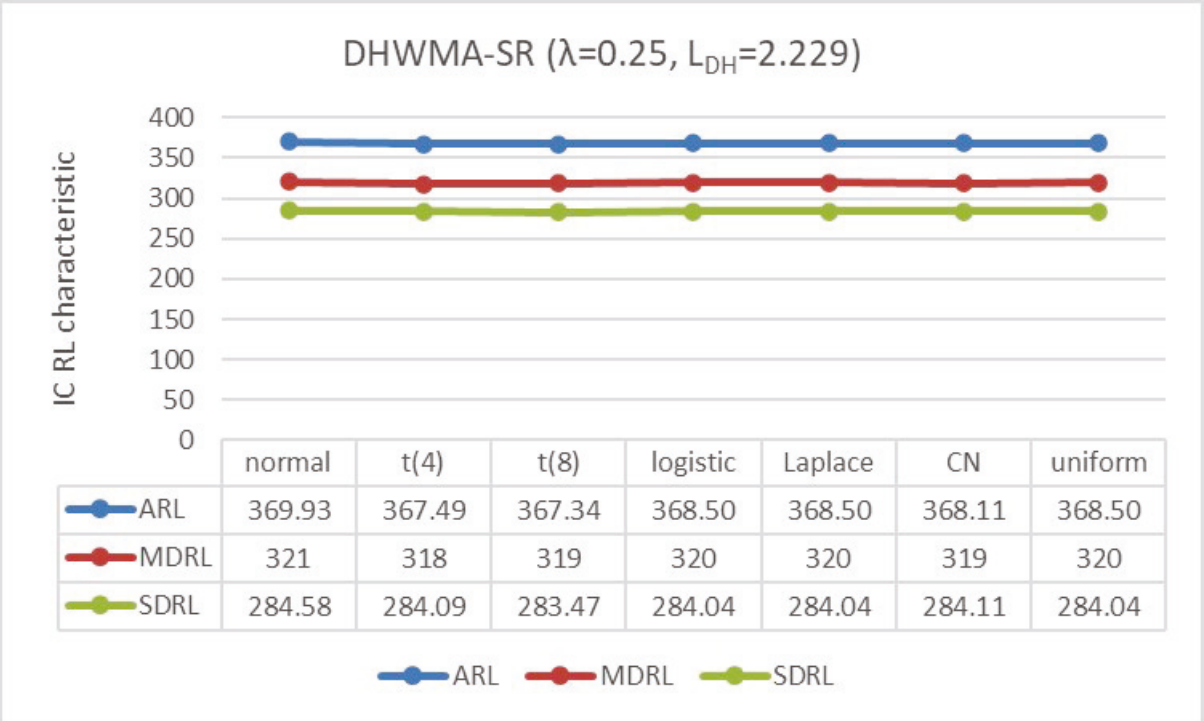


Figure A13. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.25$ and $L = 2.229$ when $n = 10$.

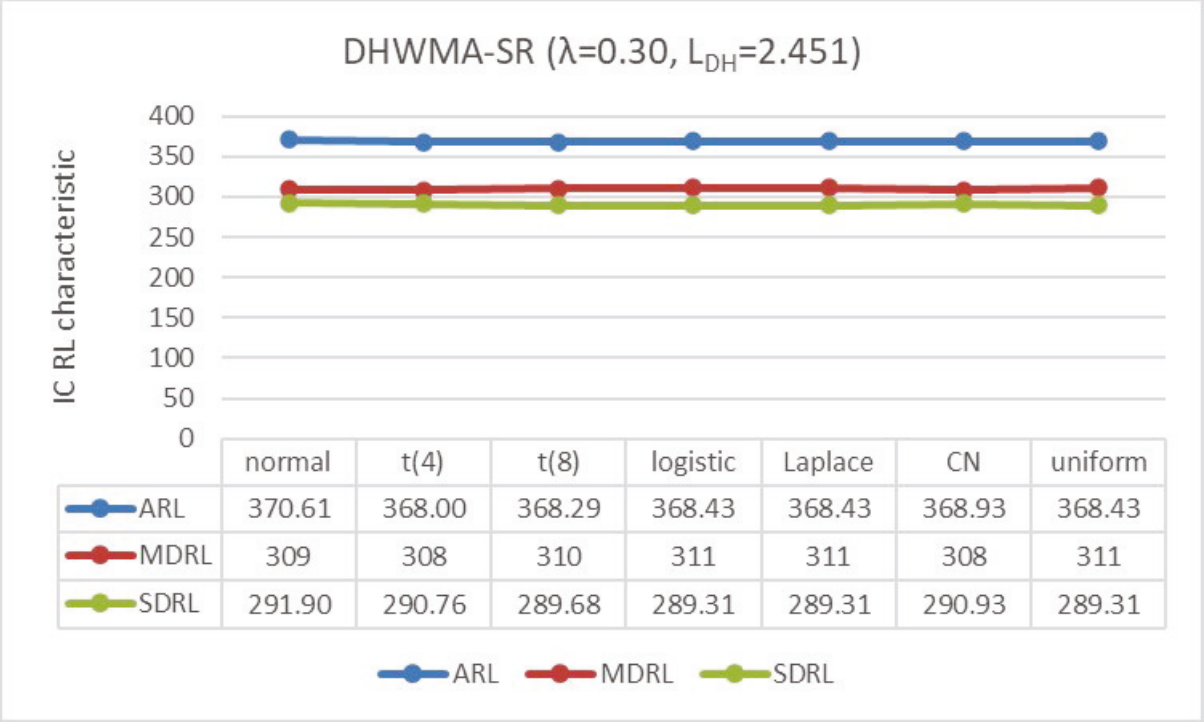


Figure A14. IC run-length characteristics for the DHWMA-SR chart with $\lambda = 0.30$ and $L = 2.451$ when $n = 10$.

References

1. Shewhart, W.A. Quality control charts. *Bell Syst. Tech. J.* **1926**, *5*, 593–603. [CrossRef]
2. Page, E.S. Continuous inspection schemes. *Biometrika* **1954**, *41*, 100–115. [CrossRef]
3. Roberts, S.W. Control chart tests based on geometric moving averages. *Technometrics* **1959**, *1*, 239–250. [CrossRef]
4. Chakraborti, S.; Graham, M.A. *Nonparametric Statistical Process Control*; John Wiley & Sons Ltd.: Hoboken, NJ, USA, 2019.
5. Chakraborti, S.; Graham, M.A. Nonparametric (distribution-free) control charts: An updated overview and some results. *Qual. Eng.* **2019**, *31*, 523–544. [CrossRef]
6. Raza, M.A.; Nawaz, T.; Aslam, M.; Bhatti, S.H.; Sherwani, R.A.K. A new nonparametric double exponentially weighted moving average control chart. *Qual. Reliab. Eng. Int.* **2020**, *36*, 68–87. [CrossRef]
7. Alevizakos, V.; Chatterjee, K.; Koukouvinos, C. A nonparametric double generally weighted moving average signed-rank control chart for monitoring process location. *Qual. Reliab. Eng. Int.* **2020**, *36*, 2441–2458. [CrossRef]
8. Raza, M.A.; Nawaz, T.; Han, D. On designing distribution-free homogeneously weighted moving average control charts. *J. Test. Eval.* **2020**, *48*, 3154–3171. [CrossRef]
9. Perdikis, T.; Psarakis, S.; Castagliola, P.; Celano, G. An EWMA-type chart based on signed ranks with exact run length properties. *J. Stat. Comput. Simul.* **2021**, *91*, 732–751. [CrossRef]
10. Perdikis, T.; Psarakis, S.; Castagliola, P.; Maravelakis, P.E. An EWMA signed ranks control chart with reliable run length performances. *Qual. Reliab. Eng. Int.* **2021**, *37*, 1266–1284. [CrossRef]
11. Alevizakos, V.; Chatterjee, K.; Koukouvinos, C. Nonparametric triple exponentially weighted moving average signed-rank control chart for monitoring shifts in the process location. *Qual. Reliab. Eng. Int.* **2021**, *37*, 2622–2645. [CrossRef]
12. Tang, A.; Ma, Y.; Hu, X.; Huang, W. The exact statistical properties of a signed-rank-based adaptive exponential weighted moving average chart. *Qual. Reliab. Eng. Int.* **2022**, *38*, 2781–2794. [CrossRef]
13. Castagliola, P.; Tran, K.P.; Celano, G.; Rakitzis, A.C.; Maravelakis, P.E. An EWMA-type sign chart with exact run length properties. *J. Qual. Technol.* **2019**, *51*, 51–63. [CrossRef]
14. Tang, A.; Sun, J.; Hu, X.; Castagliola, P. A new nonparametric adaptive EWMA control chart with exact run length properties. *Comput. Ind. Eng.* **2019**, *130*, 404–419. [CrossRef]
15. Alevizakos, V.; Chatterjee, K.; Koukouvinos, C. A nonparametric triple exponentially weighted moving average sign control chart. *Qual. Reliab. Eng. Int.* **2021**, *37*, 1504–1523. [CrossRef]
16. Qiu, P. *Introduction to Statistical Process Control*; CRC Press: Boca Raton, FL, USA, 2014. [CrossRef]

17. Koutras, M.V.; Triantafyllou, I.S. *Distribution-Free Methods for Statistical Process Monitoring and Control*; Springer: Cham, Switzerland, 2020. [CrossRef]
18. Triantafyllou, I.S.; Ram, M. Nonparametric EWMA-type control charts for monitoring industrial processes: An overview. *Int. J. Math. Eng. Manag. Sci.* **2021**, *6*, 708–751. [CrossRef]
19. Qiu, P. Some perspectives on nonparametric statistical process control. *J. Qual. Technol.* **2018**, *50*, 49–65. [CrossRef]
20. Abid, M.; Shabbir, A.; Nazir, H.Z.; Sherwani, R.A.K.; Riaz, M. A double homogeneously weighted moving average control chart for monitoring of the process mean. *Qual. Reliab. Eng. Int.* **2020**, *36*, 1513–1527. [CrossRef]
21. Riaz, M.; Abbas, Z.; Nazir, H.Z.; Abid, M. On the development of triple homogeneously weighted moving average control chart. *Symmetry* **2021**, *13*, 360. [CrossRef]
22. Alevizakos, V.; Chatterjee, K.; Koukouvinos, C. The extended homogeneously weighted moving average control chart. *Qual. Reliab. Eng. Int.* **2021**, *37*, 2134–2155. [CrossRef]
23. Knoth, S.; Tercero-Gómez, V.G.; Khakifirooz, M.; Woodall, W.H. The impracticality of homogeneously weighted moving average and progressive mean control chart approaches. *Qual. Reliab. Eng. Int.* **2021**, *37*, 3779–3794. [CrossRef]
24. Riaz, M.; Adid, M.; Shabbir, A.; Nazir, H.Z.; Abbas, Z.; Abbasi, S.A. A non-parametric double homogeneously weighted moving average control chart under sign statistic. *Qual. Reliab. Eng. Int.* **2021**, *37*, 1544–1560. [CrossRef]
25. Zhang, H.; Rasheed, Z.; Khan, M.; Namangale, J.J.; Anwar, S.M.; Hamid, A. A distribution-free THWMA control chart under ranked set sampling. *Math. Probl. Eng.* **2022**, *6*, 3823013. [CrossRef]
26. Abbas, N. Homogeneously weighted moving average control chart with an application in substrate manufacturing process. *Comput. Ind. Eng.* **2018**, *120*, 460–470. [CrossRef]
27. Adeoti, O.A.; Koleoso, S.O. A hybrid homogeneously weighted moving average control chart for process monitoring. *Qual. Reliab. Eng. Int.* **2021**, *36*, 2170–2186. [CrossRef]
28. Malela-Majika, J.-C.; Shongwe, S.C.; Adeoti, O.A. A hybrid homogeneously weighted moving average control chart for process monitoring: Discussion. *Qual. Reliab. Eng. Int.* **2021**, *37*, 3314–3322. [CrossRef]
29. Letshedi, T.I.; Malela-Majika, J.-C.; Shongwe, S.C. New extended distribution free homogeneously weighted monitoring schemes for monitoring abrupt shifts in the location parameter. *PLoS ONE* **2022**, *17*, e0261217.. [CrossRef]
30. Alevizakos, V.; Chatterjee, K.; Koukouvinos, C. An extended nonparametric homogeneously weighted moving average sign control chart. *Qual. Reliab. Eng. Int.* **2021**, *37*, 3395–3416. [CrossRef]
31. Bennett, B. On the non-null distribution of Wilcoxon's signed rank test. *Metrika* **1972**, *19*, 36–38.. [CrossRef]
32. Montgomery, D.C. *Introduction to Statistical Quality Control*, 8th ed.; John Wiley & Sons: Hoboken, NJ, USA, 2020.
33. Chan, L.K.; Zhang, J. Some issues in the design of EWMA charts. *Commun. Stat. Simul. Comput.* **2020**, *29*, 201–217. [CrossRef]
34. Riaz, M.; Ahmad, S.; Mahmoood, T.; Abbas, N. On reassessment of the HWMA chart for process monitoring. *Processes* **2022**, *10*, 1129. [CrossRef]
35. Riaz, M.; Khaliq, Q.U.A.; Gul, S. Mixed Tukey EWMA-CUSUM control chart and its applications. *Qual Technol Quant Manag* **2017**, *14*, 378–411. [CrossRef]
36. Alevizakos, V.; Chatterjee, K.; Koukouvinos, C. On the performance and comparison of various memory-type control charts. *Commun. Stat. Simul. Comput.* **2025**, *54*, 2089–2109. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Dynamic Modeling and Structural Equation Analysis of Team Innovativeness Under the Influence of Social Capital and Conflict Mediation

Ekaterina V. Orlova

Department of Economics and Management, Ufa University of Science and Technology, 450076 Ufa, Russia; ekorl@mail.ru

Abstract: The issue of modeling the personal innovativeness of project team members is determined in this study. Findings from prior research on social capital associated with innovations and innovative activities reveal that social capital factors such as trust, social networks and connections, and social values determine a person's attitude to innovations. Different connections involved in bridging (external) and bonding (internal) social capital can create conflict between project team members in different ways. To stimulate innovation in a conflict environment, a specially configured conflict management system is required that is capable of regulating the strength and intensity of the relationship between project team members. This paper analyzes the relationship between three constructs—innovativeness, social capital, and conflict. The existence of these latent constructs, which are formed by observable indicators of employees, is proven using confirmatory factor analysis (CFA). The construct of innovativeness depends on indicators such as creativity, risk propensity, and strategicity. Social capital includes observable indicators such as trust, social networks and connections, and social norms and values. Conflict consists of observable indicators of conflict between tasks, processes, and relationships. Using structural equation modeling (SEM), the causal relationship between social capital and innovativeness is substantiated with the mediating role of conflict in project groups between its participants—innovators and adaptors. The developed sociodynamic model for measuring conflict between innovators and adapters examines the required values of the controlled parameters of intra-group and inter-group connections between innovators and adapters in order to achieve equilibrium conflict dynamics, resulting in cooperation between them. This study was conducted using data from a survey of employees of a research organization. All model constructs were tested on a sample of employees as a whole, as well as for groups of innovators and adaptors separately.

Keywords: personal innovativeness; social capital; social conflict; structural equation modeling; causal inference; sociodynamic modeling

MSC: 62H25; 37N40

1. Introduction

Individual innovativeness is a personal resource, providing the subject with the ability to appear, perceive, and implement new ideas. Innovativeness is the most important factor determining one's ability to think innovatively, along with the ability to produce new ideas

and apply them in practice. Analysis of the factors of individual innovativeness has revealed two groups of factors—psychological and organizational factors [1–7]. Psychological factors contributing to individual innovativeness include personal qualities—self-efficacy, thinking styles, internal motivation and attitudes, and orientation towards achieving the goal. Organizational factors contributing to innovativeness include organizational leadership, structure, strategy, resources, and organizational culture and climate [4–7].

Other factors contributing to individual innovativeness include social networks and connections, which provide additional information that can be extracted. Therefore, the study of the social capital of an individual as a factor influencing innovativeness has become a new and relevant research task. Social capital is a set of resources that individuals and social groups possess and use for their own interest, taking the form of social networks, trust, and social norms and values [8–10].

Social capital factors—trust, social networks and connections, and social values—determine a person's attitude to innovation. Different densities and directions of bridging (external) and bonding (internal) social capital affect conflict intensity. Therefore, to solve the problem of effective development and implementation of innovations, it is necessary to take into account the direct and indirect influence of conflict on individual innovativeness.

The purpose of the investigation of individual innovativeness is to study its prerequisites (factors, indicators) and apply the acquired knowledge to develop and stimulate the innovative activity of project team members engaged in innovative activities. Knowledge of innovativeness should ensure its implementation.

The relationship of three constructs is analyzed in this study—innovativeness, social capital, and conflict. Innovativeness depends on such indicators as creativity, risk propensity, and strategicity. Social capital includes observable indicators such as trust, social networks and connections, and norms and values. Conflict consists of observable indicators of conflict in tasks, processes, and relationships.

This article is organized as follows. The Section 2 provides a literature review, reflecting theoretical and empirical studies on the influence of social capital on individual innovativeness; the connection between the architecture of the social capital of a project team and its innovations is analyzed in detail, the role of conflict in this process is assessed, and a typology of participants in social conflict within the project teams is provided. The Section 3 describes the methodology and model of this study. The Section 4 presents the results of testing the proposed model and discusses the effects of its use. The conclusion summarizes the results and shows further directions for research.

2. Literature Review

2.1. The Impact of Social Capital on Innovation

Research on management and organizational behavior examines issues related to innovativeness, studies employees' attitudes toward organizational changes [11], analyzes the relationship between employees' innovativeness and the density of their contacts [12], and diagnoses the impact of managers' innovativeness on organizational efficiency [13]. It has also been established that social capital factors influence the formation of innovations, affect creative and critical thinking, and contribute to the development of individual innovativeness. Creativity promotes the development of abilities to develop new ideas and find non-standard solutions to complex problems. A number of studies have shown the impact of socio-cultural characteristics on an individual's creative behavior [14,15] and have examined intercultural sensitivity (tolerance, attention to other people) [16]. One study [17] identified personality traits that are significantly associated with successful innovations in households. The five-factor model of personality was used, consisting of

five main personality traits: openness, extroversion, conscientiousness, agreeableness, and neuroticism. Possible ways to increase successful innovation were proposed, and the choice of components represents a spectrum from technological to hidden innovations and social innovations. The fundamental relationship between social capital and innovation was proposed by R. Putnam [18], who argued "trust promotes cooperation", clearly defining the importance of trust as a factor in cooperation. In the study by V. Tsai [19], a significant positive relationship was found between the social capital of companies and their ability to create innovations. M. Dahli [20] empirically confirmed the important role of trust as a driving force for innovation. By facilitating exchange and reducing the need for time-consuming and expensive monitoring, trust promotes broader and seamless cooperation and a freer exchange of information, which can ultimately lead to more activities and inventions related to innovation. R. Landry [21] also considered social capital to be the most important and influential factor in deciding to implement innovations. In addition, social capital can be considered as a decisive factor in the desire of an organization to become more innovative. The concept of trust, which is important for innovation, is studied in [22,23], which argue that "collaboration generates trust".

Based on a deep analysis of sources on the topic of social capital associated with the innovative activity of the subject, we assume that social capital factors such as trust, social networks and connections, and social values shape a person's attitude towards innovations [13, 15,18,24–26]; moreover, individual innovativeness is determined by factors such as creativity, or the ability to produce new ideas and concepts, a propensity to take risks in the development and implementation of innovations, and strategicity, which is the ability of an individual to set and orient their behavior towards achieving long-term goals [27–30].

2.2. The Relationship Between the Architecture of Social Capital of a Project Team and Its Innovations: The Role and Impact of Conflict

A number of recent scientific papers have shown that the social capital of an organization or its project teams, based on factors such as social networks, values, and trust, can be a catalyst for team innovation. The evidence for such connections is based on the models and structures of social relations and their impact on the innovations in project teams. The construct of the social capital of a project team is formed by the resources available to the group through social relations. There are two types of social capital: bridging, which is outwardly oriented and unites members of different teams, and bonding, which emphasizes connections within one team.

The relationship between a project team's social network architecture and its innovations is discussed in the literature. There are different opinions and approaches to explaining how and when a team's social capital influences its innovativeness, as well as what is the nature of the connection between the team's social network and the innovations it generates. It has been shown that bridging and bonding social capital do not directly affect the creativity and innovativeness of a project team [31,32] or can even harm the generation of innovative ideas [33,34]. This paper provides a logical explanation for these patterns.

The first mechanism of the influence of bridging social capital (through social networks and connections) on innovation through conflict mediation is as follows. The positive effects of bridging social capital on the generation of innovation within a project team can be reversed due to conflict within the team, which arises in response to information brought by group members from external sources. Thus, bridging (inter-group) social capital can increase the likelihood that group members will propose new rules for establishing intra-team relationships or new, better ways to distribute project tasks, processes, and resources. These new rules may differ from those already existing in the team and

may provoke conflicts. A conflict over establishing the order of implementing business processes may develop into a conflict of intra-team relations between its members or even develop into interpersonal hostility. Such conflict in business processes and interpersonal relationships can distract team members from completing work tasks, hinder the processing and exchange of information within the group, hinder effective creative work, and ultimately suppress the cognitive functions of team members.

A destructive conflict arises. The purpose of such conflict is to cause damage or, harm to the opposite (conflicting) party, because of which team members enter into open confrontation with each other, strive to destroy relationships, worsen the psychological well-being of participants, reduce the self-esteem of group members, spread aggression, and contribute to the isolation of individual group members. All this inevitably reduces the productivity and efficiency of research and development processes.

The second mechanism of the influence of binding social capital (through trust in the group) on innovation through conflict mediation is as follows. Project teams with a high level of bonding (intra-group) social capital have a high level of trust between team members and have approximately equal access to information. Such teams are less prone to intra-team conflict relations, as they are considered inappropriate. At the same time, informal hierarchies and structures in relation to the implementation of project tasks and business processes can be established in such teams. The nature of research and development processes implies differentiation of the roles of team members, their responsibilities, and resources in relation to the implementation of project tasks. This is clearly manifested in the implementation of two types of tasks, the first of which are directly related to creative activity (performed by innovators), and the second are aimed at implementing already-developed ideas and concepts (performed by adapters). A constructive, productive conflict arises. The purpose of such conflict is to improve the situation and develop new ideas. Participants strive for cooperation and mutual understanding, and the relationships between them develop, which contributes to the growth of their productivity.

If there are effective processes of knowledge exchange and information integration within the project group, then the total information resource of the group, corresponding to its social capital, will have a positive effect on the team's innovativeness. Disagreements, discussions, and disputes between group members related to the tasks can bring about the formation of new alternatives for their resolution, encourage the team to re-evaluate the task, identify its non-obvious parameters and factors, and contribute to the generation of new ideas.

Conflict is also useful at the stage of idea implementation if constructive disagreements arise about how to implement creative ideas in practice. Constructive conflict can be useful for innovation for two reasons. Firstly, it can contribute to a psychologically safe environment in which critical thinking, diversity of opinions, creative thinking, and risk-taking are encouraged, and team members are not afraid of a negative reaction to their bold ideas. Secondly, the positive effect of conflict on innovation is realized through constructive disputes and open discussion of opposing points of view, and then, constructive conflict acts as a catalyst for change and innovation.

2.3. Types of Participants in Social Conflict in Project Teams

In cognitive psychology, it is shown that all people can be divided into creative and non-creative (or reproductive personalities) [35]. In [36], an adaptation–innovation theory is proposed, which distinguishes two behavioral (creative) styles of individuals depending on the propensity for innovation—innovators and adapters. The value of such differentiation of behavioral types, depending on the complexity and type of tasks being solved, is that

it allows for the formation of balanced project teams taking into account the features of behavioral styles:

- Project team members with an adaptive behavioral style (adapters) cope better with current, operational tasks and contribute to maintaining the short-term sustainability of the enterprise; project team members with an innovative style (innovators) contribute to the formation of new ideas, and projects, solutions that determine the strategic development of the enterprise;
- Current, regular tasks are best solved by participants–adapters; for solving tasks with a high-risk component or in crisis conditions, participants–innovators are the most effective;
- Participants–adapters always form solutions to a problem situation that are adequate to external conditions and factors and also acceptable from the standpoint of different efficiency criteria; participants–innovators can offer solutions that are not always obvious or relevant to current conditions and may not directly produce the desired effects in the short term;
- Participants with different innovative thinking styles use available information differently when solving problems. Having received new data, adapters implement them into existing data structures or the existing context of the problem being solved. Innovators extract knowledge from new data, based on which they change the existing data structure, forming new paradigms.

The listed features of different innovative styles of participants in the innovation process also determine the differences in their mutual perception of each other. Innovators characterize adapters as employees who are not inclined to take risks, they are not flexible conformists, and their behavior is easy to calculate (predict). Adapters characterize innovators as risky, ready to break any trends. An adapter worker, finding himself in a project team, the participants of which are mainly innovators, can copy the behavioral style of most participants for the purpose of adaptation, but in the short term, changing the behavioral style is impossible. Therefore, the presence of workers with antagonistic behavioral styles in one project group can not only provoke psychological discomfort but also lead to conflict situations, and as a consequence, to a decrease in worker productivity.

At the same time, for the effectiveness of innovative projects, it is necessary to involve employees of both behavioral styles—innovators and adapters. The more homogeneous the team is in terms of creative styles, the less conflict can arise between its members. At the same time, the diversity of project tasks to be solved requires a project group that is more heterogeneous in its behavioral profile, which, in turn, increases the likelihood of conflict situations and increases the complexity of management decisions.

3. Research Methodology

3.1. Hypotheses and Research Model

Based on the analysis of theoretical positions and the scientific literature regarding the role of conflict in the processes of transformation of social capital into innovation, the following conceptual model is proposed, as shown in Figure 1.

When formulating hypotheses, we distinguish bridging social capital (which facilitates inter-group communication) of group members and bonding social capital (which facilitates intra-group connections and communication) of group members.

If bridging social capital exceeds a group's bonding social capital, then it can be assumed that conflict will have a negative, destructive impact on the group's innovativeness

for members of that group. This may be typical for adaptors, who are susceptible to external influence from members of another group.

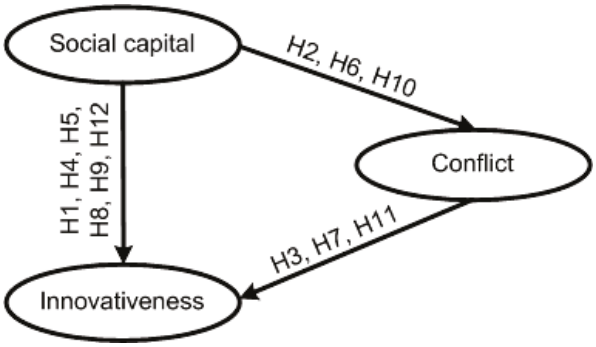


Figure 1. Conceptual model of the study.

If bridging social capital is lower than bonding social capital, then conflict will obviously have a positive and constructive impact on the group's innovativeness. This situation may characterize innovators, who are less susceptible to external influence.

If bonding social capital is high for members of one group, then social capital will have a positive impact on the level of conflict within that group. This situation may be typical for adaptors. This is due to the close interactions between group members and the potential for disputes and discussions in conflicting processes, tasks, and relationships.

If bonding social capital is low, then conflict in such a group may decrease as social capital increases, which may be characteristic of innovators. We also assume that social capital positively influences team innovation regardless of the magnitude and ratio of bonding and bridging social capital.

The main hypotheses tested are as follows:

H1: *Social capital has a direct positive effect on innovativeness.*

H2: *Social capital has a direct positive effect on conflict.*

H3: *Conflict has a direct positive effect on innovativeness.*

H4: *Conflict mediates the relationship between social capital and innovativeness.*

H5: *Innovators' social capital has a direct positive effect on innovators' innovativeness.*

H6: *Innovators' social capital has a direct positive effect on innovators' conflict with adapters.*

H7: *Innovators' conflict with adapters has a direct negative effect on innovators' innovativeness.*

H8: *Innovators' conflict with adapters mediates the relationship between social capital and innovativeness.*

H9: *Adapters' social capital has a direct positive effect on adapters' innovativeness.*

H10: *Adapters' social capital has a direct negative effect on innovators' conflict with adapters.*

H11: *Innovator–adaptor conflict has a negative positive effect on adaptor innovativeness.*

H12: *Innovator–adaptor conflict mediates the relationship between social capital and innovativeness.*

Structural equation modeling (SEM) methods are approaches and algorithms that allow for testing multivariate models. SEM is a hybrid of factor and path analysis, its goal is similar to the goal of factor analysis, and it consists in substantiating the relationships between observed characteristics of the object of study. Like path analysis, SEM is used to test the hypothesized relationships between constructs.

Three model constructs are proposed, as shown in Figure 2. Table 1 presents descriptions of the construct components.

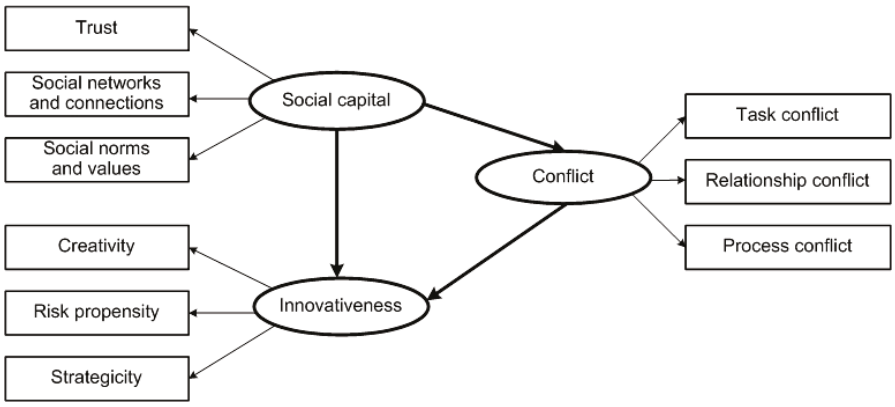


Figure 2. Proposed structural model.

Table 1. Constructs, indicators, and studies.

Construct and Indicators	Description	Theoretical Basis
Construct: Social capital SC	The set of resources that individuals and social groups possess and use to their advantage through social networks, trust, and shared norms and values	[19,29,37–43]
SC1: Trust	The employee's willingness to maintain trusting (formal and informal) relationships with his colleagues in the organization or in the team	
SC2: Social networks and connections	Networks of interaction between workers for resource exchange	
SC3: Social norms and values	Degree of civil cooperation	
Construct: Innovativeness INN	A person's readiness for innovation, that is, his ability to develop new ideas and new technologies and also to implement them	[11,14,28,31,44]
INN1: Creativity	Creativity in the form of cognitive and social processes forms the ability of an individual to produce new ideas and concepts and find non-standard solutions to problematic tasks in the field of his work activity	
INN2: Risk propensity	Risk propensity is defined as a personality trait that manifests itself in a preference for activities associated with risk without full confidence in success and a relatively easy acceptance of risky decisions	
INN3: Strategicity	Thinking style, ability to acquire and develop new qualities; ability to make decisions that will lead to the desired goal	
Construct: Conflict K	Conflict experienced by the project team	[45–49]
K1: task conflict	Task conflict is associated with differences in views and opinions regarding a team task	
K2: Relationship conflict	Relationship conflict involves tension and hostility between coworkers over interpersonal issues	
K3: Process conflict	Conflict of processes involves disagreements about the distribution, sequence of tasks, assignment of responsibility and resources to specific processes	

3.2. Measurement and Analysis Methods

The study used confirmatory factor analysis (CFA) and structural equation modeling (SEM) to examine the relationships between three constructs: social capital, innovativeness, and conflict. The observed variables included three indicators representing SC, three indicators related to INN, and three indicators related to K (see Table 1 for a description of each indicator). The selected multi-item scales for each construct were as follows.

Social Capital (SC). This construct is measured using the scales used in the studies [9,36]. The scales include 7 questions on indicator SC1: trust; 5 questions on indicator SC 2: social networks and connections; and 5 questions on indicator SC3: social norms and values.

Innovativeness (INN). This construct is measured using the scales used in the study [9]. The scales include 4 questions on indicator INN1: creativity; 2 questions on indicator INN2: risk propensity; and 4 questions on the INN indicator 3: strategicity.

Conflict (K). This construct is measured using the scales proposed in the studies [49]. The scales include 2 questions for each indicator, i.e., K1: task conflict, K2: relationship conflict, and K3: process conflict.

Obviously, different versions of scales and questions can be used for each construct. However, this study used the scales that were substantiated and used in [8,9,49]. A sample of 268 employees from a research institute is used for the study. A five-point Likert scale is adopted for the questionnaire in this study [50]. A fragment of the questionnaire is presented in Table 2.

Table 2. Rating scales. Please express your opinion on social capital (SC), innovativeness (INN), and conflict (K) over the past year. Five-point Likert scale: (1) strongly disagree; (2) moderately disagree; (3) neither agree nor disagree; (4) moderately agree; (5) strongly agree.

Indicator	Questions	Scales
SC1	1. Knowledge of the company's strategic goals and objectives 2. Teamwork 3. Satisfaction with the team and colleagues, the degree of cooperation in the team 4. Desire of colleagues to help when asked a question or request 5. Desire to share experience and knowledge with colleagues 6. Frequency of manager's approach with a question on improving the characteristics of a product/service 7. Frequency of asking management for help	1 2 3 4 5

Then, all variable values were normalized to the interval from 0 to 1. In this study, the measurement models are first assessed using reliability and validity tests. Then, CFA and SEM methods are used to test the hypotheses proposed in the theoretical model. SEM uses different types of models to map the relationships between observed variables grouped into constructs (latent variables) and substantiates causal relationships between these constructs based on quantitative tests of the proposed theoretical model. Different theoretical models can be tested based on SEM, which hypothesizes how sets of variables define constructs and how these constructs are related to each other.

The observed variables, or indicators, are variables related to social capital, innovativeness, and conflict. First, confirmatory factor analysis (CFA) is used to test the factor structure of the set of observed variables. Then, SEM is used to test the hypotheses.

3.3. Sociodynamic Model of the Conflict Between Innovators and Adaptors

The conflict between two groups is considered. The first group includes innovators as developers of innovations; the second group includes adapters as executors, implementers of ready innovations. A conflict may arise between these two groups of participants, since they have opposite goals and objectives. The behavior of participants in these groups in a conflict is described, taking into account their individual attitude, or preference, for choosing methods for resolving the conflict. The average value of such preferences for group n is designated by a discrete variable S_n^0, which can take values corresponding to behavioral styles in the conflict. Each group of participants forms its own behavioral trajectory, choosing at each moment in time one of the presented behavioral styles.

To characterize the behavioral style, the variable S_n^0 is introduced; for innovators, this variable varies in the range from $-L_1$ to L_1: $S_1^0 = \overline{-L_1; L_1}$, where negative values indicate the interaction style of cooperation, collaboration, and a tendency to negotiate, while positive values describe the style of rivalry, competition, and a tendency to protracted conflict. For adapters, the style of interaction in conflict is described by the variable $S_2^0 = \overline{-L_2; L_2}$, the values of which are defined in the same way as for innovators. This reflects the desire of the conflicting participants to compromise. The characteristic S_n^0, reflecting the average preference of the participants in the conflict for a certain method of conflict behavior, is called as involvement in the conflict, as a preference for choosing one of the styles of conflict behavior.

During the conflict, each group member interacts with both members of his or her group and members of the opposing group. To characterize these interactions, variables of intra-group interactions j_1 and j_2, as well as inter-group interactions k_{12} and k_{21}, are introduced. k_{12} reflects the intensity of inter-group interactions of group 1 members on group 2 members. Similarly, k_{21} shows the intensity of the impact of group 2 members on group 1 members. Intensities vary between 0 and 1, which shows low and high levels of interactions between participants.

When forming their own position in the conflict, members of both groups take into account the positions of the opposing side. At time t, the average preferences take on the value S_n, where n is the group number, n = 1.2. These average preferences are defined as the weighted average values of the preferences of group members. Note that the values S_n^0 and S_n differ, in that S_n^0 reflects the average preference for the method of resolving the conflict of group members, provided that there are no interactions between members within and between groups. When such interactions occur, preferences change and are characterized as average group preferences, S_n.

We use the general theory of magnetism and the theory of paramagnetism by P. Langevin and L. Brillouin [51] to assess magnetization, associating it with the degree of inclination (preference) and the level of involvement of participants in a particular group in the method of conflict resolution (from extreme competition to close cooperation). Let us formulate a model that describes the dynamics of style preferences in conflict interactions between groups, taking into account the delay factor associated with the informational reflection of the interaction between innovators and adapters. The average values $S_1(t+1)$, $S_2(t+1)$ of style preferences of conflict interactions and participants of groups of innovators and adapters are formed on the basis of exponential intensities of interactions of innovators and adapters within their groups and between opposing groups:

$$
\begin{aligned}
S_1(t+1) &= \frac{\sum\limits_{S_1^0=-L_1}^{L_1} S_1^0 \exp\left(S_1^0[j_1 S_1(t) + k_{12} S_2(t)]\right)}{\sum\limits_{S=-L_1}^{L_1} \exp\left(S_1^0[j_1 S_1(t) + k_{12} S_2(t)]\right)}, \\
S_2(t+1) &= \frac{\sum\limits_{S_2^0=-L_2}^{L_2} S_2^0 \exp\left(S_2^0[j_2 S_2(t) + k_{21} S_1(t)]\right)}{\sum\limits_{S_2^0=-L_2}^{L_2} \exp\left(S_2^0[j_2 S_2(t) + k_{21} S_1(t)]\right)}.
\end{aligned} \tag{1}
$$

Using the Brillouin function, we replace Formula (1):

$$
\begin{aligned}
S_1(t+1) &= L_1 \cdot B_{L_1}(z_1) = L_1\left(\tfrac{L_1+\frac{1}{2}}{L_1} \cdot \operatorname{cth}\tfrac{L_1+\frac{1}{2}}{L_1} \cdot z_1 - \tfrac{1}{2L_1}\operatorname{cth}\tfrac{z_1}{2L_1}\right), \\
S_2(t+1) &= L_2 \cdot B_{L_2}(z_2) = L_2\left(\tfrac{L_2+\frac{1}{2}}{L_2} \cdot \operatorname{cth}\tfrac{L_2+\frac{1}{2}}{L_2} \cdot z_2 - \tfrac{1}{2L_2}\operatorname{cth}\tfrac{z_2}{2L_2}\right),
\end{aligned} \tag{2}
$$

where

$$
\begin{aligned}
z_1 &= j_1 S_1(t) + k_{12} S_2(t), \\
z_2 &= j_2 S_2(t) + k_{21} S_1(t),
\end{aligned} \tag{3}
$$

By placing Expression (3) into Equation (2), we obtain:

$$
\begin{aligned}
S_1(t+1) &= L_1\left(\tfrac{L_1+\frac{1}{2}}{L_1} \cdot \operatorname{cth}\tfrac{L_1+\frac{1}{2}}{L_1}(j_1 S_1(t) + k_{12} S_2(t)) - \tfrac{1}{2L_1}\operatorname{cth}\tfrac{1}{2L_1}(j_1 S_1(t) + k_{12} S_2(t))\right), \\
S_2(t+1) &= L_2\left(\tfrac{L_2+\frac{1}{2}}{L_2} \cdot \operatorname{cth}\tfrac{L_2+\frac{1}{2}}{L_2}(j_2 S_2(t) + k_{21} S_1(t)) - \tfrac{1}{2L_2}\operatorname{cth}\tfrac{1}{2L_2}(j_2 S_2(t) + k_{21} S_1(t))\right).
\end{aligned} \tag{4}
$$

The resulting map (4) describes the dynamics of the average group preferences of innovators and adapters for the style method of interaction in a conflict, taking into account the parameters of intra-group and inter-group communications.

4. Empirical Results and Discussion

4.1. Estimation of Descriptive Statistics

Based on the survey results, all employees were divided into two groups based on all the characteristics studied (trust, social networks and connections, social norms and values, creativity, risk propensity, strategic approach, conflict of tasks, processes, and relationships)—innovators and adapters. Descriptive statistics for the sample as a whole, as well as for the innovators and adapters groups, are presented in Table 3.

The innovator group included 42% of the surveyed employees, or 112 people, and the adapter group included 58% (156 people). The descriptive statistical characteristics of the innovator and adapter groups show that these two groups are statistically homogeneous in almost all indicators; the variation coefficients for them do not exceed 33%. The average level of innovativeness among members of the innovator group is 1.75 versus 1.6 among adapters, while the innovator group can be characterized as more conflict-prone; the average level of conflict among members of this group is 58% higher than the average level of conflict among members of the adapter group. It should be noted that the average level of social capital among group members is comparable: 2.11 among innovators and 2.07 among adapters. In general, the innovator group can be characterized as quite creative, with members prone to risk; however, the levels of trust, social norms, and social connections do not differ significantly among these groups.

Table 3. Descriptive statistics of the studied indicators by groups.

Indicator	Total Sample (n = 268)		Group 1—Innovators (n = 112)		Group 2—Adapters (n = 156)	
	Mean	Standard Deviation	Mean	Standard Deviation	Mean	Standard Deviation
Trust	0.79	0.18	0.78	0.20	0.80	0.17
Social networks and connections	0.47	0.20	0.45	0.20	0.49	0.20
Social norms and values	0.84	0.19	0.88	0.16	0.80	0.20
Creativity	0.53	0.19	0.53	0.20	0.53	0.18
Risk propensity	0.70	0.20	0.73	0.19	0.68	0.21
Task conflict	0.52	0.21	0.49	0.21	0.53	0.21
Relationship conflict	0.45	0.22	0.52	0.21	0.39	0.22
Process conflict	0.45	0.20	0.54	0.20	0.38	0.19

4.2. Measurement Model: Confirmatory Factor Analysis, CFA

Confirmatory factor analysis (CFA) as a form of factor analysis is an approach to testing hypotheses about the relationship of items with specific factors, using structural element modeling (SEM) to test a measurement model in which factor loadings estimate the relationships between observed and latent variables. CFA is most often used to test the fit of indicators of a particular construct (latent variable) to explore the nature of that construct. The purpose of CFA is to test the fit of the data to some hypothetical measurement model.

In CFA, a hypothesis is first formed about what factors underlie the measures (constructs) used, and some constraints are imposed on the model based on these a priori hypotheses. If the constraints do not match the sample data, the results of statistical tests of model fit will indicate poor fit and the model will be rejected. If the fit is poor, this may be due to the fact that some items measure multiple factors. SEM is used to perform confirmatory factor analysis, and hypotheses are determined about the relationship between the observed indicators and their supposed primary latent factors (constructs).

The measurement model is assessed based on a series of reliability and validity tests. The measurement model for each indicator is represented by the factor loading values, as shown in Tables 4–6. The loading values for all indicators are acceptable as they are greater than 0.7. The composite reliability (CR) values are also acceptable as they are greater than 0.7. Similarly, the convergent validity is also confirmed as the average variance extracted for all latent variables (AVE) is greater than 0.5. In addition, the square root of AVE of each construct is greater than the correlations between constructs, indicating satisfactory validity, as shown in Tables 7–9.

Table 4. Reliability, internal consistency, and convergent validity for the total sample (n = 268).

Construct	Mean	Standard Deviation	Observed Variable	Factor Loading	Cronbach's Alpha	Composite Reliability (CR)	Average Variance Extracted (AVE)
SC	0.70	0.19	SC1	0.623	0.81	0.82	0.54
			SC2	0.736			
			SC3	0.756			
INN	0.53	0.20	INN1	0.821	0.85	0.85	0.68
			INN2	0.736			
			INN3	0.802			
K	0.45	0.22	K1	0.812	0.86	0.86	0.71
			K2	0.736			
			K3	0.712			

Table 5. Reliability, internal consistency, and convergent validity for a sample of adapters (n = 156).

Construct	Mean	Standard Deviation	Observed Variable	Factor Loading	Cronbach's Alpha	Composite Reliability (CR)	Average Variance Extracted (AVE)
SC	0.70	0.19	SC1	0.623	0.83	0.83	0.59
			SC2	0.736			
			SC3	0.756			
INN	0.52	0.21	INN1	0.821	0.89	0.88	0.72
			INN2	0.736			
			INN3	0.802			
K	0.57	0.19	K1	0.812	0.84	0.85	0.70
			K2	0.736			
			K3	0.712			

Table 6. Reliability, internal consistency, and convergent validity for a sample of innovators (n = 112).

Construct	Mean	Standard Deviation	Observed Variable	Factor Loading	Cronbach's Alpha	Composite Reliability (CR)	Average Variance Extracted (AVE)
SC	0.70	0.19	SC1	0.733	0.82	0.83	0.56
			SC2	0.766			
			SC3	0.792			
INN	0.53	0.19	INN1	0.801	0.89	0.89	0.72
			INN2	0.733			
			INN3	0.812			
K	0.36	0.18	K1	0.832	0.86	0.86	0.72
			K2	0.763			
			K3	0.719			

Table 7. Construct correlation matrix for the adapter group (n = 156).

Construct	SC	INN	K
SC	*0.76*		
INN	0.39 *	*0.84*	
K	0.49 *	−0.55 *	*0.83*

The italic figures on the diagonal are the square root of AVE for the constructs; * indicates significance at $p < 0.05$.

Table 8. Construct correlation matrix for the group of innovators (n = 112).

Construct	SC	INN	K
SC	*0.74*		
INN	0.49 *	*0.84*	
K	−0.39 *	−0.54 *	*0.85*

The italic figures on the diagonal are the square root of AVE for the constructs; * indicates significance at $p < 0.05$.

Table 9. Construct correlation matrix for total sample (n = 268).

Construct	SC	INN	K
SC	*0.73*		
INN	0.29 *	*0.82*	
K	0.39 *	0.31 *	*0.84*

The italic figures on the diagonal are the square root of AVE for the constructs; * indicates significance at $p < 0.05$.

The model fit to the empirical data is determined based on the calculation of the following statistical tests and indices [52]:

- The chi-squared test (χ^2) shows the difference between the observed and expected covariance matrices. Values close to zero indicate a better fit of the model to the empirical data. The disadvantage of this test is that an inappropriate model may not be rejected on small samples and that a suitable model may be rejected on large samples;
- The Goodness-of-Fit Index to the degrees of freedom (χ^2/df) should be less than 2 for high-quality models;
- The Root Mean Square Error of Approximation (RMSEA) avoids problems with sample size; RMSEA ranges from 0 to 1, where lower values indicate a better fit of the model. A value of 0.06 or less indicates an acceptable fit of the model to the empirical data;
- The Comparative Fit Index (CFI) analyzes the fit of the model by examining the discrepancy between the data and the hypothesized model, adjusting for sample size issues inherent in the chi-square criterion. CFI values range from 0 to 1, with larger values indicating better fit. CFI values of 0.95 and above are accepted as an indicator of good fit;
- The Goodness-of-Fit Index (GFI) is a measure of the fit between the hypothesized model and the observed covariance matrix. GFI values can range from 0 to 1, and for a quality model, it should be greater than 0.9.

CFA conducted on the collected data showed that the measurement models for all constructs—SC, INN, K—correspond to the hypotheses, and these constructs can be measured based on the described indicators, as shown in Table 10.

Table 10. Results of the measurement model analysis.

Model	Quality Criteria				
	χ^2	χ^2/df	RMSEA	CFI	GFI
Structural model for total sample (n = 268)	20.855	0.83	0.044	1.000	0.982
Structural model for a group of innovators (n = 112)	35.749	1.49	0.06	0.951	0.931
Structural model for a group of adapters (n = 156)	36.389	1.45	0.055	0.955	0.950
Reference value	-	<2	<0.06	>0.95	>0.9

Both for the sample of employees as a whole and for the groups of innovators and adapters, the model constructs showed high results for all quality criteria, and their calculated values correspond to the reference ones. Therefore, the proposed constructs of social capital, innovativeness, and conflict can be used to assess its causal relationships.

4.3. Structural Model: Structural Equation Modeling, SEM

Once the reliability and validity of the measurement models were established, the hypothesized causal relationships within the model were assessed using SEM. The results of the study are presented in Table 11.

Table 11. Results of the structural model analysis (direct effects), total sample, n = 268.

Hypotheses	Relationship	Anticipated Impact	Path Coefficient	t-Value	p-Value	Decision
H1	SC → INN	Positive	0.047	1.749	0.081	Not Supported
H2	SC → K	Positive	0.019	1.373	0.170	Not Supported
H3	K → INN	Positive	0.884 *	2.684	0.007	Supported

* indicates significance at $p < 0.05$.

Hypothesis H1, which assumed that there is a positive relationship between social capital and innovativeness in the overall sample of industrial enterprise employees, was not confirmed (the standardized coefficient is 0.047, $p < 0.05$). It was also found that SC has a positive, but statistically insignificant, effect on K (the standardized coefficient is 0.019), that is, hypothesis H2 is also rejected. However, in the overall sample of employees, conflict has a statistically significant effect on innovativeness, so hypothesis H3 is supported.

The results of the analysis showed that the causal relationship between SC and INN is as follows. In general, employees demonstrated a positive role of social capital in the growth of innovativeness. In conditions of conflict, this relationship generally tends to increase due to the constructive impact of conflict on innovativeness.

The indirect effect of the relationship between social capital and innovativeness through the mediation of conflict SC → K → INN can be determined as 0.019 * 0.884 = 0.017. To assess the significance of the mediation effect, we used the bootstrapping procedure and estimated this effect based on confidence intervals. We used 1000 bootstrapped samples and formed a 95% confidence interval. The results showed significant full mediation: the point value of the regression coefficient was 0.02, and the 95% confidence interval was [0.012; 0.035]. Hypothesis H4 is accepted. The results of the analysis of the mediating role of conflict in establishing the relationship between social capital and innovativeness, measured using a system of special indicators, are summarized in Table 12. The total effect, taking into account both direct and indirect influence, is 0.067 (0.02 + 0.047).

Table 12. Characterization of mediation analysis results.

Relationship	Direct Effect	Indirect Effect	Decision
SC → K → INN	0.047	0.02 *	Non-direct, only mediation

* indicates significance at $p < 0.05$.

As can be seen from Tables 3 and 4, the direct effect of SC to INN is 0.047, but it is statistically insignificant, while the indirect effect of SC to INN is 0.02 with significance at the level of $p < 0.05$. This result can be characterized as follows. The insignificant direct relationship between social capital and innovativeness is mediated by conflict between employees, increasing conflict for teams with higher levels of social capital. The mediating role of conflict in establishing the relationship between social capital and innovativeness is significant, and the insignificant relationship between social capital and innovativeness is mediated by conflict between employees. Therefore, when designing management decisions in the field of social communications of employees, increasing the level of their intra-group and institutional trust, and forming a conflict-free working environment, it is necessary to take into account the positive and significant relationship between the growth of conflict and innovativeness in a team, which can subsequently ensure the growth of innovativeness due to the positive mediating role of conflict in the formation of innovativeness.

The analysis of the relationships between social capital and innovativeness, testing hypotheses H4–H12, for groups of innovators and adapters showed the following results, as shown in Tables 13–16.

The indirect effect of the relationship between social capital and innovativeness for the sample of adapters through the mediation of the conflict SC → K → INN can be determined as 0.067 * (−0.368) = −0.025. To assess the significance of this mediation effect, we also used the bootstrapping procedure. The results showed significant full mediation: the point value of the regression coefficient was −0.026, and the 95% confidence interval was [−0.023; −0.030]. Hypothesis H8 is accepted.

Table 13. Results of structural model analysis (direct effects), adapters group, n = 156.

Hypotheses	Relationship	Anticipated Impact	Path Coefficient	t-Value	p-Value	Decision
H5	SC → INN	Positive	0.039	0.872	0.384	Not Supported
H6	SC → K	Positive	0.067 *	2.938	0.003	Supported
H7	K → INN	Negative	−0.368	−1.231	0.218	Supported (by sign)

* indicates significance at $p < 0.05$.

Table 14. Characteristics of the analysis of the mediating influence of the conflict, adapters group.

Relationship	Direct Effect	Indirect Effect	Decision
SC → K → INN	0.039	−0.026	Non-direct, only mediation

Table 15. Results of structural model analysis (direct effects), innovators group, n = 112.

Hypotheses	Relationship	Anticipated Impact	Path Coefficient	t-Value	p-Value	Decision
H9	SC → INN	Positive	0.126 *	2.226	0.027	Supported
H10	SC → K	Negative	−0.004	−0.431	0.666	Supported (by sign)
H11	K → INN	Positive	−2.774 *	−2.270	0.024	Supported (by strength)

* indicates significance at $p < 0.05$.

Table 16. Characteristics of the analysis of the mediating influence of conflict, innovators group.

Relationship	Direct Effect	Indirect Effect	Decision
SC → K → INN	0.126 *	0.013 *	Direct and mediation

* indicates significance at $p < 0.05$.

The indirect effect of the relationship between social capital and innovativeness for the sample of innovators through the mediation of conflict SC → K → INN can be determined as (−0.004) * (−2.774) = 0.011. The results of applying the bootstrapping procedure showed significant mediation: the point value of the regression coefficient was 0.013, and the 95% confidence interval was [0.009; 0.016]. Hypothesis H12 is accepted.

For the adapter group, a direct relationship between social capital and innovativeness has not been proven (it is statistically insignificant), which does not exclude the presence of significant conflict mediation. In general, social capital has a low, but positive, impact on the innovativeness of adapters. Including a conflict in the system of this relationship disrupts its direction, changing the sign of influence to negative. In such a design, the conflict forms a destructive context. With an increase in the frequency and depth of interactions between employees, conflict in the team among adapters can increase. This conclusion is consistent with previous studies.

The indirect effect of the relationship between social capital and innovativeness through conflict mediation for innovators is 0.013, and the combined effect, taking into account both direct and indirect influence, is 0.139, which indicates an increase in the innovativeness of innovative employees when conflicts arise during the performance of work processes and tasks. This demonstrates the constructive role of conflict for the creative part of employees—innovators. It should also be noted that all regression models reflecting direct dependencies of social capital, innovativeness, and conflict are statistically significant, since R2 for all of them exceeds 0.26, while the Fisher F-criterion is greater than 12.

Thus, a stable and statistically significant influence of social capital on innovativeness has been proven, both for the team as a whole and for its most creative group of innovators. For innovators, a constructive role of conflict in the growth of innovativeness of innovator employees has been established. At the same time, due to a decrease in the level of conflict with the growth of social capital, a mediating negative role of conflict in the causal relationships of social capital and innovativeness has been proven for adapters, which can contribute to a decrease in the overall level of innovativeness in conditions of conflict. A significant relationship between the growth of social capital and conflict has been shown for the team as a whole and for the group of adapters. In order to increase the innovativeness of employees, when managing teams, it is necessary to pay special attention to intra-group and inter-group interactions between innovators and adaptors, and changing the relationships between these parameters can ensure the smoothing of destructive conflicts and the moderation of constructive conflicts.

This problem of regulating the ratio of inter-group and intra-group relations of innovators and adapters for the purpose of conflict resolution is solved in the works of the author [52,53] and is briefly presented below.

4.4. Modeling Conflict Dynamics

In the next stage, in order to substantiate the values of the parameters of intra-group and inter-group connections of the studied groups, we additionally evaluate the influence of factors that make up the social capital of group members—trust, social networks and connections, and social norms and values, as well as factors that form innovativeness—creativity, risk propensity, and strategicity. The matrices of paired correlations of all these factors by groups are given in Tables 17 and 18.

Table 17. Matrix of paired correlation coefficients of social capital and innovativeness for innovators group.

Variable	Trust	Social Networks and Connections	Social Norms and Values	Social Capital	Creativity	Risk Appetite	Strategicness	Innovativeness
Trust	1.000	−0.200	−0.256	−0.013	0.051	0.004	−0.224	−0.121
Social networks and connections	−0.200	1.000	0.074	0.781 *	−0.211	−0.349 *	−0.043	0.242 *
Social norms and values	−0.256	0.074	1.000	0.608 *	−0.147	0.103	0.148	0.068
Social capital	−0.013	0.781 *	0.608 *	1.000	−0.245	−0.212	−0.010	0.346 *
Creativity	0.051	−0.211	−0.147	−0.245	1.000	0.029	−0.394	0.498 *
Risk appetite	0.004	−0.349 *	0.103	−0.212	0.029	1.000	−0.208	0.587 *
Strategicness	−0.224	−0.043	0.148	−0.010	−0.394 *	−0.208	1.000	0.275
Innovativeness	−0.121	0.242 *	0.068	0.346 *	0.498 *	0.587 *	0.275	1.000

* statistically significant parameter for $p < 0.05$.

Table 18. Matrix of paired correlation coefficients of social capital and innovativeness for adapters group.

Variable	Trust	Social Networks and Connections	Social Norms and Values	Social Capital	Creativity	Risk Appetite	Strategicness	Innovativeness
Trust	1.000	−0.072	0.006	0.397 *	−0.073	0.033	0.061	0.028
Social networks and connections	−0.072	1.000	−0.260 *	0.532 *	−0.098	−0.040	−0.104	−0.231 *
Social norms and values	0.006	−0.260 *	1.000	0.557 *	0.027	−0.172	0.158	−0.019
Social capital	0.397 *	0.532 *	0.557 *	1.000	−0.087	−0.145	0.067	0.154
Creativity	−0.073	−0.098	0.027	−0.087	1.000	−0.170	−0.203	0.416 *
Risk appetite	0.033	−0.040	−0.172	−0.145	−0.170	1.000	−0.453 *	0.492 *
Strategicness	0.061	−0.104	0.158	0.067	−0.203	−0.453 *	1.000	0.231
Innovativeness	0.028	−0.231 *	−0.019	0.154	0.416 *	0.492 *	0.231	1.000

* statistically significant parameter for $p < 0.05$.

A series of experiments were conducted analyzing a variety of possible conflict dynamics for various combinations of controllable parameters of intra- and inter-group connections between innovators and adaptors [52]. It was shown that for values of intra- and inter-group connections between innovators and adaptors at levels greater than 0.5, no change in the nature of the conflict dynamics is observed, and the pattern of conflict development remains unchanged. Therefore, we use a range of values for these parameters from 0 to 0.5. To correlate the levels of social capital indicators—trust and social norms and values that form bonding social capital—with the levels of social networks and connections that form bridging social capital, we transform their average values to the interval (0; 0.5). We also consider the sign of the correlations between the social networks and connections indicator and innovativeness.

We set the values j and k of intra- and inter-group connections for groups based on the following considerations. The values range from 0 to 0.5, with low values corresponding to weak ties and high values to strong ties. We associate the parameters of intra-group connections with the level of trust within groups and the level of social norms and values, while the parameters of inter-group connections associated with the social networks and connections indicator, as well as the sign of this indicator's correlation with the innovativeness of the corresponding group.

Tables 17 and 18 show that for innovators, the social networks and connections indicator has a positive effect on innovativeness (0.242), while for adaptors, this same correlation is negative (-0.231). The average levels of trust and social norms and values for members of both groups are similar, as shown in Table 3. The values of the normalized level of trust indicator can vary from 0 to 1, and for members of the studied groups, they range from 0.8 to 0.9. Based on these considerations and transforming these values to the interval from 0 to 0.5, we assume the same values for the intra-group cohesion (intra-group communication) parameters for adapters and innovators: $j_1 = j_2 = 0.3$.

To diagnose the inter-group communication parameters, we will make a number of introductory remarks. The inter-group communication parameter $k_{ij} > 0$ is considered positive if the impact of one group i on the opposing group j is symmetrical in response to the group's actions j. For example, if members of a group of innovators behave oppositionally and hostilely, then the adapter group adopts a copying strategy and reacts symmetrically, escalating the conflict. If members of one group behave hostilely but, in response, the opposing group attempts to smooth over the conflict, then such behavior is characterized as conciliatory and asymmetrical. In this case, the parameter $k_{ij} < 0$. Based on this and taking into account the average level of the "social networks and connections" indicator (Table 3) as well as the signs of the correlations between this indicator and innovativeness for each group of innovators and adapters (Tables 17 and 18), taking into account the transformation of the average level of the indicator into the interval from 0 to 0.5, we have $k_{12} = 0.23$, which shows that the members of the innovator group are intolerant of the criticism of the adapters; and $k_{21} = -0.2$, which means that the adapters smooth out the actions of the innovators or simply ignore their criticism.

At the next step, the conflict dynamics of groups were modeled, as shown in Figure 3a. The initial conditions were set as opposite for group members, that is, $S_1 = -2$, $S_2 = 2$. Scenario experiments were conducted with variations in the parameters of intra-group and inter-group communications, and the conditions for the convergence of group positions, that is, conflict smoothing, were shown. In the situation shown in Figure 3b, a compromise between the conflicting parties can be reached if the innovators adopt a more conciliatory position. At the same time, such a compromise cannot ensure a complete resolution of the conflict between the parties. If we strengthen the intra-group connections of innovators and

adapters, Figure 3e, and simultaneously strengthen the conciliatory power of both parties, Figure 3c, then, in this case, a compromise is not reached, and the contradictions of the parties only increase over time. The conflict is resolved only by increasing the intra-group cohesion of the innovators, Figure 3d. In this case, over time, the conflicting parties make concessions to each other, and this leads to their cooperation, which, in turn, leads to an increase in the innovativeness of group members.

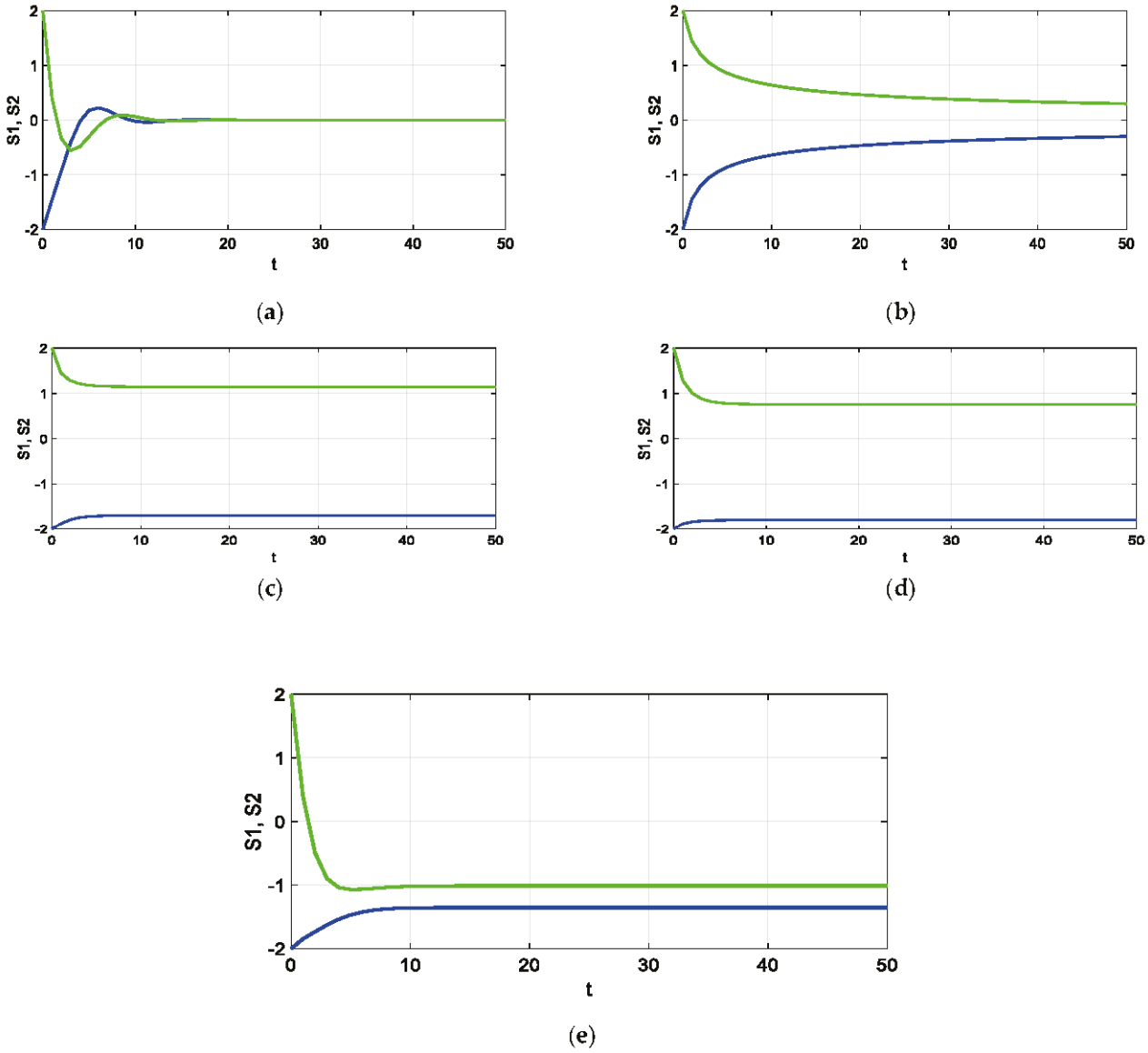


Figure 3. Dynamics of interaction system of two groups, with the following initial conditions: $S_1 = -2, S_2 = 2$: (**a**) $j_1 = 0.3, j_2 = 0.3; k_{12} = 0.23, k_{21} = -0.2$; (**b**) $j_1 = 0.3, j_2 = 0.3; k_{12} = -0.2, k_{21} = -0.2$; (**c**) $j_1 = 0.3, j_2 = 0.3; k_{12} = -0.2, k_{21} = -0.8$; (**d**) $j_1 = 0.89, j_2 = 0.3; k_{12} = -0.1, k_{21} = -0.2$; (**e**) $j_1 = 0.8, j_2 = 0.3; k_{12} = 0.23, k_{21} = -0.2$.

Thus, based on model experiments, the types of interactions between innovators and adapters are considered, and it is established that a conflict may arise between these two groups, and the possible consequences of its development are demonstrated. It is shown that in various situations, differing in the strength and direction of intra-group and inter-group communications, various scenarios of conflict development are realized.

In real situations, the conflict dynamics of groups can be characterized as convergence (cooperation), polarization (competition), or oscillatory mode (unstable mode).

Specific mechanisms for human capital development and its justification in the form of factors of cause-and-effect relationships are developed in the works [53,54].

5. Conclusions

This paper has constructively solved the problem of modeling and managing the personal innovativeness of employees in project teams in research and development business processes. We have investigated the innovativeness of the subject as a resource as the ability to think innovatively, to produce new ideas, and to apply them in practice. We have assumed that social capital can be a hidden resource of social networks and connections for obtaining information that will help to enhance the innovativeness of the subject. At the same time, we take into account the mediating role of conflict in the process of such a transformation of social capital into innovativeness.

Based on the SEM methods, the constructs of social capital, innovativeness, and conflict, which depend on a number of observable characteristics of employees, have been proven. We investigated three constructs—personal innovativeness, social capital of the individual, and conflict in the project team—in research and development business processes. Personal innovativeness as a construct includes indicators of creativity, risk propensity, and strategicity. The social capital construct includes such observable indicators as trust, social networks and connections, and norms and values. The conflict construct consists of observable indicators of conflict of tasks, processes, and relationships. The study was conducted using data from a survey of scientific organization employees. For a more detailed analysis, the employees were divided into groups of innovators and adapters, and all model constructs were tested for the sample of employees as a whole, as well as for subsamples of innovators and adapters. A stable and statistically significant influence of social capital on innovativeness has been proven, both for the team as a whole and for its most creative group of innovators.

The mediating role of conflict arising in project groups during innovation was analyzed. It was shown that, unlike the group of innovators, for the group of adapters, neither a direct nor indirect effect of the influence of social capital and innovativeness was observed when mediating conflict. For innovators, a constructive role of conflict in the growth of innovativeness of innovative employees was established. At the same time, due to the decrease in the level of conflict with the growth of social capital, the mediating destructive role of conflict in the causal relationships of social capital and innovativeness has been proven for innovators, which can contribute to a decrease in the overall level of innovativeness in a conflict situation. For the team as a whole and for the group of adapters, a significant relationship between the growth of social capital and conflict has been shown.

In order to increase the innovativeness of employees, it is necessary to pay special attention to intra-group and inter-group interactions between innovators and adapters, and changing the relationships between these parameters can ensure the smoothing of destructive conflicts and the moderation of constructive conflicts. This problem is solved on the basis of the sociodynamics model, which allows us to substantiate the values of the controlled parameters of intra-group and inter-group interactions of project groups, ensuring the achievement of equilibrium in the form of a strategy for cooperation between the parties to the conflict. In practice, the results of such studies can be applied to promote innovation in project groups and differentiated by areas that are characterized by different strengths, focuses, and combinations of intra- and inter-group connections of group members.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to data privacy.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Midgley, C.; Kaplan, A.; Middleton, M.; Maehr, M.L.; Urdan, T.; Anderman, L.H. The development and validation of scales assessing students' achievement goal orientations. *Contemp. Educ. Psychol.* **1998**, *23*, 113–131. [CrossRef]
2. Linnenbrink, E.A.; Pintrich, P.R. Motivation as an enabler for academic success. *Sch. Psychol. Rev.* **2002**, *31*, 313–327. Available online: https://psycnet.apa.org/record/2002-18945-003 (accessed on 1 August 2025). [CrossRef]
3. Patterson, F.; Kerrin, M.; Gatto-Roissard, G. *Characteristics & Behaviours of Innovative People in Organisations*; Literature Review Prepared for the NESTA Policy & Research Unit; NESTA: London, UK, 2009.
4. Anderson, N.; Potocnik, K.; Zhou, J. Innovation and creativity in organizations. *J. Manag.* **2014**, *40*, 1297–1333. [CrossRef]
5. Wisdom, J.P.; Chor, K.H.B.; Hoagwood, K.E.; Horwitz, S.M. Innovation adoption: A review of theories and constructs. *Adm. Policy Ment. Health Ment. Health Serv. Res.* **2014**, *41*, 480–502. [CrossRef]
6. Cai, Y. From an analytical framework for understanding the innovation process in higher education to an emerging research field of innovations in higher education. *Rev. High. Educ.* **2014**, *40*, 585–616. [CrossRef]
7. Aldahdouh, T.; Korhonen, V.; Nokelainen, P. What contributes to the individual innovativeness? A multilevel perspective. *Int. J. Innov. Stud.* **2019**, *3*, 23–39. [CrossRef]
8. Orlova, E.V. Methodology and Statistical Modeling of Social Capital Influence on Employees' Individual Innovativeness in a Company. *Mathematics* **2022**, *10*, 1809. [CrossRef]
9. Orlova, E.V. Design Technology and AI-Based Decision Making Model for Digital Twin Engineering. *Future Internet* **2022**, *14*, 248. [CrossRef]
10. Orlova, E.V. Design of Personal Trajectories for Employees' Professional Development in the Knowledge Society under Industry 5.0. *Soc. Sci.* **2021**, *10*, 427. [CrossRef]
11. Christensen, K.S. Losing innovativeness: The challenge of being acquired. *Manag. Decis.* **2006**, *44*, 1161–1182. [CrossRef]
12. Rodan, S. Innovation and heterogeneous knowledge in managerial contact networks. *J. Knowl. Manag.* **2002**, *6*, 2. [CrossRef]
13. Belkin, V.N.; Belkina, N.A.; Antonova, O.A.; Bochkaeva, I.V. Theory of social capital of an organization. *Soc. Power* **2012**, *36*, 4.
14. Arieti, S. *Creativity: The Magic Synthesis*; Basic Books: New York, NY, USA, 1976.
15. Bykov, A.V.; Nastina, E.A. Relationships between value attitudes and career achievements (according to youth research). *Sociol. Res.* **2020**, *8*, 67–77. [CrossRef]
16. Cukier, W.; Hodson, J.; Omar, A. *Soft' Skills Are Hard: A Review of the Literature*; Ryerson University: Toronto, Canada, 2015; Available online: https://www.ryerson.ca/content/dam/diversity/reports/KSG2015_SoftSkills_FullReport.pdf (accessed on 1 August 2025).
17. Hippel, E. Personality Traits of Successful Free Innovators. In *Free Innovation*; MIT Press: Cambridge, MA, USA, 2017. [CrossRef]
18. Putnam, R. The prosperous Community. Social capital and Public Life. *Am. Prospect.* **1993**, *4*, 13.
19. Tsai, W.; Ghoshal, S. Social capital and value creation: The role of intrafirm networks. *Acad. Manag. J.* **1998**, *41*, 464–476. [CrossRef]
20. Dakhli, M.; De Clercq, D. *Human Capital, Social Capital and Innovation: A Multi-Country Study*; Vlerick Leuven Gent Management School Working Paper Series; Vlerick Leuven Gent Management School: Brussels, Belgium, 2003; p. 16. [CrossRef]
21. Landry, R.; Amara, N.; Lamari, M. Does social capital determine innovation? To what extent? *Technol. Forecast. Soc. Change* **2002**, *69*, 681–701. [CrossRef]
22. Beugelsdijk, S.; Van Schaik, T. Differences in social capital between 54 Western European regions. *Reg. Stud.* **2005**, *39*, 1053–1064. [CrossRef]
23. Murphy, L.J. Social Capital and Innovation: A Theoretical Perspective. In *Innovation and Social Capital in Organizational Ecosystems*; Brychan, C.T., Lyndon, J.M., Eds.; IGI Global: London, UK, 2019; pp. 1–22. [CrossRef]
24. Coleman, J.S. Social capital in the creation of human capital. *Am. J. Sociol.* **1988**, *94*, 94–120. [CrossRef]
25. Yip, W.; Subramanian, S.V.; Mitchell, A.D.; Lee, D.T.; Wang, J.; Kawachi, I. Does social capital enhance health and well-being? Evidence from rural China. *Soc. Sci. Med.* **2007**, *64*, 35–49. [CrossRef]
26. Siegler, V. *Measuring Social Capital*; Office for National Statistics: London, UK, 2014.
27. Styhre, A.; Borjesson, S. Innovativeness and Creativity in Bureaucratic Organizations: Evidence from the Pharmaceutical and the Automotive Industry. In Proceedings of the OLKC 2006 Conference at the University of Warwick, Warwick, UK, 20–22 March 2006.

28. Goleman, D.; Boyatzis, R. Social intelligence and the biology of leadership. *Harv. Bus. Rev.* **2008**, *86*, 74–81. Available online: https://hbr.org/2008/09/social-intelligence-and-the-biology-of-leadership (accessed on 1 August 2025). [PubMed]
29. Gkonou, C.; Mercer, S. *Understanding Emotional and Social Intelligence Among English Language Teachers*; British Council: London, UK, 2017.
30. Lebedeva, N.M.; Bushina, E.V.; Cherkasova, L.L. Values, social capital and attitude towards innovations. *Soc. Sci. Mod.* **2013**, *4*, 28–41.
31. Lee, J. Heterogeneity, brokerage, and innovative performance: Endogenous formation of collaborative inventor networks. *Organ. Sci.* **2010**, *21*, 804–822. [CrossRef]
32. Han, J. Team-bonding and team-bridging social capital: Conceptualization and implications. *Team Perform. Manag. Int. J.* **2018**, *24*, 17–42. [CrossRef]
33. Dokko, G.; Kane, A.A.; Tortoriello, M. One of us or one of my friends: How social identity and tie strength shape the creative generativity of boundary-spanning ties. *Organ. Stud.* **2014**, *35*, 703–726. [CrossRef]
34. Carnabuci, G.; Diószegi, B. Social networks, cognitive style, and innovative performance: A contingency perspective. *Acad. Manag. J.* **2015**, *58*, 881–905. [CrossRef]
35. Voronin, A.N. Intellectual activity: Manifestation of intelligence and creativity in real interaction. *Psychol. HSE J.* **2006**, *3*, 35–58.
36. Bourdieu, P. Forms of social capital. In *Handbook of Theory and Research for the Sociology of Education*; Richards, J.C., Ed.; Greenwood Press: New York, NY, USA, 1983.
37. Olson, M. *The Rise and Decline of Nations*; Yale University Press: New Haven, CT, USA, 1982.
38. Fukuyama, F. *Trust: The Social Virtues and the Creation of Prosperity*; Free Press: New York, NY, USA, 1995.
39. Ponzetto, G.A.M.; Troiano, U. Social Capital, Government Expenditures, and Growth. *J. Eur. Econ. Assoc.* **2018**, *23*, 632–681. [CrossRef]
40. Ma, L.; Zhang, X.; Wang, G.; Zhang, G. How to build employees' relationship capital through different enterprise social media platform use: The moderating role of innovation culture. *Internet Res.* **2021**, *31*, 1823–1848. [CrossRef]
41. Lizardo, O. Improving Cultural Analysis: Considering Personal Culture in its Declarative and Nondeclarative Modes. *Am. Sociol. Rev.* **2017**, *82*, 88–115. [CrossRef]
42. Barton, H.; Thorpe, J.; Dufur, M. Social Capital and Prosocial Behavior among German Children. *Soc. Sci.* **2020**, *9*, 215. [CrossRef]
43. Chursin, A.A.; Yudin, A.V.; Grosheva, P.Y. Intellectual Capital of Technological Transformation of Economy of Industrial Region. *J. Appl. Econ. Res.* **2021**, *20*, 641–665. [CrossRef]
44. De Dreu, C. When too little or too much hurts: Evidence for a curvilinear relationship between task conflict and innovation in teams. *J. Manag.* **2006**, *32*, 83–107. [CrossRef]
45. Jehn, K.; Greer, L.; Levine, S. The effects of conflict types, dimensions, and emergent states on group outcomes. *Group Decis. Negot.* **2008**, *17*, 465–495. [CrossRef]
46. O'Neill, T.A.; McLarnon, M.J.W.; Hoffart, G. The multilevel nomological net of team conflict profiles. *Int. J. Confl. Manag.* **2018**, *29*, 24–46. [CrossRef]
47. Behfar, K.J.; Mannix, E.A.; Peterson, R.S. Conflict in small groups: The meaning and consequences of process conflict. *Small Group Res.* **2011**, *42*, 127–176. [CrossRef]
48. Kirton, M.J. Adaptors and Innovators: A Description and Measure. In *Creativity Assessment: Readings and Resources*; Puccio, G.J., Murdock, M.C., Eds.; Creative Education Foundation Press: Buffalo, NY, USA, 1999; 197p.
49. Stollberger, J.; Ali Al-Atwi, A.; De Cremer, D. Untangling the team social capital–team innovation link: The role of proportional task conflict as well as group- and differentiated individual-focused transformational leadership. *Hum. Relations* **2022**, *76*, 871–900. [CrossRef]
50. Likert, R. *A Technique for the Measurement of Attitudes*; The Science Press: New York, NY, USA, 1932.
51. Orlova, E.V. A Novel Brillouin and Langevin Functions Dynamic Model for Two Conflicting Social Groups: Study of R&D Processes. *Mathematics* **2024**, *12*, 2788. [CrossRef]
52. Kline, R.B. *Principles and Practice of Structural Equation Modeling*, 3rd ed.; Guilford Press: New York, NY, USA, 2010.
53. Orlova, E.V. Dynamic Regimes for Corporate Human Capital Development Used Reinforcement Learning Methods. *Mathematics* **2023**, *11*, 3916. [CrossRef]
54. Orlova, E.V. Inference of Factors for Labor Productivity Growth Used Randomized Experiment and Statistical Causality. *Mathematics* **2023**, *11*, 863. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Interpretable Machine Learning Models for Estimating Electric Energy Consumption in Steel Industries

Paulino José García-Nieto [1,*], Esperanza García-Gonzalo [1], Luis Alfonso Menéndez-García [1], Laura Álvarez-de-Prado [2], Marta Menéndez-Fernández [2] and Antonio Bernardo-Sánchez [2]

[1] Department of Mathematics, Faculty of Sciences, University of Oviedo, C/Leopoldo Calvo Sotelo, 18, 33007 Oviedo, Spain; espe@uniovi.es (E.G.-G.); menendezalfonso@uniovi.es (L.A.M.-G.)
[2] Department of Mining Technology, Topography and Structures, Higher and Technical School of Mining Engineering, University of León, Campus de Vegazana s/n, 24071 León, Spain; laura.alvarez@unileon.es (L.Á.-d.-P.); mmenf@unileon.es (M.M.-F.); antonio.bernardo@unileon.es (A.B.-S.)
* Correspondence: pjgarcia@uniovi.es; Tel.: +34-985-103-417

Abstract: The substantial energy consumption and associated CO_2 emissions from industrial operations pose significant environmental and economic challenges for factories and surrounding communities. Within the context of industrial energy management, the steel industry represents a major energy consumer. The imperative to optimize energy use in this sector is driven by a combination of environmental concerns, economic incentives, and technological advancements. This study presents a machine learning model that integrates the whale optimization algorithm (WOA) with multivariate adaptive regression splines (MARS) to forecast electric energy consumption. Utilizing a dataset comprising 35,040 real-world energy consumption records from Gwangyang Steelworks in South Korea, the model was benchmarked against other regression techniques (ridge, lasso, and elastic-net), demonstrating that the proposed WOA-MARS approach achieves a significant improvement in the RMSE (vs. elastic-net or lasso regression techniques) while maintaining interpretability through hinge function analysis. The WOA-tuned MARS model achieves a coefficient of determination (R^2) of 0.9972, underscoring its effectiveness for energy optimization in steel manufacturing. The key findings reveal that CO_2 emissions and reactive power variables are the strongest predictors.

Keywords: multivariate adaptive regression splines (MARS); whale optimization algorithm (WOA); ridge (RR), lasso (RL), and elastic-net (ENR) regressions; steelworks electric energy consumption

MSC: 62G08; 92B20; 68T20

1. Introduction

As industrialization advances, energy demand correspondingly increases, necessitating the continual adaptation of national policies. Economic growth further intensifies energy consumption, rendering energy both an essential and finite resource. Promoting energy conservation benefits consumers, producers, and overall sustainability, while safeguarding long-term efficiency. The acceleration of economic expansion amplifies the demand for electricity, a fundamental component of daily life and a critical driver of economic development [1]. Environmental challenges stem from the steel industry's substantial energy consumption and associated CO_2 emissions [2]. Carbon dioxide, the principal

greenhouse gas, is predominantly released during blast furnace and coking operations, which together account for approximately 70% of total emissions [3,4]. On a global scale, steel production is responsible for nearly 7% of anthropogenic CO_2 emissions [5]. To mitigate climate change, advanced technologies are being developed to quantify and reduce energy consumption and CO_2 emissions in steel production. Current research focuses on emission-reduction strategies such as carbon capture and the adoption of green hydrogen technologies [6–8]. By 2030, these innovative approaches are projected to decrease CO_2 emissions by approximately 14–21% and energy consumption by 7–11% [9]. Consequently, forecasting energy demand has become a critical priority for both industrial sectors and national policymakers [10]. Recent research has increasingly employed data-mining techniques to predict energy consumption, utilizing methods such as gradient-boosted machines [11], neural networks [12], time-series models [13], support vector machines [14], regression analysis [15], and deep learning architectures [16]. One study evaluated predictive models, including generalized linear models (GLMs), support vector regression (SVR), k-nearest neighbors (k-NN), random forest, and M5 model trees within a smart city framework [17], although further enhancements in predictive accuracy remain necessary.

In contrast with time-series forecasting methods, which predict future energy demand using lagged variables, this study focuses on estimating instantaneous energy consumption as a function of concurrent operational parameters. The objective is to model the nonlinear relationships between real-time measurements (e.g., reactive power, CO_2 emissions, and load type) and energy usage, providing interpretable insights for operational optimization rather than temporal prediction.

This study accurately predicts the electric energy consumption of Gwangyang Steelworks (see Figure 1) using a multivariate adaptive regression splines (MARS) model [18–25], optimized through the whale optimization algorithm (WOA) [26–29]. To the best of our current knowledge, this specific combination has not yet been explored in the existing literature. For comparative analysis, the same dataset is also evaluated using ridge regression (RR), lasso regression (LR), and elastic-net regression (ENR) techniques [20–22,30–35]. MARS is a statistical learning approach capable of capturing nonlinearities and variable interactions, evolving from linear models to effectively represent complex relationships [18–28,30,31]. It offers several advantages over both traditional and metaheuristic regression techniques [18–25,30,31,36,37]: (i) straightforward implementation compared with linear regression, (ii) consistent basis functions derived from identical datasets, (iii) ease of interpretability, (iv) ability to handle both continuous and categorical variables, and (v) an explicit mathematical formulation of the dependent variable through fundamental basis functions. This latter characteristic distinguishes MARS from black-box machine learning models such as multilayer perceptrons. Previous studies have demonstrated the effectiveness of MARS in predicting diverse phenomena, including antitumor activity [38], soil organic carbon [39], regolith geochemical grades [40], drilling mud viscosity [41], used car prices [42], benzene concentration [43], and dissolved oxygen levels [44]. However, to date, it has not been applied to predicting the electric energy consumption of Gwangyang Steelworks.

The structure of this manuscript is organized as follows: It begins with a detailed description of the experimental design, followed by an outline of the parameters utilized in this investigation. Section 2 provides a comprehensive review of the regression modeling approaches employed, including lasso, ridge, elastic-net, and WOA/MARS-based methods. Section 3 presents the results of the WOA/MARS analysis, comparing predicted values with experimental measurements and evaluating the relative importance of the input

parameters. Finally, Section 4 summarizes this study's conclusions and highlights its principal findings.



(**a**) (**b**)

Figure 1. (**a**) Aerial view of Gwangyang Steelworks; (**b**) side view of Gwangyang Steelworks.

2. Materials and Methods

2.1. Experimental Setup

The dataset comprises synchronous measurements of nine independent variables and the target variable (energy consumption (EC)), all recorded at 15 min intervals. The dataset for this study comprises 35,040 samples of electric energy consumption and the following process variables: carbon dioxide emissions (tCO_2), lagging current reactive power (LaCRP), lagging current power factor (LaCPF), leading current reactive power (LeCRP), leading current power factor (LeCPF), day of the week (DoW), week status (WS), seconds since midnight (NSM), and load type (LT). The dataset was obtained from the Gwangyang Steelworks, a major industrial facility specializing in the production of steel coils, iron plates, and related steel products [45]. The records span a full year, capturing detailed measurements of key operational variables. The dependent variable in this study is the electric energy consumption (EC), measured in kilowatt-hours (kWh) per 15 min interval. This metric represents the electrical energy used by individual production units or processes within the Gwangyang Steelworks during each sampling period. The 15 min interval aligns with the plant's data-logging system, capturing high-resolution variations in energy demand associated with operational phases such as smelting, casting, and material handling [46]. The importance of these data comes from their ability to track energy consumption in detail and in real time, providing valuable insights into daily, monthly, and yearly usage patterns [17]. This study does not frame the problem as a time-series forecasting task (e.g., predicting future EC from past values). Instead, the problem is approached as a multivariate regression task, in which EC is estimated from variables measured at the same timestamp. This approach is justified by the industrial need to understand how current operational states (e.g., furnace load and power factor) directly impact energy consumption, without assuming temporal dependencies. Figure 2 shows the experimental setup of the process.

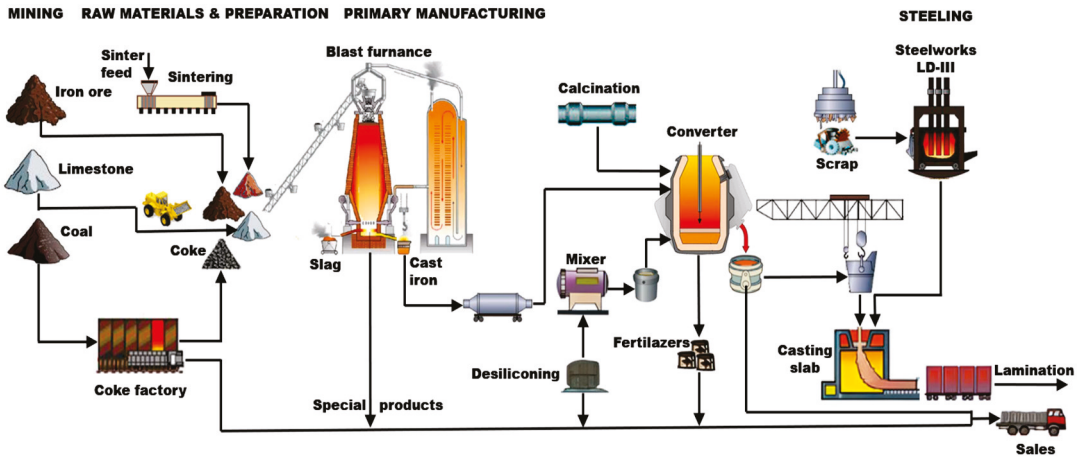


Figure 2. Schematic diagram illustrating the industrial experimental procedure: experimental setup.

2.2. Variables in Model and Materials

The primary objective of this study was to identify the most influential parameters for estimating energy consumption using the WOA/MARS model. Energy consumption, representing the electrical energy utilized in steel production from iron ore, was designated as the output variable. A total of nine input variables were considered, as summarized in Table 1:

Table 1. List of the operational variables analyzed in this investigation, gathered with the corresponding standard deviations (STDs) and averages.

Input Variables	Parameter Name	Mean	Standard Deviation	Minimum Value	Maximum Value
Lagging current reactive power (kVarh)	LaCRP	13.035	16.306	0	96.91
Leading current reactive power (kVarh)	LeCRP	3.871	7.424	0	27.76
Carbon dioxide emissions (ton CO_2/15 min)	tCO_2	0.012	0.016	0	0.07
Lagging current power factor (%)	LaCPF	80.578	18.921	0	100
Leading current power factor (%)	LeCPF	84.368	30.457	0	100
Number of seconds from midnight for each day	NSM	42,750	24,940.534	0	85,500
Week status	WS	-	-	-	-
Day of week	DoW	-	-	-	-
Load type	LT	-	-	-	-
Output Variable					
Steelworks' energy consumption (kWh)	EC	27.387	33.444	0	157.18

Lagging current reactive power (kVarh): Indicates an inductive nature with reactive power consumption when the load current peaks at 90° after the voltage peak;

Leading current reactive power (kVarh): Occurs when the current waveform leads the voltage waveform, indicating capacitive loads supplying reactive power;

Carbon dioxide emission (ton/15 min): Emissions from combustion in steel plants and other industrial processes;

Lagging current power factor (%): Reflects the load current peaking after the voltage, with capacitive loads used to compensate;

Leading current power factor (%): Indicates a capacitive load supplying reactive power when current leads the voltage;

Number of seconds from midnight for each day: Time duration from midnight to the specified datetime value;

Week status: Categorical variable indicating whether the day is a "weekend" or "weekday" [47];

Day of week: Categorical variable indicating the day of the week for casting operations;

Load type: Categorical variable with three types of furnace load: "light load," "medium load," and "maximum load".

The operation output variable of this study was as follows:

The Gwangyang Steelworks' specific electric energy consumption (kWh): The Gwangyang plant is categorized as a high-energy-consumption facility due to the substantial amount of electrical energy required for steel production.

Exploratory data analysis represents a fundamental step in deriving meaningful insights from a dataset. This analysis confirmed that all dataset features were complete, with no missing values. In addition, the interrelationships among the variables were systematically examined. A correlation matrix was employed to visualize the associations between the different variables, and the corresponding heatmap illustrates the strength of the linear relationships among them. Figure 3 depicts the correlations of various variables with respect to energy consumption. The strongest correlation was observed between CO_2 emissions and lagging current reactive power, reflecting their direct relationship with energy consumption.

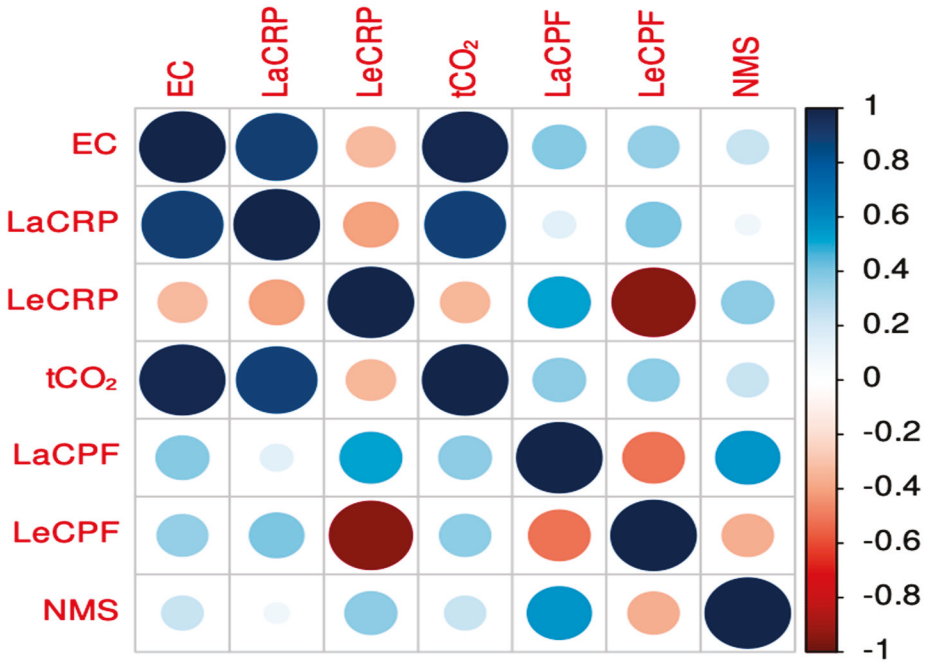


Figure 3. Correlation matrix for the numerical variables.

2.3. Techniques for Mathematical Modeling

2.3.1. Multivariate Adaptive Regression Splines (MARS) Method

The MARS approach provides a versatile non-parametric approach to regression problems [18–25,30,31]. This technique can be considered as extending both stepwise linear (SL)

regression and classification and regression tree (CART) decision trees [30,48,49], while addressing their respective limitations. The primary objective of MARS is to foretell a continuous dependent output variable's results $\boldsymbol{y}(n \times 1)$, relying on a multitude of independent input features, collectively referred to as $\boldsymbol{X}(n \times p)$. The basic principles underlying the MARS methodology can be explained by a specific mathematical formulation [18–25,30,31]:

$$\boldsymbol{y} = f(\boldsymbol{X}) + \boldsymbol{e} \tag{1}$$

where

f: This component represents an aggregation of basis functions, each of which depends on the input variable set $\boldsymbol{X}$. These functions are combined by a weighted summation process.

$\boldsymbol{e}$: This term denotes the error vector, which is characterized by its n-dimensional nature, where n corresponds to the number of observations in the dataset. It is typically represented as a column vector with dimensions $(n \times 1)$.

The MARS approach does not necessitate a predetermined functional relationship between the output and input features. Mathematically, it can be articulated as an assemblage of segmented polynomial expressions of order q, referred to as basis functions, with their parameters entirely determined through a detailed analytical process involving $(\boldsymbol{X}, \boldsymbol{y})$.

The methodology of MARS is implemented through the adjustment of specific ranges of the predictor variables using these basis functions. In particular, two-sided truncated power functions, or splines, are the fundamental building blocks of the MARS model. These functions, called hinge functions, can be expressed mathematically as follows [18–25,30,31]:

$$\begin{aligned} [-(x-t)]_+^q &= \begin{cases} (t-x)^q & \text{if} \quad x < t \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \\ [+(x-t)]_+^q &= \begin{cases} (t-x)^q & \text{if} \quad x \geq t \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \end{aligned} \tag{2}$$

The hinge function knots in the MARS model are data-derived thresholds optimized to minimize prediction error. These values reflect empirical breakpoints in the relationship between predictors and energy consumption, rather than physical constants.

The power q (where $q(\geq 0)$) determines how smooth the resulting approximated function is and, at the same time, also determines the category of splines employed. Specifically, when $q = 1$, the model uses linear splines, which are the focus of the present investigation. Quadratic splines are generated when $q = 2$, while cubic splines are generated when $q = 3$, with this pattern continuing for higher values of q. To illustrate this concept, Figure 4 provides a visual representation of two splines in the case where $q = 1$, with the node (also referred to as the knot) positioned at $t = 3.5$. The purpose of this graphical representation is to demonstrate the behavior of linear splines at a particular node location. The choice of q has significant implications for the flexibility of the model and its capacity to identify nonlinear patterns in the data. Linear splines $(q = 1)$ offer a balance between model simplicity and the ability to approximate nonlinear functions, making them a popular choice in many applications. Higher-order splines $(q > 1)$ can provide greater smoothness but can also introduce additional complexity into the model. The MARS algorithm's capability to automatically select appropriate basis functions and node locations contributes to its versatility in different areas of scientific investigation.

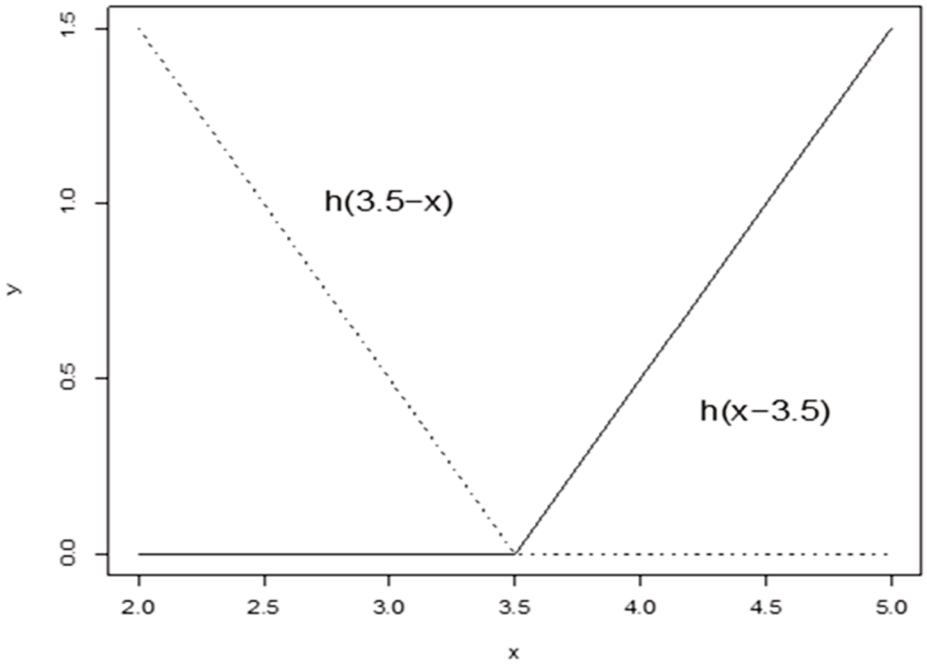


Figure 4. Graphical representation of linear hinge functions. This figure depicts two linear hinge functions, foundational components of the MARS methodology, commonly referred to as spline foundational components. The diagram includes the following: left spline (dashed line), which represents (x < t, −(x − t)), illustrating behavior for x-values of less than the knot location t; right spline (solid line), which describes (x > t, +(x − t)), characterizing behavior for x-values exceeding t. This visualization highlights the functional symmetry about the knot location t.

Unlike regularized regression methods, the implementation of MARS does not require predictor scaling or centering [50]. The algorithm's hinge functions adapt to the natural scale of each variable, while the forward/backward selection process ensures parsimony by excluding irrelevant predictors (e.g., unused categorical levels).

The MARS approximation can be characterized as an ensemble of M piecewise linear multivariate splines of a dependent variable derived from M basis functions, built using the hinge functions. The mathematical formulation is as follows [18,19,21–25,30,31]:

$$\hat{\boldsymbol{y}} = \hat{f}_M(\boldsymbol{x}) = c_0 + \sum_{m=1}^{M} c_m B_m(\boldsymbol{x}) \tag{3}$$

where the MARS method's predicted output variable is $\hat{\boldsymbol{y}}$ and

c_0 represents a constant coefficient, commonly referred to as the intercept;
$B_m(\boldsymbol{x})$ signifies the m-th basis function;
c_m corresponds to the coefficient related to the basis function $B_m(\boldsymbol{x})$.

To obtain the optimal collection of basis functions for the comprehensive method, the MARS methodology uses the technique of generalized cross-validation (GCV) [18–25,30,31]. GCV is calculated as the mean-squared residual error divided by a penalty factor, the latter being directly proportional to model complexity, and can be expressed mathematically as [18,19,21,23–25,30,31,50,51]

$$GCV(M) = \frac{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}_M(\boldsymbol{x}_i)\right)^2}{\left(1 - C(M)/n\right)^2} \tag{4}$$

The more basis functions that are employed in the MARS model, the higher the complexity penalizing term, $C(M)$. This term is expressed as [18–25,30,31,50,51]

$$C(M) = (M + 1) + dM \tag{5}$$

where M is the number of model basis functions in Equation (3), and d is the model penalty parameter for each basis function.

With the purpose of ensuring the robustness of the findings, it is essential to consider additional factors alongside the aforementioned GCV. These include the following [18,19,21–25,30,31,50,51]:

RSS (residual sum of squares): This measure quantifies the difference between the observations and predictions;

N-subsets: The factor that specifies the total count of groups within the model encompassing all variables.

2.3.2. Ridge Regression and the Least Absolute Shrinkage and Selection Operator (Lasso) Regression Methods

Both regression procedures represent regularization techniques employed in this study to predict the energy consumption (EC) of Gwangyang Steelworks [20–22,30,31]. These methods are particularly effective in addressing multicollinearity, which frequently arises in linear regression models with numerous predictors. Overall, they introduce a controlled level of bias in exchange for enhanced efficiency and stability in parameter estimation.

Ridge regression (RR) was developed as a potential solution to address inaccuracies in least squares estimators arising from multicollinearity among independent variables in linear regression models. By providing variance and mean square estimates that are often lower than those obtained via ordinary least squares, RR enables more precise estimation of model parameters. Consequently, it is also employed to predict the energy consumption (EC) of Gwangyang Steelworks [20–22,30,31,52,53].

By adding positive components, the problem of a quasi-singular moment matrix $\boldsymbol{X}^T\boldsymbol{X}$ is mitigated by lowering the condition number of the diagonals. The following formula for the ordinary least squares estimator can also be used to find the simple ridge estimator [20–22,30,31,52,53]:

$$\hat{\beta}_{RR} = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y} \tag{6}$$

When $\boldsymbol{y}$ is the regressand, $\boldsymbol{I}$ represents an identity matrix, with $\boldsymbol{X}$ as the design matrix, and the constant that moves the diagonals of the moment matrix is called the ridge parameter $\lambda \geq 0$, sometimes dubbed as the regularization or complexity parameter. It can be demonstrated that this estimator may be used to solve the following least squares problem with the restriction $\|\beta\| \leq t$ (a predetermined free parameter t determines the degree of regularization), whereby a Lagrangian can be used to represent it [25–27,31,32], as follows:

$$L^{RR} = \min_{\beta}(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta) + \lambda\|\beta\|_2^2 \text{subject to} \|\beta\|_2^2 \leq t \tag{7}$$

where $\|\beta\|_2 = \left(\sum_{i=1}^{N}\beta_i^2\right)^{1/2}$. This proves that λ is just the Lagrange multiplier of the restriction. The ridge estimator turns into ordinary least squares when $\lambda = 0$, demonstrating that the restriction is not enforceable. Typically, λ is chosen heuristically, which means that it will not exactly meet the requirement. Therefore, the principal advantage of ridge regression over standard linear regression lies in its capacity to balance the trade-off between bias

and variance. Specifically, an increase in bias is accompanied by a corresponding reduction in variance, and vice versa.

The lasso regression (LR) method, also applied in this study to address the aforementioned issue, shares several key similarities with ridge regression (RR). The loss function of lasso regression is formally defined as follows [25–27,31,32,54,55]:

$$L^{LR} = \min_{\beta}(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta) + \lambda\|\beta\|_1 \text{subject to} \|\beta\|_1 \leq t \tag{8}$$

so that $\|\beta\|_1 = \sum_{i=1}^{N}|\beta_i|$ and where the data determines the exact relationship between t and λ.

To sum up, LR can reduce coefficients to zero, whereas RR cannot [20–22,30,31,52,53]. This distinction arises from the difference in the shapes of their constraint boundaries. Although they are subject to different constraints—$\|\beta\|_1 \leq t$ for lasso regression and $\|\beta\|_2^2 \leq t$ for ridge regression—both ridge and lasso regression can be interpreted as minimizing a common objective function.

2.3.3. Elastic-Net Regression (ENR)

Elastic-net regression (ENR) combines the principles of both lasso and ridge regressions. The elastic-net Lagrangian function is formulated by integrating the penalty terms of these two methods [20–22,30,31,52,53]:

$$L^{EN} = \min_{\beta}(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta) + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 \tag{9}$$

In fact, lasso regression is obtained if $\lambda_2 = 0$, while ridge regression is obtained if $\lambda_1 = 0$. The pair of parameters λ can be substituted with the $L_1 - ratio$ parameter, which determines the ratio of the L_1 penalty to λ. Since most computational codes use the number $L_1 - ratio = 0.5$ the most, it is used in the calculations here. In this regard, the elastic-net regression's final Lagrangian is provided by [21,22,31,32,34,35]

$$L^{EN} = \min_{\beta}(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta) + \lambda(1 - L_{ratio})\|\beta\|_1 + \lambda L_{ratio}\|\beta\|_2^2 \tag{10}$$

2.3.4. Whale Optimization Algorithm (WOA)

The Whale Optimization Algorithm (WOA) was originally proposed by Mirjalili and Lewis [28]. It is an optimization technique inspired by the complex hunting behavior of humpback whales. In a process known as bubble-net feeding, whales generate bubbles to encircle their prey during the hunt. Typically, the whales dive approximately 12 m before creating a spiral of bubbles around the target and subsequently ascend in pursuit. The WOA operates through three primary phases [26–29]:

Exploration phase: During this phase, each search agent (whale) explores the search space randomly to locate the optimal solution (prey). To update the position of a given search agent, a randomly selected agent is used instead of the current best-performing agent. Mathematically, this behavior is expressed as follows:

$$\begin{aligned} \boldsymbol{D} &= |\boldsymbol{C} \cdot \boldsymbol{X}_{rand}(t) - \boldsymbol{X}(t)|, \\ \boldsymbol{X}(t+1) &= \boldsymbol{X}_{rand}(t) - \boldsymbol{A} \cdot \boldsymbol{D}, \end{aligned} \tag{11}$$

where $\boldsymbol{X}_{rand}$ is the position vector of a randomly selected prey, t denotes the actual iteration, and $\vec{X}$ indicates the whale position vector The coefficient vectors are defined as $\boldsymbol{A} = 2\boldsymbol{a} \cdot \boldsymbol{r}_1 - \boldsymbol{a}$ and $\boldsymbol{C} = 2\boldsymbol{r}_2$, with $\vec{a}$ decreasing linearly from 2 to 0 during the iterations,

and r_1, r_2 are random vectors in the range [0, 1]. Exploration is promoted when $|A| > 1$, forcing whales to move away from a given agent and thereby enhancing the global search ability of the algorithm.

Encircling prey: During the hunting process, humpback whales surround their prey. Other candidate solutions adjust their positions to converge toward the best-performing agent, with the current optimal solution regarded as the global best. Mathematically, this behavior is represented as follows:

$$\begin{aligned} \boldsymbol{D} &= \left|\boldsymbol{C} \cdot \boldsymbol{X}_p(t) - \boldsymbol{X}(t)\right|, \\ \boldsymbol{X}(t+1) &= \boldsymbol{X}_p(t) - \boldsymbol{A} \cdot \boldsymbol{D}, \end{aligned} \tag{12}$$

where $\boldsymbol{X}$ denotes the whale's position vector, and $\boldsymbol{X}_p$ denotes the position vector of the best solution found so far (the prey). In this case, exploitation is promoted when $|A| < 1$, ensuring that whales move toward the best solution identified.

Exploitation phase: The objective of this stage is to employ a bubble-net strategy to capture the prey. The bubble-net approach combines two mechanisms: (1) the shrinking encircling mechanism, in which the coefficient vector decreases from 2 to 0 over successive iterations; and (2) the spiral updating position mechanism, which calculates the distance between the whale and the prey and updates the position along a logarithmic spiral:

$$\boldsymbol{X}(t+1) = \boldsymbol{D}' e^{bt} cos(2\pi t) + \boldsymbol{X}^* \tag{13}$$

where $\boldsymbol{D}' = |\boldsymbol{X}^*(t) - \boldsymbol{X}(t)|$ is the distance between the whale and the prey, b is a constant defining the shape of the spiral, t is a random number in $[-1, 1]$, and $\boldsymbol{X}^*$ is the prey's position.

To integrate these two strategies, the WOA assigns a 50% probability of applying either the shrinking encircling mechanism or the spiral updating mechanism in each iteration, which can be formalized as follows:

$$\boldsymbol{X}(t+1) = \begin{cases} \boldsymbol{X}^*(t) - \boldsymbol{A} \cdot \boldsymbol{D} & \text{if} \quad p < 0.5, \\ \boldsymbol{D}' e^{bt} cos(2\pi t) + \boldsymbol{X}^* & \text{if} \quad p \geq 0.5, \end{cases} \tag{14}$$

where $p \in [0, 1]$ is a random probability.

In summary, the WOA technique is a very effective tool for solving difficult optimization issues in a variety of fields.

2.4. Goodness of Fit

Nine input variables were employed in the development of the WOA/MARS model, with energy consumption (EC) serving as the dependent variable. The objective was to predict the output variable (SEC) based on these nine process variables and to identify the model that best aligns with the experimental data. The coefficient of determination, R^2, was adopted as the primary metric to assess model performance [52,53]. For each data point t_i in the dataset, there exists a corresponding model-estimated value y_i. The observed values correspond to the actual measurements, whereas the predicted values are derived from the model. The variability within the dataset is quantified using the following sums of squares [52,53]:

1. First item, total sum of squares (SS_{tot}): $SS_{tot} = \sum_{i=1}^{n} (t_i - \bar{t})^2$. This measure is proportional to the sample variance.
2. Regression sum of squares (SS_{reg}): $SS_{reg} = \sum_{i=1}^{n} (y_i - \bar{t})^2$, sometimes dubbed as the total of squares described;

3. Residual sum of squares (RSS): $RSS = \sum_{i=1}^{n}(t_i - y_i)^2$.

In these equations, $\bar{t}$ is the arithmetic mean of the n data points in the experimental dataset and is calculated as

$$\bar{t} = \frac{1}{n}\sum_{i=1}^{n} t_i \tag{15}$$

The coefficient of determination R^2 is defined as [52,53]

$$R^2 \equiv 1 - \frac{RSS}{SS_{tot}} \tag{16}$$

As the R^2 statistic approaches unity, it indicates a diminishing disagreement between the experimental and projected data, thus signifying an improved model fit. Similarly, the mathematical expressions for the other two statistics used in this study, root-mean-square error (RMSE) and mean absolute error (MAE), are as follows:

$$RMSE \equiv \sqrt{\frac{1}{n}\sum_{i=1}^{n}(t_i - y_i)^2} \tag{17}$$

$$MAE \equiv \frac{\sum_{i=1}^{n}|t_i - y_i|}{n} \tag{18}$$

In order to optimize estimation accuracy (R^2), this study employs various approximation techniques. These methods model the EC as the dependent variable using nine input factors from experimental samples [47,54]. While the WOA-tuned MARS model was optimized using 10-fold cross-validation to handle its nonlinear structure and hyperparameter tuning, the linear baseline models—ridge regression (RR), linear regression (LR), and elastic-net regression (ENR)—were evaluated on the same independent test set without cross-validation. Given the inherent simplicity of these models and the absence of complex hyperparameters (beyond fixed regularization terms for RR/ENR), a single train–test split (80-20) was deemed sufficient to establish a fair and computationally efficient performance baseline for comparative purposes.

The efficacy of the MARS approximation is heavily contingent upon the selection of its hyperparameters [18,19,21–25,30,31], namely,

(a) *Maxfuncs*: the highest amount of hinge functions (*MF*).
(b) The degree to which variables interact (*D*).
(c) The penalty parameter (d), a GCV parameter imposing a penalty on each node, linked to model complexity (Equation (4)). When $d = 0$, terms are penalized, but nodes are not; when $d = -1$, no penalty applies. The common value is $d = 2$.
(d) The pruned model's maximum term count (*P*). The backward pruning stage eliminates ineffective terms to improve model generalization after the forward phase's overfitting.

To optimize these hyperparameters, this study employs the whale optimization algorithm (WOA) [26–29]. The WOA is a metaheuristic search algorithm that emulates the hunting behavior of humpback whales in nature, efficiently exploring the hyperparameter search space to identify optimal configurations.

The methodology for implementing the WOA/MARS model is as follows:

Data partitioning: The dataset is bifurcated; twenty percent of the data is employed for testing, while the remaining eighty percent is employed for training.

Model training and hyperparameter optimization:

(a) The training set is utilized to construct the WOA/MARS model.
(b) To adjust the MARS model parameters, a k-fold cross-validation method is used, taking the value k = 10 [52,53,55].
(c) The WOA algorithm is applied within this cross-validation framework to identify the optimal hyperparameter configuration.

Model construction: Once the best parameters are determined, the full training set is employed to construct the final model.

Model evaluation:

(a) The forecasted results are then evaluated against the actual values.
(b) The observed values are then contrasted with these forecasts.
(c) This comparison is used to assess how well the model fits the data.

In this study, several key parameters were defined to implement the WOA and guide the optimization process. The search space parameters are summarized in Table 2. The algorithm was configured with a population size of 40 and a maximum of 50 iterations as the stopping criterion. Unlike certain other optimization methods, this WOA implementation does not use a convergence threshold to terminate the process, relying instead on the predetermined number of iterations.

Table 2. Search limits of MARS hyperparameters for WOA tuning.

Hyperparameters	Lower Limit	Upper Limit
MF	3	100
D	1	4
d	1	30
P	−1	4

Predictive modeling frequently uses cross-validation to determine the true coefficient of determination R^2 [52,53]. In this investigation, where k = 10, the k-fold cross-validation technique was utilized to assess the predictive capacity of the WOA/MARS-based approximation [52,53,55].

The computational experiments were conducted on an iMac equipped with 24 GB of RAM and an Intel® Core™ i5-4570 CPU operating at 3.20 GHz with 4 cores. The software environment included the Ubuntu 22.04 operating system, R version 4.5.1, and several key R packages outlined below:

MARS approximation: `earth` package, version 5.3.4 [56,57];

WOA optimizer: `metaheuristicOpt` package, version 2.0.0 [28];

RR, LR, and elastic-net regression: `glmnet` package, version 4.1-9 [58].

The solution space's variation intervals used in this investigation are displayed below in Table 2.

The WOA metaheuristic optimizer was employed to fine-tune the MARS model parameters, determining optimal values for the upper bound of hinge functions, the level of variable interactions, the penalty parameter, and the maximum number of terms in the pruned model by evaluating cross-validation error differences. The four-dimensional search space corresponds to these four parameters under optimization. The root-mean-square error (RMSE) was adopted as the primary fitness metric and objective function [52,53], providing a robust measure of predictive accuracy for effective parameter evaluation and optimization.

Figure 5 provides a schematic representation of the WOA/MARS-based approximation process utilized in this investigation.

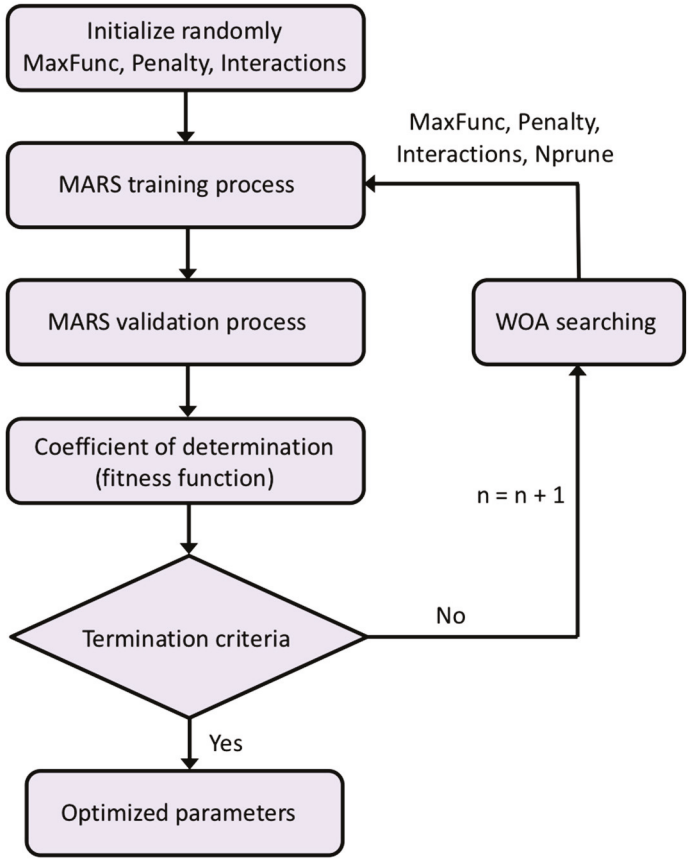


Figure 5. Flowchart of the WOA/MARS-based approximation.

3. Results and Discussion

Table 3 presents the best optimized hyperparameters for predicting electric energy consumption using the MARS-based approach, as found by the WOA optimizer.

Table 3. Optimized hyperparameters of the most accurately fitted MARS model, as determined by the WOA optimizer, for predicting electric energy consumption (EC).

Hyperparameters	Optimal Values
MF	17
D	3
d	24
P	0

The primary basis functions and associated coefficients for the optimally adjusted WOA/MARS-based model that was utilized to forecast EC are listed in Table 4. It is important to note that a basis function is defined using the hinge functions that follow the expression [18,19,21–25,30,31,58]:

$$B(x) = \begin{Bmatrix} x & \text{if} & x > 0 \\ 0 & \text{if} & x \leq 0 \end{Bmatrix} \quad (19)$$

Table 4. List of hinge functions and the associated c_i coefficients in the optimally adjusted WOA/MARS-based model for predicting energy consumption.

B_i	Definition	c_i
B_1	1	50.5129
B_2	$h(15.16 - \text{LaCRP})$	−2.3608
B_3	$h(\text{LaCRP} - 15.16)$	1.8091
B_4	$h(0.03 - \text{tCO}_2)$	−748.1992
B_5	$h(\text{tCO}_2 - 0.03)$	28.3717
B_6	$h(94.79 - \text{LaCPF})$	−3.0689
B_7	$h(\text{LaCPF} - 94.79)$	3.0469
B_8	$h(29.74 - \text{LaCRP}) \times h(0.03 - \text{tCO}_2)$	10.8099
B_9	$h(\text{LaCRP} - 29.74) \times h(0.03 - \text{tCO}_2)$	−12.4142
B_{10}	$h(0.03 - \text{tCO}_2) \times h(\text{LaCPF} - 94.36)$	−73.7558
B_{11}	$h(0.03 - \text{tCO}_2) \times h(94.36 - \text{LaCPF})$	95.9375
B_{12}	$h(\text{tCO}_2 - 0.03) \times h(\text{LaCPF} - 88.40)$	138.3409
B_{13}	$h(\text{tCO}_2 - 0.03) \times h(88.40 - \text{LaCPF})$	−97.0837
B_{14}	$h(\text{LaCPF} - 94.79) \times h(\text{LeCPF} - 71.15)$	0.0932
B_{15}	$h(\text{LaCPF} - 94.79) \times h(71.15 - \text{LeCPF})$	−0.0051
B_{16}	$h(\text{LaCRP} - 29.74) \times h(0.03 - \text{tCO}_2) \times h(\text{LaCPF} - 86.80)$	10.6701
B_{17}	$h(\text{LaCRP} - 29.74) \times h(0.03 - \text{tCO}_2) \times h(86.80 - \text{LaCPF})$	−0.1697

Fundamentally, MARS represents a type of nonparametric regression technique and can be considered an extension of linear regression. To systematically capture nonlinearities and interactions among input variables, MARS utilizes a weighted sum of the hinge functions described previously [18,19,21–25,30,31].

For comparative analysis, the lasso, ridge, and elastic-net regression techniques were also employed in this study. Their accuracy is contingent upon the parameter λ [30,32–35]. Table 5 displays the parameter λ's ideal value for these regression models, as established by the WOA optimizer.

Table 5. Optimized λ parameters for lasso, ridge, and elastic-net regression models, as determined by WOA optimization.

Optimal λ	SEC
RR	3.2982
LR	0.0939
ENR	0.1711

To ensure the robustness and validity of the WOA-optimized MARS model, diagnostic analyses of the residuals were performed. Figure 6a presents the residuals versus fitted values plot, which exhibits a random scatter around zero without discernible patterns, confirming the absence of heteroskedasticity and supporting the appropriateness of the model's functional form. Figure 6b displays the Q–Q plot of the residuals, indicating minor deviations from normality, a phenomenon commonly observed in industrial datasets due to operational outliers. Importantly, these deviations do not impair the model's predictive performance, as MARS is a nonparametric method that does not rely on the assumption of normally distributed residuals.

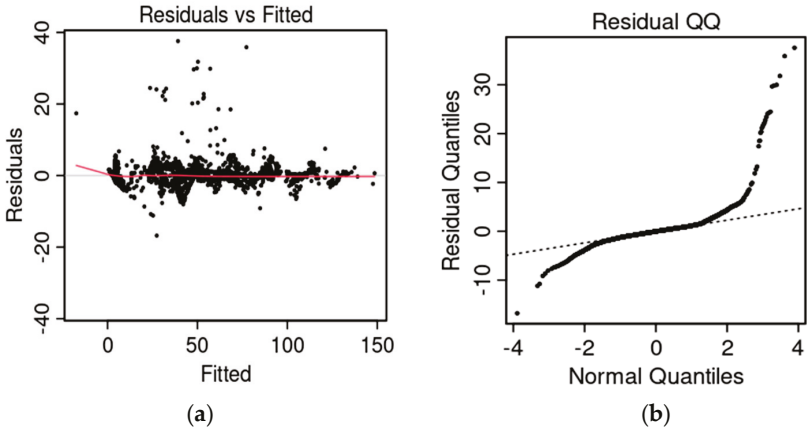


Figure 6. (**a**) Residual vs. fitted values graph and (**b**) Q–Q plot for the WOA/MARS model.

To further assess the interpretability of the model, partial dependence plots (PDPs) were generated for the three most influential variables (CO_2, LaCRP, and LaCPF), as presented in Figure 6. These plots depict the relationships between the input variables and the predicted energy consumption, incorporating both first- and second-order terms derived from the WOA-optimized MARS model, thereby elucidating their individual and interactive effects. In Figure 7a, EC (ordinate) is plotted against lagging current reactive power (abscissa), with four other factors kept constant. Similarly, Figure 7b,c show EC versus carbon dioxide emissions and the lagging current power factor, respectively, keeping eight variables fixed. Figure 7d presents a 3D plot of EC (applicate) against carbon dioxide emissions (abscissa) and lagging current reactive power (ordinate), with other variables constant. Likewise, Figure 7e,f depict 3D plots of EC versus carbon dioxide emissions and the lagging current power factor, and lagging versus leading current power factors.

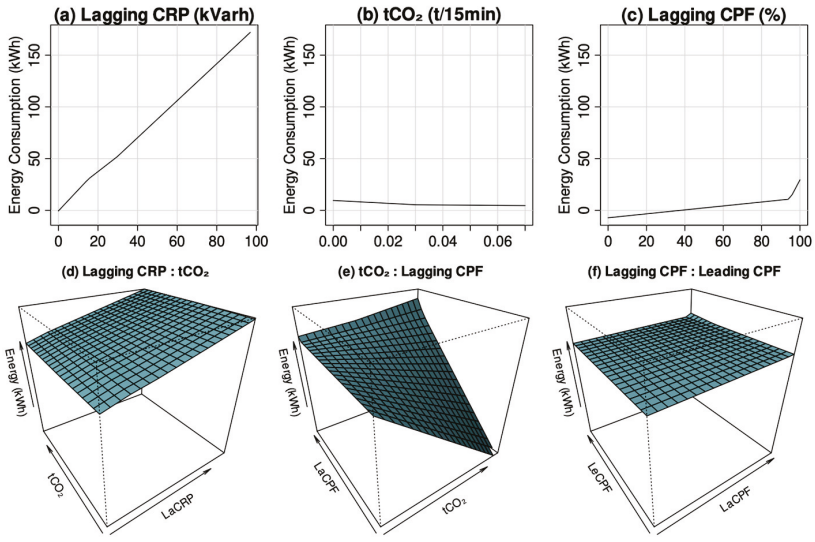


Figure 7. Visual illustration of the primary and secondary components that constitute the WOA/MARS model for estimating energy consumption: (**a**) first-order lagging current reactive power term; (**b**) first-order carbon dioxide term; (**c**) first-order lagging current power factor term; (**d**) second-order lagging current reactive power and carbon dioxide terms; (**e**) second-order carbon dioxide and lagging current power factor terms; and (**f**) second-order lagging current power factor and leading current power factor terms.

Table 6 summarizes the values of the coefficient of determination (R^2), correlation coefficients, and RMSE for various regression models applied to the EC output variable using the testing dataset. The models evaluated include the WOA/MARS-based approach [18,19,21–25,30,31], as well as ridge regression (RR), lasso regression (LR), and elastic-net regression (ENR) [30,32–35].

Table 6. Statistical performance indicators (*RMSE*, *MAE*, R^2, and *r*) of regression models for EC prediction on the test dataset.

Model	*RMSE* (kWh)	*MAE* (kWh)	R^2	*r*
WOA/MARS	1.7962	1.1801	0.9972	0.9985
RR	4.5150	2.6336	0.9820	0.9911
LR	4.4017	2.5274	0.9829	0.9914
ENR	4.4088	2.5413	0.9821	0.9912

The most recent statistical evaluation indicates that the optimally fitted MARS model achieves an R^2 value of 0.9972 and a correlation coefficient (*r*) of 0.9985 for the EC variable. These results demonstrate a strong agreement between the experimental data and the MARS predictions, indicating a high level of reliability in the model's goodness of fit. Consequently, this model was selected as the most suitable approach for estimating the dependent variable (EC).

These results are now contextualized by discussing comparable data-driven models reported in the literature. Mubarak et al. [59] introduced a stacked ensemble approach, Stack-XGBoost, which combines multiple machine learning models to predict active and reactive energy consumption in the steel industry. Their methodology employs extra trees regressor (ETR), adaptive boosting (AdaBoost), and random forest regressor (RFR) as base models, with an extreme gradient boosting (XGBoost) algorithm serving as the meta-learner. This approach demonstrated superior accuracy for short-term forecasting horizons. Nevertheless, this MARS model also exhibits strong performance, particularly with respect to the coefficient of determination, indicating an excellent fit to the data. Furthermore, a key advantage of the MARS model is its ability to quantify the importance of each independent variable, providing deeper insights into the factors driving energy consumption predictions.

As a supplementary outcome of these analyses, Table 7 and Figure 7 illustrate the hierarchical importance of the process variables (input factors) in predicting the EC (output dependent factor) [58,60] for this complex investigation.

Table 7. Ranking of the relevance of the operation variables in the best-fit WOA/MARS model for the EC as stated in the GCV criterion.

Input Variable	*Nsubsets*	*GCV*	*RSS*
Carbon dioxide emission (tCO_2)	16	100.0	100.0
Lagging current reactive power (LaCRP)	14	14.4	14.4
Lagging current power factor (LaCPF)	12	12.1	12.1
Leading current power factor (LeCPF)	12	12.1	12.1

In the MARS model, variable importance is estimated using three distinct criteria. First, the nsubsets criterion counts how many model subsets include the variable, with variables appearing in more subsets considered more important. Here, "subsets" refer to the subsets of terms generated during the backward pass of the model-building process. Second, the RSS criterion calculates the decrease in residual sum of squares (RSS) for each

subset relative to the previous subset during the backward pass. Variables causing larger net decreases in the RSS are deemed more important. Third, the GCV criterion operates similarly to RSS but uses generalized cross-validation (GCV) instead of the RSS. This criterion was employed to determine variable importance. Both the GCV and RSS columns are normalized so that the largest net decrease is set to 100, facilitating comparison.

Within the MARS framework, carbon dioxide emissions (tCO_2) were identified as the most influential predictor of the output variable EC. This is followed, in descending order of importance, by the lagging current reactive power (LaCRP), the lagging current power factor (LaCPF), and the leading current power factor (LeCPF). It is noteworthy that the remaining input variables were not included in the model as explanatory factors.

Figure 8 illustrates the proportional significance of process variables in the optimally fitted WOA/MARS-based approach for EC prediction, determined by the generalized cross-validation (GCV) criterion.

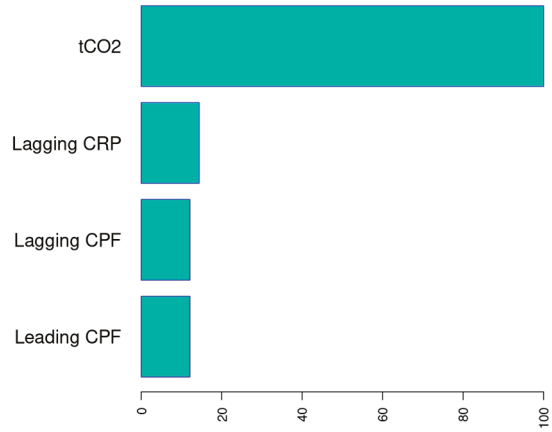


Figure 8. Variable importance in the optimal WOA/MARS model for EC prediction according to GCV criterion.

In steel plants, electrical energy consumption plays a critical role in production efficiency. Among the influencing factors, CO_2 emissions are the most significant, as they exhibit a direct correlation with energy usage, particularly in high-temperature operations and fossil fuel combustion processes [61]. The generation of CO_2 serves as an indicator of both energy consumption and process efficiency.

Lagging current reactive power, the lagging current power factor, and the leading current power factor exert a secondary influence, primarily affecting power quality and equipment efficiency rather than total energy consumption. While reactive power impacts supply stability, its effect is less pronounced than that of CO_2 emissions [62]. Similarly, the power factor influences energy losses but does not directly quantify total energy usage [63]. In contrast, CO_2 emissions provide a direct measure of both consumption and process efficiency, establishing them as the most relevant indicator of energy performance in steel mills.

Industries incur significant costs due to reactive energy, which is consumed without performing useful work, thereby increasing electricity expenses. Large energy consumers, particularly those operating machinery with transformers, tend to generate higher amounts of reactive energy. A leading power factor indicates that voltage lags behind current (capacitive load and negative reactive power), whereas a lagging power factor signifies that current lags behind voltage (inductive load and positive reactive power). Low power

factors necessitate higher currents to maintain the same active power (P), which, in turn, requires larger conductor sizes.

Although reactive power and the power factor influence operational efficiency, CO_2 emissions remain the primary determinant for optimizing electric energy consumption in steelworks. Reducing CO_2 emissions is critical for managing energy usage and minimizing associated emission costs [64].

Energy efficiency plays a crucial role in reducing industrial greenhouse gas emissions, with the iron and steel sector accounting for approximately 5% of global anthropogenic CO_2 emissions [65]. The implementation of energy-efficient technologies in steel plants decreases energy consumption (EC), thereby reducing operational costs and mitigating environmental impacts.

This study effectively estimates EC using the WOA/MARS-based model, demonstrating strong agreement with observed data. Figure 9 presents a comparison of experimental and predicted EC values obtained from the LR, RR, ENR, and WOA/MARS models. The MARS model provides the most accurate regression fit, achieving the highest R^2, thereby confirming the WOA/MARS approach as the optimal solution.

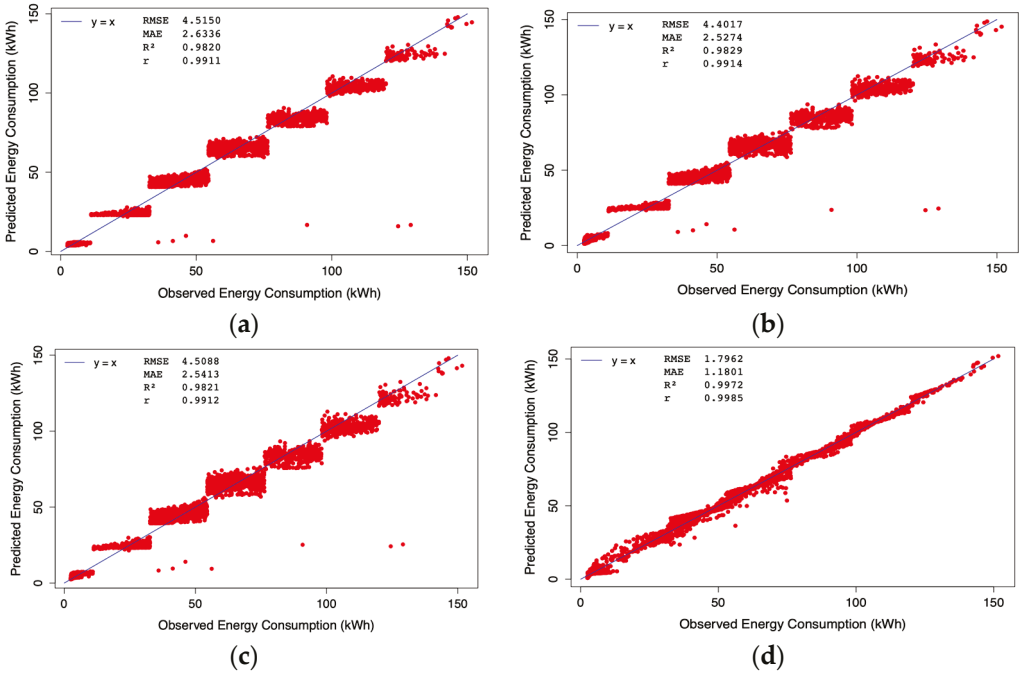


Figure 9. Predicted EC vs. observed values on the test set with (**a**) RR model ($R^2 = 0.9820$); (**b**) LR model ($R^2 = 0.9829$); (**c**) ENR model ($R^2 = 0.9821$); and (**d**) WOA/MARS model ($R^2 = 0.9972$).

4. Conclusions

This study demonstrates the effectiveness of a WOA-tuned MARS model for estimating energy consumption in steel production, achieving exceptional predictive accuracy (R^2 = 0.9972). The model's superiority over linear alternatives (RR, LR, and ENR) stems from its ability to capture nonlinear interactions between critical process variables, which align with established steelmaking physics:

- CO_2 emissions and energy consumption (EC) exhibit a strong coupling due to shared dependencies on production volume and combustion intensity.
- Lagging reactive power (LaCRP) reflects inductive load dominance, where a poor power factor increases the apparent power demand and, thus, EC.

- Lagging current power factor (LaCPF) extremes indicate suboptimal electrical efficiency, correlating with higher energy losses.

In conclusion, the WOA-tuned MARS model presents a robust methodology for estimating electrical energy consumption in the steel industry. Given the absence of analytical equations capable of accurately predicting steelworks' electrical energy consumption (SEC) from experimental data alone, alternative diagnostic approaches become indispensable. This methodology demonstrates adaptability to a broad spectrum of manufacturing processes, though its implementation must account for the specific operational characteristics and conditions of individual steelworks.

Limitations and future work: While the proposed methodology demonstrates high accuracy, several limitations and opportunities for future research should be acknowledged:

- Dataset specificity: The current analysis focuses on data from Gwangyang Steelworks (South Korea). Future studies should validate the model's generalizability across diverse geographical regions and steel plant configurations.
- External factors: Incorporating market demand fluctuations, regulatory changes, and seasonal variations could refine the model's adaptability to dynamic industrial conditions.
- Temporal validation: Employing time-aware data splits (e.g., training on historical data and testing on recent periods) would further assess the model's robustness to operational evolution.
- Potential methodological extensions include the following:
    - Dynamic hyperparameter tuning using expanding/rolling window strategies to capture long-term trends.
    - Hybrid approaches (e.g., MARS + ARIMA) to balance interpretability and predictive power for time-series forecasting.

Author Contributions: Conceptualization, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; methodology, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; software, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; validation, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; formal analysis, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; data curation, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; writing—original draft preparation, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; writing—review and editing, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; visualization, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S.; supervision, P.J.G.-N., E.G.-G., L.A.M.-G., L.Á.-d.-P., M.M.-F. and A.B.-S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from the following link: https://archive.ics.uci.edu/dataset/851/steel+industry+energy+consumption (accessed on 15 April 2024).

Acknowledgments: The Mathematics Department at the University of Oviedo generously provided computational assistance. Likewise, the authors would like to thank Anthony Ashworth for revising this research paper in English.

Conflicts of Interest: The authors declare no competing interests.

References

1. Xiao, L.; Shao, W.; Liang, T.; Wang, C. A Combined Model Based on Multiple Seasonal Patterns and Modified Firefly Algorithm for Electrical Load Forecasting. *Appl. Energy* **2016**, *167*, 135–153. [CrossRef]
2. Conejo, A.N.; Birat, J.-P.; Dutta, A. A Review of the Current Environmental Challenges of the Steel Industry and Its Value Chain. *J. Environ. Manag.* **2020**, *259*, 109782. [CrossRef]
3. Gogeri, I.; Gouda, K.C.; Sumathy, T. Modeling and Forecasting Atmospheric Carbon Dioxide Concentrations at Bengaluru City in India. *Stoch. Environ. Res. Risk Assess.* **2024**, *38*, 1297–1312. [CrossRef]
4. Tian, W.; An, H.; Li, X.; Li, H.; Quan, K.; Lu, X.; Bai, H. CO_2 Accounting Model and Carbon Reduction Analysis of Iron and Steel Plants Based on Intra- and Inter-Process Carbon Metabolism. *J. Clean. Prod.* **2022**, *360*, 132190. [CrossRef]
5. Holappa, L. A General Vision for Reduction of Energy Consumption and CO_2 Emissions from the Steel Industry. *Metals* **2020**, *10*, 1117. [CrossRef]
6. González-Álvarez, M.A.; Montañés, A. CO_2 Emissions, Energy Consumption, and Economic Growth: Determining the Stability of the 3E Relationship. *Econ. Modell.* **2023**, *121*, 106195. [CrossRef]
7. Bai, H.; Lu, X.; Li, H.; Zhao, L.; Liu, X.; Li, N.; Wei, W.; Cang, D. The Relationship between Energy Consumption and CO_2 Emissions in Iron and Steel Making. In *Energy Technology 2012*; Salazar-Villalpando, M.D., Neelameggham, N.R., Guillen, D.P., Pati, S., Krumdick, G.K., Eds.; Wiley: Hoboken, NJ, USA, 2012; pp. 125–132, ISBN 978-1-118-29138-2.
8. Griffin, P.W.; Hammond, G.P. Industrial Energy Use and Carbon Emissions Reduction in the Iron and Steel Sector: A UK Perspective. *Appl. Energy* **2019**, *249*, 109–125. [CrossRef]
9. Pardo, N.; Moya, J.; Vatapoulos, K. *Prospective Scenarios on Energy Efficiency and CO2 Emissions in the EU Iron & Steel Industry*; Publications Office of European Union, Institute for Energy and Transport: Luxembourg, 2015.
10. Wang, Z.-X.; Li, Q.; Pei, L.-L. A Seasonal GM(1,1) Model for Forecasting the Electricity Consumption of the Primary Economic Sectors. *Energy* **2018**, *154*, 522–534. [CrossRef]
11. Candanedo, L.M.; Feldheim, V.; Deramaix, D. Data Driven Prediction Models of Energy Use of Appliances in a Low-Energy House. *Energy Build.* **2017**, *140*, 81–97. [CrossRef]
12. Le, T.; Zhao, J. A Data Fusion Algorithm Based on Neural Network Research in Building Environment of Wireless Sensor Network. *Int. J. Future Gener. Commun. Netw.* **2015**, *8*, 295–306. [CrossRef]
13. Alsharif, M.H.; Younes, M.K.; Kim, J. Time Series ARIMA Model for Prediction of Daily and Monthly Average Global Solar Radiation: The Case Study of Seoul, South Korea. *Symmetry* **2019**, *11*, 240. [CrossRef]
14. Lv, J.; Li, X.; Ding, L.; Jiang, L. Applying Principal Component Analysis and Weighted Support Vector Machine in Building Cooling Load Forecasting. In Proceedings of the 2010 International Conference on Computer and Communication Technologies in Agriculture Engineering, Chengdu, China, 12–13 June 2010; pp. 434–437.
15. Amiri, S.S.; Mottahedi, M.; Asadi, S. Using Multiple Regression Analysis to Develop Energy Consumption Indicators for Commercial Buildings in the U.S. *Energy Build.* **2015**, *109*, 209–216. [CrossRef]
16. Mocanu, E.; Nguyen, P.H.; Gibescu, M.; Kling, W.L. Deep Learning for Estimating Building Energy Consumption. *Sustain. Energy Grids Netw.* **2016**, *6*, 91–99. [CrossRef]
17. Sathishkumar, V.E.; Shin, C.; Cho, Y. Efficient Energy Consumption Prediction Model for a Data Analytic-Enabled Industry Building in a Smart City. *Build. Res. Inf.* **2021**, *49*, 127–143. [CrossRef]
18. Friedman, J.H. Multivariate Adaptive Regression Splines. *Ann. Statist* **1991**, *19*, 1–67. [CrossRef]
19. Kooperberg, C. Multivariate Adaptive Regression Splines. In *Wiley StatsRef: Statistics Reference Online*; Kenett, R.S., Longford, N.T., Piegorsch, W.W., Ruggeri, F., Eds.; Wiley: Boca Raton, FL, USA, 2014; ISBN 978-1-118-44511-2.
20. Lange, C. *Practical Machine Learning with R: Tutorials and Case Studies*, 1st ed.; CRC Press LLC: Abingdon, UK, 2024; ISBN 978-1-040-02124-8.
21. Marsland, S. *Machine Learning: An Algorithmic Perspective*, 2nd ed.; CRC Press, Taylor and Francis Group: Boca Raton, FL, USA, 2015; ISBN 978-1-4665-8333-7.
22. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; Adaptive Computation and Machine Learning Series; MIT Press: Cambridge, MA, USA, 2012; ISBN 978-0-262-01802-9.
23. Sekulic, S.; Kowalski, B.R. MARS: A Tutorial. *J. Chemom.* **1992**, *6*, 199–216. [CrossRef]
24. Vidoli, F. Evaluating the Water Sector in Italy through a Two Stage Method Using the Conditional Robust Nonparametric Frontier and Multivariate Adaptive Regression Splines. *Eur. J. Oper. Res.* **2011**, *212*, 583–595. [CrossRef]
25. Xu, Q.-S.; Daszykowski, M.; Walczak, B.; Daeyaert, F.; De Jonge, M.R.; Heeres, J.; Koymans, L.M.H.; Lewi, P.J.; Vinkers, H.M.; Janssen, P.A.; et al. Multivariate Adaptive Regression Splines—Studies of HIV Reverse Transcriptase Inhibitors. *Chemom. Intell. Lab. Syst.* **2004**, *72*, 27–34. [CrossRef]

26. Ebrahimgol, H.; Aghaie, M.; Zolfaghari, A.; Naserbegi, A. A Novel Approach in Exergy Optimization of a WWER1000 Nuclear Power Plant Using Whale Optimization Algorithm. *Ann. Nucl. Eng.* **2020**, *145*, 107540. [CrossRef]
27. Huang, K.-W.; Wu, Z.-X.; Jiang, C.-L.; Huang, Z.-H.; Lee, S.-H. WPO: A Whale Particle Optimization Algorithm. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 115. [CrossRef]
28. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
29. Onwubolu, G.C.; Babu, B.V. *New Optimization Techniques in Engineering*; Studies in Fuzziness and Soft Computing; Springer: Berlin/Heidelberg, Germany, 2004; Volume 141, ISBN 978-3-642-05767-0.
30. Zheng, G.; Zhang, W.; Zhou, H.; Yang, P. Multivariate Adaptive Regression Splines Model for Prediction of the Liquefaction-Induced Settlement of Shallow Foundations. *Soil Dyn. Earthq. Eng.* **2020**, *132*, 106097. [CrossRef]
31. Kuhn, M.; Johnson, K. *Applied Predictive Modeling*; Springer: New York, NY, USA, 2013; ISBN 978-1-4614-6848-6.
32. Izenman, A.J. *Modern Multivariate Statistical Techniques*; Springer Texts in Statistics; Springer: New York, NY, USA, 2008; ISBN 978-0-387-78188-4.
33. Keeney, A.J.; Beseler, C.L.; Ingold, S.S. County-Level Analysis on Occupation and Ecological Determinants of Child Abuse and Neglect Rates Employing Elastic Net Regression. *Child Abus. Negl.* **2023**, *137*, 106029. [CrossRef] [PubMed]
34. Saleh, A.K.M.E.; Arashi, M.; Kibria, B.M.G. *Theory of Ridge Regression Estimation with Applications*; Wiley: Hoboken, NJ, USA, 2019; ISBN 978-1-118-64447-8.
35. Zhang, L.; Tedde, A.; Ho, P.; Grelet, C.; Dehareng, F.; Froidmont, E.; Gengler, N.; Brostaux, Y.; Hailemariam, D.; Pryce, J.; et al. Mining Data from Milk Mid-Infrared Spectroscopy and Animal Characteristics to Improve the Prediction of Dairy Cow's Liveweight Using Feature Selection Algorithms Based on Partial Least Squares and Elastic Net Regressions. *Comput. Electron. Agric.* **2021**, *184*, 106106. [CrossRef]
36. Rogers, S.; Girolami, M. *A First Course in Machine Learning*; Chapman and Hall/CRC: London, UK, 2016; ISBN 978-1-4987-3854-5.
37. Vapnik, V.N. *Statistical Learning Theory*; Adaptive and Learning Systems for Signal Processing, Communications, and Control; Wiley: New York, NY, USA; Weinheim, Germany, 1998; ISBN 978-0-471-03003-4.
38. Gackowski, M.; Szewczyk-Golec, K.; Pluskota, R.; Koba, M.; Mądra-Gackowska, K.; Woźniak, A. Application of Multivariate Adaptive Regression Splines (MARSplines) for Predicting Antitumor Activity of Anthrapyrazole Derivatives. *Int. J. Mol. Sci.* **2022**, *23*, 5132. [CrossRef] [PubMed]
39. De Benedetto, D.; Barca, E.; Castellini, M.; Popolizio, S.; Lacolla, G.; Stellacci, A.M. Prediction of Soil Organic Carbon at Field Scale by Regression Kriging and Multivariate Adaptive Regression Splines Using Geophysical Covariates. *Land* **2022**, *11*, 381. [CrossRef]
40. Majeed, F.; Ziggah, Y.Y.; Kusi-Manu, C.; Ibrahim, B.; Ahenkorah, I. A Novel Artificial Intelligence Approach for Regolith Geochemical Grade Prediction Using Multivariate Adaptive Regression Splines. *Geosystems Geoenvironment* **2022**, *1*, 100038. [CrossRef]
41. Agwu, O.E.; Elraies, K.A.; Alkouh, A.; Alatefi, S. Mathematical Modelling of Drilling Mud Plastic Viscosity at Downhole Conditions Using Multivariate Adaptive Regression Splines. *Geoenergy Sci. Eng.* **2024**, *233*, 212584. [CrossRef]
42. Sharma, J.; Kumar Mitra, S. Developing a Used Car Pricing Model Applying Multivariate Adaptive Regression Splines Approach. *Expert Syst. Appl.* **2024**, *236*, 121277. [CrossRef]
43. Menéndez García, L.A.; Sánchez Lasheras, F.; García Nieto, P.J.; Álvarez De Prado, L.; Bernardo Sánchez, A. Predicting Benzene Concentration Using Machine Learning and Time Series Algorithms. *Mathematics* **2020**, *8*, 2205. [CrossRef]
44. Ali, H.M.; Mohammadi Ghaleni, M.; Moghaddasi, M.; Moradi, M. A Novel Interpretable Hybrid Model for Multi-Step Ahead Dissolved Oxygen Forecasting in the Mississippi River Basin. *Stoch. Environ. Res. Risk Assess.* **2024**, *38*, 4629–4656. [CrossRef]
45. Sathishkumar, V.E.; Lim, J.; Lee, M.; Cho, K.; Park, J.; Shin, C.; Cho, Y. Steel Industry Energy Consumption. 2021. Available online: https://archive.ics.uci.edu/dataset/851/steel+industry+energy+consumption (accessed on 15 April 2024).
46. Sathishkumar, V.E.; Lim, J.; Lee, M.; Cho, K.; Park, J.; Shin, C. Industry Energy Consumption Prediction Using Data Mining Techniques. *Int. J. Energy Inf. Commun.* **2020**, *11*, 7–14. [CrossRef]
47. Dutta, S.K.; Chokshi, Y.B. *Basic Concepts of Iron and Steel Making*; Springer: Singapore, 2020; ISBN 978-981-15-2436-3.
48. Bishop, C.M. *Pattern Recognition and Machine Learning*; Information science and statistics; Springer: New York, NY, USA, 2006; ISBN 978-0-387-31073-2.
49. Smith, C. *Decision Trees and Random Forests: A Visual Introduction for Beginners*; Blue Windmill Media: Thornton, ON, Canada, 2017; ISBN 978-1-5498-9375-9.
50. Qureshi, M.U.; Mahmood, Z.; Rasool, A.M. Using Multivariate Adaptive Regression Splines to Develop Relationship between Rock Quality Designation and Permeability. *J. Rock Mech. Geotech. Eng.* **2022**, *14*, 1180–1187. [CrossRef]
51. Verma, V.K.; Banodha, U.; Malpani, K. Optimization with Adaptive Learning: A Better Approach for Reducing SSE to Fit Accurate Linear Regression Model for Prediction. *Int. J. Adv. Comput. Sci. Appl.* **2024**, *15*, 168. [CrossRef]

52. Faraway, J.J. *Linear Models with R*, 2nd ed.; Texts in Statistical Science; CRC Press: Boca Raton, FL, USA, 2014; ISBN 978-1-4398-8734-9.
53. Thijssen, J.J.J. *A Concise Introduction to Statistical Inference*; CRC Press: Boca Raton, FL, USA, 2017; ISBN 978-1-4987-5578-8.
54. Henderson, S.; Royall, K.E. *Barrow Steelworks—Continuous Casting*; Stanley Henderson: Los Angeles, CA, USA, 2021; ISBN 978-1-913898-24-3.
55. Jung, Y. Multiple Predicting K-Fold Cross-Validation for Model Selection. *J. Nonparametr. Stat.* **2018**, *30*, 197–215. [CrossRef]
56. Agresti, A.; Kateri, M. *Foundations of Statistics for Data Scientists: With R and Python*; Chapman & Hall/CRC Texts in Statistical Science; CRC Press: Boca Raton, FL, USA, 2021; ISBN 978-1-000-46293-7.
57. Milborrow, S. *Earth: Multivariate Adaptive Regression Splines*; R Package; 2023. Available online: https://cran.r-project.org/web/packages/earth/earth.pdf (accessed on 5 October 2024).
58. Tay, J.K.; Narasimhan, B.; Hastie, T. Elastic Net Regularization Paths for All Generalized Linear Models. *J. Stat. Soft.* **2023**, *106*, 1–31. [CrossRef]
59. Mubarak, H.; Sanjari, M.J.; Stegen, S.; Abdellatif, A. Improved Active and Reactive Energy Forecasting Using a Stacking Ensemble Approach: Steel Industry Case Study. *Energies* **2023**, *16*, 7252. [CrossRef]
60. Deisenroth, M.P.; Faisal, A.A.; Ong, C.S. *Mathematics for Machine Learning*; Cambridge University Press: Cambridge, UK, 2020; ISBN 978-1-108-47004-9.
61. Pan, S.-Y.; Chang, E.E.; Chiang, P.-C. CO_2 Capture by Accelerated Carbonation of Alkaline Wastes: A Review on Its Principles and Applications. *Aerosol Air Qual. Res.* **2012**, *12*, 770–791. [CrossRef]
62. Maarif, M.R.; Saleh, A.R.; Habibi, M.; Fitriyani, N.L.; Syafrudin, M. Energy Usage Forecasting Model Based on Long Short-Term Memory (LSTM) and eXplainable Artificial Intelligence (XAI). *Information* **2023**, *14*, 265. [CrossRef]
63. Turkovskyi, V.; Malinovskyi, A.; Muzychak, A.; Turkovskyi, O. Using the Constant Current—Constant Voltage Converters to Effectively Reduce Voltage Fluctuations in the Power Supply Systems for Electric Arc Furnaces. *East.-Eur. J. Enterp. Technol.* **2020**, *6*, 54–63. [CrossRef]
64. Sizirici, B.; Fseha, Y.; Cho, C.-S.; Yildiz, I.; Byon, Y.-J. A Review of Carbon Footprint Reduction in Construction Industry, from Design to Operation. *Materials* **2021**, *14*, 6094. [CrossRef] [PubMed]
65. Perpiñán, J.; Bailera, M.; Romeo, L.; Peña, B.; Eveloy, V. CO_2 Recycling in the Iron and Steel Industry via Power-to-Gas and Oxy-Fuel Combustion. *Energies* **2021**, *14*, 7090. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Integrating Machine Learning and Scenario Modelling for Robust Population Forecasting Under Crisis and Data Scarcity

Michael Politis [1], Nicholas Christakis [1,*], Zoi Dorothea Pana [2] and Dimitris Drikakis [1]

[1] Institute for Advanced Modelling and Simulation, University of Nicosia, CY 2417 Nicosia, Cyprus; politis.michael@outlook.com (M.P.); drikakis.d@unic.ac.cy (D.D.)
[2] Department of Basic and Clinical Studies, University of Nicosia Medical School, CY 2408 Nicosia, Cyprus; pana.z@unic.ac.cy
* Correspondence: christakis.n@unic.ac.cy

Abstract: This study introduces a new ensemble framework for demographic forecasting that systematically incorporates stylised crisis scenarios into rate and population projections. While scenario reasoning is common in qualitative foresight, its quantitative application in demography remains underdeveloped. Our method combines autoregressive lags, global predictors, and robust regression with a trend-anchoring mechanism, enabling stable projections from short official time series (15–20 years in length). Scenario shocks are operationalised through binary event flags for pandemics, refugee inflows, and financial crises, which influence fertility, mortality, and migration models before translating into cohort and population trajectories. Results demonstrate that shocks with strong historical precedence, such as Germany's migration surges, are convincingly reproduced and leave enduring effects on projected populations. Conversely, weaker or non-recurrent shocks, typical in Norway and Portugal, produce muted scenario effects, with baseline momentum dominating long-term outcomes. At the national level, total population aggregates mitigate temporary shocks, while cohort-level projections reveal more pronounced divergences. Limitations include the short length of the training series, the reduction of signals when shocks do not surpass historical peaks, and the loss of granularity due to age grouping. Nevertheless, the framework shows how robust statistical ensembles can extend demographic forecasting beyond simple trend extrapolation, providing a formal and transparent quantitative tool for stress-testing population futures under both crisis and stability.

Keywords: demographic forecasting; cohort component method; machine learning; sparse data; scenario modelling

MSC: 68T07; 62H99; 62F99; 92D25

1. Introduction

Population modelling sits at the crossroads of demography, applied mathematics, and computational science, offering vital tools for understanding societal change. Classical methods—from stable population theory to matrix-based models—have provided important insights into growth, decline, and long-term dynamics, yet they often struggle to balance detailed demographic variation with the need for large-scale, policy-relevant predictions. This tension motivates the current work, which broadens traditional frameworks by connecting micro-level demographic processes with macro-level population trajectories through a clear, scenario-aware forecasting design.

Precise and clear population forecasts are vital for public policy, infrastructure planning, and resource allocation. The cohort-component method (CCM) [1] is the main approach in official projections, analysing population change by age- and sex-specific fertility, mortality, and migration. While appreciated for its simplicity, CCM's reliance on trend continuation reduces its effectiveness during demographic shocks like pandemics, forced migration, and financial crises. These challenges emphasise the need for forecasting frameworks that remain reliable in volatile environments and operate effectively with limited data.

This study addresses the following question: *Can population forecasts in crisis-prone or data-limited settings be made more robust by integrating scenario-sensitive signals into a machine-learning (ML) ensemble that models demographic rates while suppressing spurious fluctuations?* Our aim is to produce forecasts in which deviations from the baseline stem solely from explicit, well-defined scenarios, thereby enhancing interpretability and credibility.

To this end, we suggest a regression-ensemble framework that combines robust Huber regression with a correction mechanism based on the slope of an Exponential Moving Average (EMA) of historical rates. The ensemble retains lasting demographic signals, prevents overfitting to short-term noise, and incorporates explicit scenario flags to trigger genuine shocks. This design improves the CCM with data-driven modelling while keeping transparency.

1. It formalises a scenario-conditioned regression ensemble combining Huber regression with an EMA-slope anchor to stabilise trajectories.
2. It specifies a transparent mapping from scenario switches to fertility, mortality, and migration models, which are then propagated through age-structured CCM projections.
3. It defines a minimal, reproducible validation protocol for short official series using rolling one-year-ahead evaluation and decorrelation-guided lag limits.
4. It documents how scenario responsiveness depends on historical signal strength, clarifying when shocks generate observable divergence from baseline.

Two broad classes of population forecasting approaches contextualise our contribution: deterministic CCMs and statistical/ML methods. CCM remains foundational in national projections [1–12], with demographic transitions modelled via fertility, mortality, and migration [13–16]. However, their dependence on extrapolation makes them vulnerable to unexpected shocks [17–19].

Statistical models such as ARIMA, exponential smoothing, and transfer-function models [20–25] provide data-driven alternatives. The Lee–Carter model [26,27] and Bayesian extensions [28,29] underpin modern probabilistic forecasts, while multilevel models address data scarcity [30–32]. Model averaging further improves robustness, especially for migration and fertility [33–38].

Machine-learning methods are increasingly used in demographic forecasting [39–50], with techniques including LSTM, random forests, XGBoost [51,52], and regularised regressions [53–55]. Yet, these models often lack interpretability, struggle under sparse data, and neglect scenario-based reasoning [14,30]; they have historically been used to demonstrate how flexible function approximators can capture complex demographic patterns when long, high-frequency, or granular datasets are available. Ensemble learning [56–58] and symbolic discovery such as SINDy [59], though promising, face difficulties with noise [60,61]. Geospatial and census-integrated models [62–69] and spatial Bayesian hierarchies [70–76] further highlight the need for robust, interpretable methods in sparse contexts.

A central methodological gap concerns the treatment of shocks. Demographic components—especially migration—are highly shock-sensitive [37], yet many models implicitly assume temporal continuity, leading to poor prediction of inflexion points [45]. Empirical

evidence shows distinctive crisis responses: short-term fertility rebounds after pandemics [77], increased health expenditure during public health emergencies [78], rising unemployment in financial shocks [79], and altered fertility among refugees [80]. In our framework, such stylised effects enter through scenario flags rather than being mistaken for noise.

We therefore extend the CCM by embedding scenario-sensitive flags within a regression ensemble that uses Huber regression [81,82] to down-weight outliers and an EMA-slope anchor [25,83] to preserve long-run signals. A smoothing safeguard prevents implausible year-on-year jumps. This reframes population forecasts as conditional statements tied to explicit assumptions, rather than unqualified projections of past trends.

The approach is designed for sparse data environments: lagged dynamics, socioeconomic indicators, and scenario switches capture available structure without overfitting. Validation uses data from Germany, Norway, and Portugal (2001–2024), generating forecasts to 2040 under baseline and crisis scenarios. One-year-ahead out-of-fold scoring evaluates both overall and age-specific accuracy. Scenario design remains grounded in empirical evidence to avoid unrealistic deviations.

Conceptually, the framework elevates scenario reasoning from a qualitative planning exercise to a formal element of statistical population modelling, linking narrative foresight to quantitative projection. While limited by short historical series and by the simplicity required for interpretability, the method offers a transparent and reproducible basis for exploring demographic futures under both stability and disruption.

The remainder of the paper is organised as follows. Section 2 describes the data and preprocessing; Section 3 details the modelling framework, regression ensemble, scenario flags, and stability analysis; Section 4 presents results for Germany, Norway, and Portugal including baseline and crisis forecasts to 2040; and Section 5 discusses limitations, policy implications, and directions for future research.

2. Data

Our analysis relies on demographic and socioeconomic data from Germany, Portugal, and Norway, collected from official national statistical agencies [6,8,10]. The core of the dataset includes age-specific fertility rates (ASFRs), mortality rates, and emigration rates, with immigration recorded as absolute figures to reflect each national system's reporting conventions. Each variable is consistently broken down by age group and sex, providing a structured set of cohort-level transition components. This structure ensures comparability across countries and aligns the dataset with the cohort-component framework that underpins most demographic forecasting models.

The selection of Germany, Portugal, and Norway reflects both data availability and the chance to test the framework under contrasting conditions. These countries provide consistently curated series with enough depth to support robust model evaluation, while differing in demographic and institutional settings. Germany represents a large continental population with significant migration inflows and periodic census revisions. Portugal exemplifies a medium-sized European nation with strong emigration patterns and a more recent record of digitisation. Norway offers a smaller but data-rich population, where migration closely relates to economic cycles and policy. This variation enables the assessment of the framework's performance across diverse contexts, thereby increasing the relevance of the results beyond any single case.

The choice of three countries is based on the fact that they offer harmonised annual demographic and socioeconomic series with relatively few structural breaks. Broader international datasets, such as those from the United Nations or the World Bank, indeed provide longer historical coverage; however, they often include definitional shifts, aggregation

changes, or interpolated intervals that complicate the estimation of short-horizon annual models. The present study, therefore, focuses on demonstrating the mechanism of the hybrid ensemble under controlled, internally consistent conditions. We acknowledge that extending the approach to a broader set of countries is a valuable next step and would help assess its generalisability under more heterogeneous data environments.

To complement the demographic series, we include socioeconomic indicators that provide additional context for population change. Specifically, GDP per capita, unemployment rates, and health expenditure as a percentage of GDP are incorporated, with data sourced from the World Bank [84]. These indicators are not considered direct causal factors but rather auxiliary covariates that enable the regression ensemble to adjust forecasts in response to broader economic conditions. This approach ensures that the data environment reflects both the internal momentum of demographic rates and the external pressures that influence them.

The analysis covers roughly the last twenty years of available data. For Germany and Norway, continuous data series date back to the early 2000s, while for Portugal, reliable migration data begin in 2008. This time frame is selected based on data availability and statistical considerations. Despite Europe's long history of demographic record-keeping, digitised and harmonised series are limited before the 1990s, particularly for migration and socioeconomic indicators. Additionally, older population data are highly autocorrelated, offering little extra predictive value once recent demographic trends are taken into account. Focusing on the past 20 years thus balances the capture of relevant changes and ensures comparability across countries.

Although official data sources form the basis of the dataset, they are not without limitations. Migration statistics remain vulnerable to under-reporting and definitional changes, and mortality and fertility series can also be affected by reclassification or methodological revisions. In the absence of external benchmarks, we assume that official figures are internally consistent and representative for modelling purposes. This approach provides advantages in terms of comparability and alignment with international standards, but it also introduces potential discontinuities, especially around census revisions. From a modelling perspective, this makes it essential to adopt methods that can accommodate sudden adjustments without mistaking them for genuine demographic shocks.

To assess internal consistency, we calculated the residual between the reported total population and the value implied by demographic accounting:

$$R_t = P_t - \big(I_t - D_t + M_t\big), \tag{1}$$

where P_t denotes the reported population estimate, I_t inflows (births and the previous population P_{t-1}), D_t reported deaths, and M_t net migration in the year t. Applying this diagnostic across all three countries showed that Norway and Portugal exhibit only minor deviations consistent with routine adjustments, whereas Germany shows clear negative spikes in 2011 and 2022, both census years. These patterns indicate substantial revisions following enumeration and illustrate how population totals can diverge from component-based accounting even in well-developed statistical systems. For modelling, such corrections highlight the need to prevent short-term discontinuities from influencing long-term forecasts, while still allowing explicit scenario shocks to enter transparently (see Figure 1).

We also assessed the statistical redundancy in the demographic rate series to inform lag selection for the regression ensemble. The decorrelation time is defined as the lag at which the absolute value of the autocorrelation function (ACF) drops below 20%, indicating that observations beyond this point provide little additional predictive information. Based on this diagnosis, the maximum lag was set at five years to strike a balance between

capturing useful information and the limited length of the datasets. This complements the decision to restrict the analysis to approximately the past two decades, as extending further back would offer minimal useful signal and could introduce measurement inconsistencies. Table 1 displays the estimated decorrelation steps for each country, illustrating that the five-year cap maintains the primary predictive content within the selected timeframe.

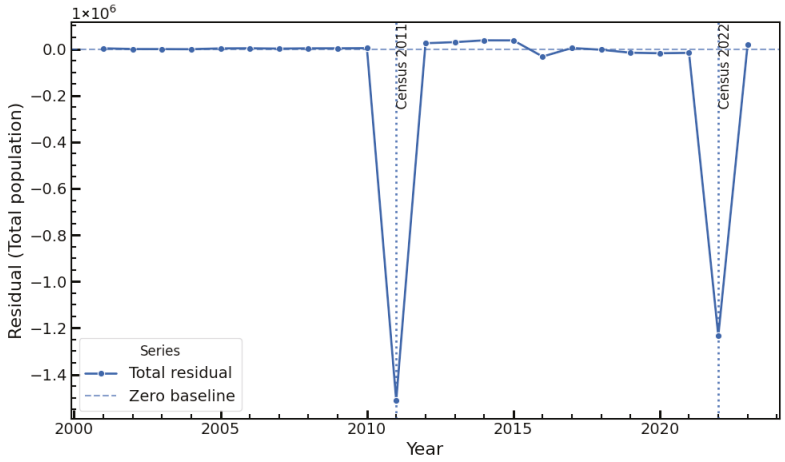


Figure 1. Residual between official population estimates and demographic accounting (inflows − deaths + net migration) for Germany, all ages and both sexes.

Table 1. Estimated decorrelation steps for each country's total population series.

Country	Years Available	Decorrelation Step
Germany	2001–2023	22
Portugal	2008–2024	16
Norway	2001–2024	18

To summarise the structure of the dataset, Table 2 lists all series by category. Population counts serve as the projection target, while demographic components provide fertility, mortality, and migration rates at the cohort level, and socioeconomic indicators supply auxiliary context. Together, these categories define the inputs used throughout the study. While the design adheres to established practices in demographic forecasting, the inclusion of socioeconomic covariates ensures that the dataset reflects both demographic trends and the broader economic context in which they occur. This summary concludes the description of inputs and prepares the ground for the methodological framework presented in the following section.

Table 2. Data series used in the forecasting pipeline.

Series	Category
Population by age group and sex	Projection target
Births	Demographic rate
Deaths	Demographic rate
Immigration	Demographic rate
Emigration	Demographic rate
GDP per capita	Socioeconomic indicator
Unemployment	Socioeconomic indicator
Health expenditure	Socioeconomic indicator

3. Methods

Our forecasting framework is founded on the cohort-component methodology (CCM), the standard demographic tool for projecting populations by age and sex. We expand this traditional structure by incorporating a regression ensemble for rate forecasting, socioeconomic covariates, and event indicators, and implementing smoothing procedures for volatility. Together, these elements form an integrated population projection system that advances cohorts year by year while remaining flexible enough to accommodate external shocks.

3.1. Cohort-Component Projection Core

The population is disaggregated by age group a and sex $s \in \{M, F\}$. Let $P_{a,s,t}$ denote the population of age a, sex s, in the year t. Cohorts evolve according to the update:

$$P_{a,s,t+1} = P_{a-1,s,t} \cdot (1 - q_{a-1,s,t} - e_{a-1,s,t}) + m^{\text{in}}_{a-1,s,t}, \tag{2}$$

where $q_{a-1,s,t}$ is the mortality rate, $e_{a-1,s,t}$ the emigration rate, and $m^{\text{in}}_{a-1,s,t}$ the number of immigrants into that group.

New births replenish the youngest cohorts based on sex-specific age-specific fertility rates (ASFRs):

$$P_{0,F,t+1} = \sum_{a=15}^{49} f^F_{a,t}\, P_{a,F,t}, \tag{3}$$

$$P_{0,M,t+1} = \sum_{a=15}^{49} f^M_{a,t}\, P_{a,F,t}, \tag{4}$$

where $f^F_{a,t}$ and $f^M_{a,t}$ are fertility rates producing female and male births, respectively, from mothers aged a. Fertility rates are computed relative to mid-year exposures, following demographic convention [85].

This core provides the structural backbone of the projection, ensuring demographic consistency. The innovation of our framework lies in how the rates $f^s_{a,t}$, $q_{a,s,t}$, $e_{a,s,t}$, and $m_{a,s,t}$ are themselves forecast.

3.2. Incorporation of Exogenous Factors

To move beyond purely autoregressive dynamics, we allow for external shocks. Our approach follows the logic of time-series models with exogenous regressors—such as ARIMAX and transfer functions [21,22]—and parallels modern machine learning approaches to time-series forecasting [43,47,86].

We define a vector of binary event indicators

$$E_t = (E^1_t, E^2_t, \ldots, E^K_t)^\top, \tag{5}$$

where $E^k_t \in \{0, 1\}$ marks the presence of event type k in the year t. Examples include:

$$E^1_t = \text{Pandemic}_t, \quad E^2_t = \text{Refugee}_t, \quad E^3_t = \text{Economic}_t.$$

We interpret these flags as scenario levers rather than passive covariates. For instance, immigration among 25-year-old males in Germany exhibits sharp peaks in 2015 and 2022 (Figure 2) [87,88], illustrating how refugee crises manifest demographically and motivating the explicit representation of such events.

Three classes of events are considered:

- *Pandemic:* Disruptions to fertility, mortality, and migration during 2020–2021 [89].

- *Refugee crisis:* The Syrian influx of 2015–2016 and Ukraine-related displacement from 2022 onward [87,88].
- *Economic crisis:* The 2008 global financial crisis and subsequent Eurozone downturn, particularly salient in Portugal [90,91].

By manipulating these flags into the future, we can construct counterfactual scenarios—such as recurring pandemics or prolonged recessions—and explore their demographic consequences.

The binary event vector E_t may be generalised to an intensity vector $S_t = (S_{1,t}, \ldots, S_{K,t})^\top$ with $S_{k,t} \in [0,1]$, and a duration or memory term may be defined

$$\widetilde{S}_{k,t} = \sum_{j=0}^{J} \lambda^j S_{k,t-j}, \quad \lambda \in [0,1),$$

which captures multi-year persistence. The same code accepts $S_{k,t} \in [0,1]$ and $\lambda \in [0,1)$, which directly encode severity and persistence. For the reported experiments, we set $S_{k,t} \in \{0,1\}$ and $\lambda = 0$ for robustness with short series, but users can specify alternative S_t and λ to simulate and estimate different intensities and durations.

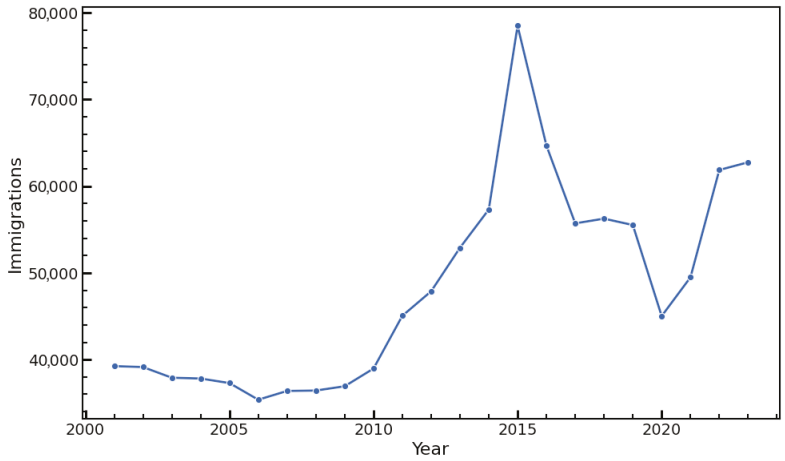


Figure 2. Immigration time series for 25-year-old males in Germany, highlighting spikes in 2015 (the Syrian war) and 2022 (the Ukrainian war).

3.3. EMA-Slope Anchor Under ARIMA-Type Dynamics

We anchor the short-run trajectory of demographic rates using the slope of an exponential moving average (EMA). For a time series $\{y_t\}$ with a smoothing parameter $\alpha \in (0,1)$, the EMA is

$$\mathrm{EMA}_t = \alpha y_t + (1-\alpha)\mathrm{EMA}_{t-1}. \tag{6}$$

We define the EMA-slope as the one-step change:

$$\Delta_t = \mathrm{EMA}_t - \mathrm{EMA}_{t-1}. \tag{7}$$

When $\{y_t\}$ follows an ARIMA(p,d,q) process, the EMA-slope inherits stationarity from the differenced process. In particular, if y_t is ARIMA$(1,0,0)$ with an autoregressive coefficient ϕ, then

$$\mathbb{E}[\Delta_t] = 0, \quad \mathrm{Var}(\Delta_t) = \frac{\alpha^2}{1-(1-\alpha)^2\phi^2}\sigma_\varepsilon^2, \tag{8}$$

where σ_ε^2 is the innovation variance. This illustrates that a smaller α dampens volatility but increases lag, while a larger α yields responsiveness but higher variance. Thus,

the EMA-slope provides a tunable anchor balancing smoothness and adaptability under ARIMA-type dynamics.

If $\{y_t\}$ is covariance-stationary with absolutely summable autocovariances, then the EMA $\{\mathrm{EMA}_t\}$ defined in (6) is also stationary because it is a stable linear filter. The EMA slope $\Delta_t = \mathrm{EMA}_t - \mathrm{EMA}_{t-1}$ is a first difference of a stationary process, hence stationary as well. Under ARIMA(p, d, q), differencing of order d restores stationarity of the filtered series; see [21,83,86]. Error propagation under the cohort recursion follows from a first-order expansion, detailed in Appendix A.

3.4. Stability and Consistency of the Cohort-Component Recursion

The cohort-component recursion can be expressed in vector form as

$$\mathbf{P}_{t+1} = \mathbf{A}_t \mathbf{P}_t + \mathbf{u}_t, \tag{9}$$

where $\mathbf{P}_t$ is the age-by-sex population vector, $\mathbf{A}_t$ encodes survival and migration transitions, and $\mathbf{u}_t$ injects newborns.

Stability requires that the spectral radius of the expected transition operator satisfies

$$\rho\big(\mathbb{E}[\mathbf{A}_t]\big) < 1. \tag{10}$$

This ensures boundedness of long-run trajectories. Moreover, under ergodicity and bounded shock variance, forecasts are consistent in the sense that

$$\lim_{h\to\infty} \frac{1}{h} \sum_{t=1}^{h} \|\hat{\mathbf{P}}_t - \mathbf{P}_t\|^2 \to 0, \tag{11}$$

where $\hat{\mathbf{P}}_t$ denotes forecasts and $\mathbf{P}_t$ the realised populations. Thus, the recursion maintains internal demographic balance while asymptotically stabilising under regularity conditions.

3.5. Error Propagation Under Shock Regimes

Let $\widehat{r}_{a,s,t}$ denote the forecasted rate (mortality, fertility, or migration) and $r_{a,s,t}$ its truth. Define the rate error $\varepsilon_{a,s,t} = \widehat{r}_{a,s,t} - r_{a,s,t}$. Because CCM recursion multiplies survival and fertility rates across successive ages, errors compound multiplicatively:

$$\widehat{P}_{a,s,t+h} \approx P_{a,s,t} \prod_{\tau=1}^{h} (1 - r_{a+\tau,s,t+\tau} - \varepsilon_{a+\tau,s,t+\tau}). \tag{12}$$

Linearising for small errors yields

$$\frac{\widehat{P}_{a,s,t+h} - P_{a,s,t+h}}{P_{a,s,t+h}} \approx -\sum_{\tau=1}^{h} \frac{\varepsilon_{a+\tau,s,t+\tau}}{1 - r_{a+\tau,s,t+\tau}}. \tag{13}$$

Thus, the variance of forecast errors grows approximately linearly in h under independent shocks. However, clustering of shocks introduces temporal correlation, amplifying uncertainty through interaction terms. This motivates the explicit modelling of shock processes rather than treating them as white-noise disturbances.

3.6. Regression Ensemble for Demographic Rates

The regression ensemble forecasts the key inputs to the cohort-component equations. Two categories of targets are distinguished: (i) demographic rates—fertility, mortality,

emigration, and immigration counts—modeled at the cohort level by age and sex, and (ii) socioeconomic indicators, modeled globally.

Let $y_{a,s,t}$ denote a demographic rate and z_t a socioeconomic indicator. Then:

$$z_t = g\big(\{z_{t-l}\}_{l=1}^{L}, E_t\big), \tag{14}$$

$$y_{a,s,t} = g\big(\{y_{a,s,t-l}\}_{l=1}^{L}, \{z_t\}, E_t\big), \tag{15}$$

where L is the maximum lag determined via decorrelation analysis.

The regression function $g(\cdot)$ is defined as an ensemble of two components: a Huber regressor, robust to outliers, and an exponential moving average (EMA), which anchors long-term trends. The Huber component solves

$$\hat{\beta} = \arg\min_{\beta} \sum_{t} \rho_{\delta}\big(y_t - x_t^{\top}\beta\big), \tag{16}$$

with ρ_{δ} the Huber loss. The EMA component extrapolates smoothed trends using a rolling regression on recent EMA values. The ensemble forecast is then

$$g(x_t) = w\, g^{\text{Huber}}(x_t) + (1-w)\, g^{\text{EMA}}(x_t), \tag{17}$$

with w determined by out-of-fold one-year-ahead RMSE. This weighting ensures that the ensemble adapts: leaning on the Huber regressor when covariates improve short-horizon accuracy, and on EMA when long-run momentum dominates.

Conceptually, EMA stabilises the trajectory, while the Huber component responds to short-term signals and shocks. Hyperparameters, including the Huber threshold δ, are tuned using the same OOF validation. As an additional safeguard, forecasts that exhibit implausible volatility are smoothed using an exponentially weighted correction [25,86].

Unlike in standard applications, where EMA is used to smooth levels, here we use the EMA slope as a forecasting baseline. This provides a transparent and adaptive estimate of long-run momentum, ensuring that forecasts are anchored to a plausible trajectory even when short-run predictors are noisy or underperform. To our knowledge, applying the EMA slope in this role is novel in demographic forecasting.

Lagged predictors are included up to five years, based on the decorrelation analysis presented in the Data section. This yields parsimonious autoregressive structures consistent with the limited annual data available.

It should be noted that the ensemble weight w is based on OOF one-year-ahead RMSE to prioritise stability in short annual series. Longer-horizon validation would be desirable but cannot be supported reliably when the available history spans only a few decades. By basing the weight on short-run predictive skill, the ensemble avoids overfitting structural trends that are weakly identifiable in sparse data. Similarly, the fixed decorrelation lags are designed as pragmatic approximations that maintain numerical stability across components. We acknowledge that fertility, mortality, and migration exhibit distinct autocorrelation patterns, and that more flexible lag structures or Monte Carlo sensitivity analysis represent valuable methodological extensions. Incorporating such refinements is identified as an important avenue for future work, particularly when richer datasets become available.

Given data sparsity, cross-method numerical comparisons can be fragile and sensitive to specification choices and leakage. We therefore report internal performance via out-of-fold error and use the ensemble's two components (EMA-only and Huber-only) as conceptual baselines. Table 3 reports quantitative errors for key series; fuller head-to-head baselines are deferred to future work with longer histories.

Table 3. Mean one-year-ahead normalised RMSE (nRMSE) of rate models by country (2001–2024).

Rate	Germany	Portugal	Norway
ASFR (female)	0.145	0.120	0.171
ASFR (male)	0.148	0.125	0.159
Immigrations	0.257	0.340	0.179
Emigration rate	0.228	0.194	0.315
Mortality rate	0.150	0.206	0.179
GDP per capita	0.125	0.452	0.277
Unemployment	0.054	0.033	0.104
Health exp. (% GDP)	0.163	0.159	–

3.7. Population Projection Algorithm

The full forecasting pipeline proceeds iteratively as follows:

1. *Initialization:* Start from the observed population by age and sex, P_{a,s,t_0}, in the last available year t_0.
2. *Scenario design:* Define event trajectories E_t for the forecast horizon, encoding assumptions about exogenous shocks.
3. *Rate model training:* Estimate the regression ensembles for demographic rates and socioeconomic indicators using lagged histories, covariates, and event flags.
4. *Rate forecasting:* Produce annual forecasts for each rate, combining Huber and EMA components with RMSE-based weighting and volatility smoothing.
5. *Population updating:* Substitute forecasted rates into the cohort-component equations, advancing cohorts and generating new births.
6. *Iteration:* Repeat the process sequentially for each forecast year until the end of the horizon, yielding age-structured population trajectories under the specified scenarios.

This workflow ensures projections remain internally consistent with demographic accounting, responsive to socioeconomic and event-driven signals, and resilient to noise and volatility.

The purpose of the proposed framework is not to claim high-precision point forecasts at such horizons. Instead, the central outcome of the study is the development of a transparent, scenario-conditioned mechanism for *stress-testing* demographic futures under explicit assumptions.

The methodological contribution lies in showing that CCM-based demographic processes can be combined with robust regression components and scenario-sensitive flags to produce conditional trajectories that respond in controlled ways to shocks such as pandemics, refugee inflows, or economic downturns. The resulting forecasts should therefore be interpreted as *conditional projections*: they map each explicitly stated scenario to a corresponding demographic pathway. This differs fundamentally from attempts to produce single "best" long-term forecasts, which are known to be dominated by structural uncertainty arising from volatile components such as migration.

By highlighting the limits of long-term point accuracy, the paper follows established demographic practice and emphasises that uncertainty is an inherent property of medium- and long-run population dynamics. The contribution of the proposed framework is thus not to eliminate this uncertainty, but to make it explicit, structured, and reproducible. This allows researchers and policymakers to evaluate how alternative assumptions propagate through fertility, mortality, and migration pathways, and to compare the robustness of outcomes across scenarios. In this sense, the ultimate aim of the paper is clarity and transparency in demographic stress testing, rather than precise deterministic forecasts.

Having established the forecasting framework and its components, we now move to an empirical evaluation of its performance. The results (in the following section) show how the model performs across the three country contexts, assessing both the accuracy of short-term forecasts and the plausibility of long-term projections under various scenarios. In doing so, we emphasise the practical value of integrating socioeconomic covariates and event indicators into a cohort-component projection system.

4. Results

Given the limited availability of annual demographic time series from official sources, we employ a time-series regression ensemble to project future rates. The ensemble automatically selects lags of past values alongside global predictors, with recent lags strengthened to decrease sensitivity to shocks. A fallback smoothed-trend mechanism guarantees stable forecasts when the ensemble performs poorly. This method balances flexibility and simplicity, delivering reliable rate estimates from short historical records.

A natural question is whether demographic projections could be produced at a higher temporal resolution, such as monthly, in order to capture short-run volatility in migration, births, or deaths. However, two structural constraints make this infeasible. First, harmonised monthly demographic series do not exist for the countries examined, nor for most official statistical systems; where monthly releases are available, they are typically incomplete, non-harmonised, or subject to substantial revision.

Second, and more fundamentally, the decorrelation analysis shows that demographic rates lose statistically meaningful temporal dependence after approximately 20–25 years of annual data. This implies that lags beyond this horizon contain no predictive signal that can be exploited reliably by any forecasting model. In other words, the intrinsic memory of demographic processes is short. Even if monthly data were available, their high-frequency fluctuations would not correspond to additional predictive structure, but rather to noise that cannot be estimated or validated, given the identified decorrelation limits.

These structural properties also constrain the validation strategy. When the informative temporal window is limited by decorrelation rather than by data quantity, annual out-of-fold evaluation becomes the only statistically defensible approach: it preserves independence between training and validation intervals, while avoiding the overfitting that would arise from attempting to learn from non-informative or decorrelated lags. More complex validation schemes or artificially upsampled high-frequency series would violate these identifiability constraints and give a false impression of forecasting precision.

For these reasons, the proposed framework focuses on transparent, scenario-conditioned annual projections. The aim is not to model speculative month-to-month dynamics, but to provide a reproducible mechanism for assessing the impact of explicit crisis assumptions on demographic trajectories, within the limits set by the intrinsic temporal structure of demographic processes.

4.1. Model Fit and Predictive Accuracy

To evaluate model performance, the ensemble is trained on historical data between 2001 and 2024 and validated using a one-year-ahead out-of-fold (OOF) approach. This simulates the real forecasting process, thus providing a realistic assessment of predictive accuracy. Then, the model is used to make predictions under different scenarios through 2040. The forecast horizon up to 2040 is constrained by the decorrelation analysis, which indicates that temporal correlations in the data effectively vanish beyond approximately 20–25 years.

Table 3 reports the mean normalised RMSE (nRMSE) for the Huber regression component, which anchors the short-horizon forecast. The exponential moving average (EMA) slope component is not optimised for OOF accuracy but instead provides long-term trend stability. Across countries, fertility rates (ASFR) show comparatively low errors (0.12–0.17), followed by unemployment (0.03–0.05). Mortality rates and health expenditure fall within an intermediate range (0.15–0.21). Migration-related series exhibit higher error levels, with immigration reaching 0.34 and emigration reaching 0.32. GDP per capita is moderately accurate for Germany and Norway, but substantially noisier in Portugal (0.45).

Lag selection typically includes three to five annual lags, with five as the maximum, due to the limited length of official demographic series (15–20 observations per cohort). This provides sufficient temporal depth without leading to overfitting. Emphasising the most recent lags further stabilises forecasts during shocks, while the fallback trend mechanism ensures sensible continuation if the ensemble fails. Overall, these design features improve both reliability and interpretability.

For Norway, health expenditure data were excluded because the available series is outdated and not comparable with Germany and Portugal. When included, its predictive accuracy falls within the same error range as fertility and unemployment.

4.2. Scenario Design

To examine the ensemble's behaviour under different future scenarios, we establish a series of stylised "what-if" cases using binary event flags. Each scenario introduces a hypothetical shock between 2028 and 2030, targeting one event type at a time. This allows for structured counterfactual experiments that are uncommon in demographic forecasting, where structural shocks are often omitted or only modelled retrospectively.

While the event flags are explicitly included as exogenous predictors in the regression ensembles, their role is similar to that of intervention variables in ARIMA-type models: they influence rate forecasts only when historical data provides a clear signal to learn from. In practice, this means that activating a flag will not create large discontinuities in the series without prior evidence. Instead, the ensemble utilises past structural shocks (e.g., refugee surges in Germany) to interpret future flag activations as plausible deviations, while reverting to the baseline in smoother contexts where shocks have been infrequent or muted.

The modelled scenarios are:

Baseline: No crises after 2024; all event flags remain off. This represents a smooth continuation of recent trends.

Pandemic: A new pandemic emerges in 2028–2030, activating only the pandemic flag.

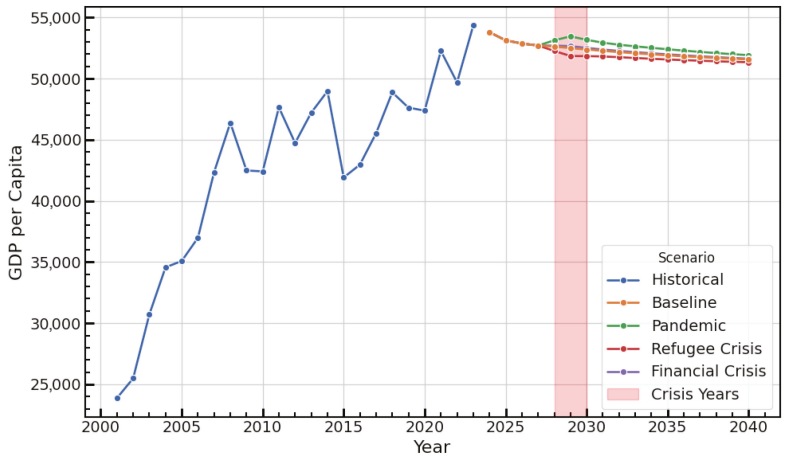
Refugee Crisis: A geopolitical conflict in 2028–2030 produces a surge in immigration, activated through the refugee crisis flag.

Financial Crisis: A sharp economic downturn in 2028–2030 is simulated by enabling only the economic crisis indicator.

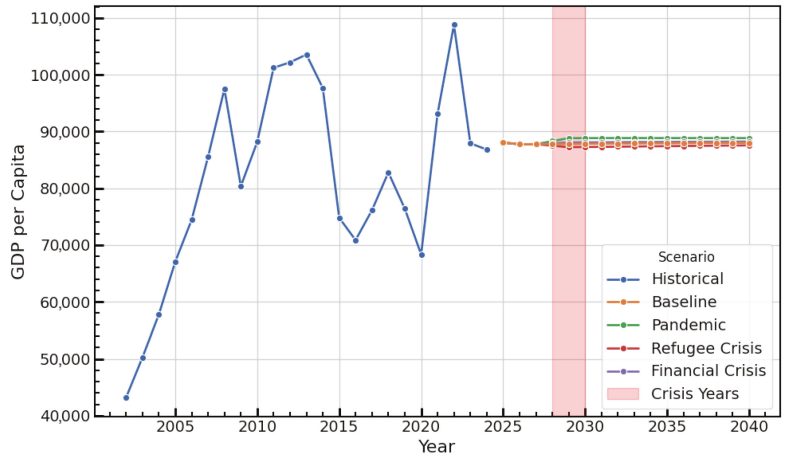
Applied to the rate ensemble and propagated forward via the CCM population equation, these shocks affect fertility, mortality, and migration, as well as the global predictors, before resulting in population outcomes. They are not meant as forecasts, but as stress tests of plausible extremes. Unlike standard extrapolations, this approach enables policy-relevant stress testing of demographic futures, bridging quantitative forecasting with the more commonly used scenario-based reasoning in qualitative foresight. Note that scenario curves are close when historical signals linking crises to different shocks were weak or irregular, consistent with the ensemble's conservative design.

4.3. Scenario Effects on Key Demographic Rates

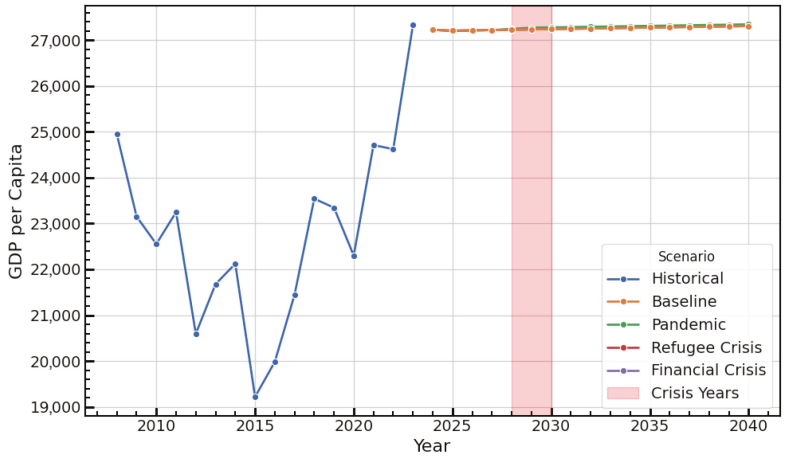
The repository of all past data for the figures presented in this section is provided in the Data Availability Section; here, we focus on interpreting their implications for model behaviour and scenario responses. We begin with global predictors, starting with GDP per capita (Figure 3).



(**a**) Germany



(**b**) Norway



(**c**) Portugal

Figure 3. Historical and projected GDP per capita under baseline and alternative scenarios. Shaded area shows the crisis period (2028–2030).

Across all three countries, responses to the scenario are subdued. The autoregressive models capture general baseline trends but show little differentiation between scenarios, suggesting limited recurring patterns in GDP per capita for the ensemble to utilise. Germany's historically volatile series stabilises after 2025, with only minor deviations under different crises. Norway stabilises quickly, with scenario trajectories essentially indistinguishable. Portugal, despite previous fluctuations, also converges closely to the baseline. In this case, scenario switches contribute little beyond trend continuation, and deviations are mostly superficial.

The limited responsiveness of GDP per capita to scenario flags underscores a vital distinction: when historical shocks do not leave lasting statistical marks, the model defaults to baseline trajectories. This aligns with the ensemble's design but also implies that GDP dynamics need a broader set of predictors beyond autoregressive lags and binary event indicators. Specifically, adding sectoral or macroeconomic covariates (such as investment rates, trade balances, or fiscal shocks) could enhance scenario differentiation, whereas the current setup views GDP mainly as a smooth continuation process. The appearance of linearity should therefore be interpreted as a data-driven consequence of sparse annual observations rather than as an indication of insufficient model flexibility.

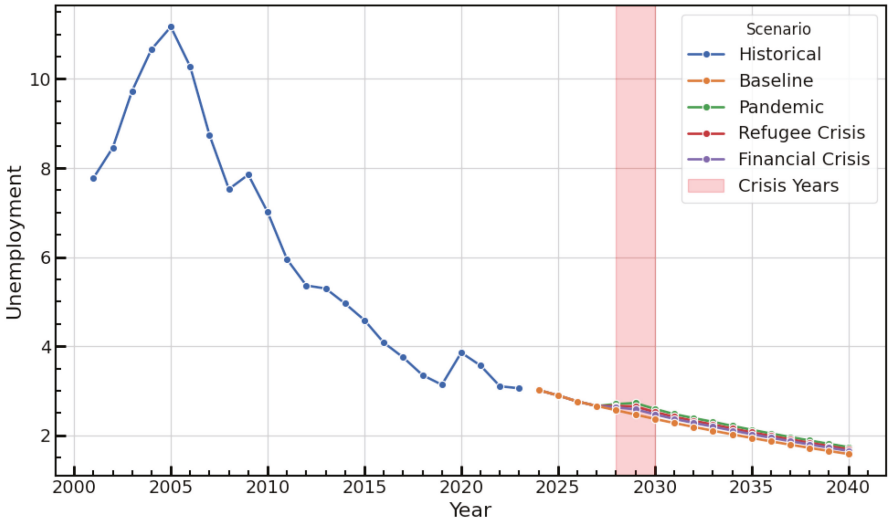
Unemployment (Figure 4) shows a similar pattern. Since historical crises have not caused significant, visible spikes in unemployment, scenario levers have little effect. In Germany, the ensemble reflects the decline from double digits in the early 2000s to below three per cent today, continuing downward towards 1.5–2% by 2040. Norway remains steady around 3.5–3.7%. Portugal, in contrast, is expected to gradually rise back to 7–8% by 2040 after the steep decline post-2013. Once again, historical trends determine the strength of the projected scenario effects.

Health expenditure as a share of GDP offers more interpretable responses to scenario changes (Figure 5). In Germany, spikes such as the 2020 surge are replicated when the pandemic indicator is activated, leading to a sustained increase above the baseline. Other shocks tend to stay closer to the trend. This alignment shows that the ensemble can translate external scenario switches into plausible deviations when there is historical precedent. Portugal's projections are more subdued, as the financial crisis occurs near the start of its data, which limits broader applicability. Norway is excluded due to the absence of comparable data series.

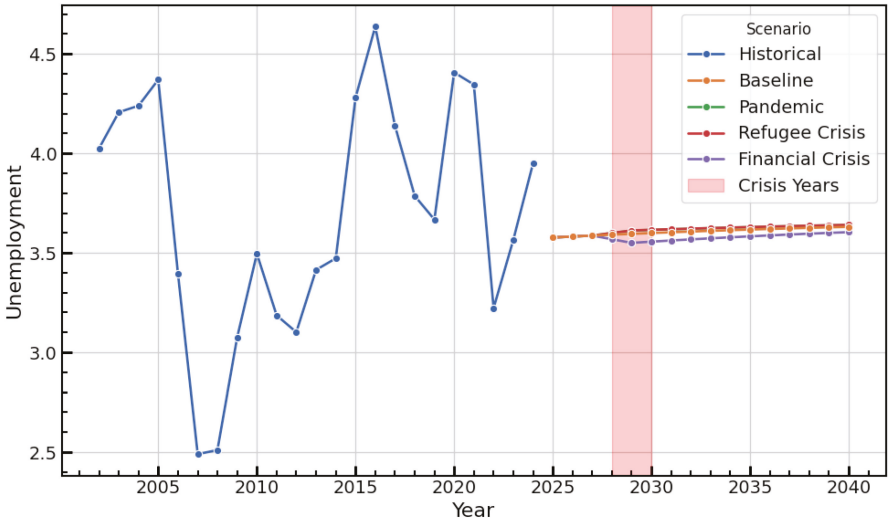
Having considered global predictors, we now examine cohort-specific rates. While global variables inform these models, we do not interpret their individual coefficients, as doing so would require inspecting numerous specifications. Instead, we rely on Huber regression's feature selection to robustly identify relevant predictors.

Male age-specific fertility rates (ASFRs) are shown in (Figure 6). In Germany, scenario trajectories clearly diverge: refugee and pandemic crises push fertility above the baseline, while financial turmoil slightly lowers it. Norway exhibits flatter projections, as its historical pandemic-related fertility rise was too modest to register as a structural shock. Portugal shows the opposite: the model replicates the 2015 fertility rise in the refugee crisis scenario, while long-term trajectories revert to the trend.

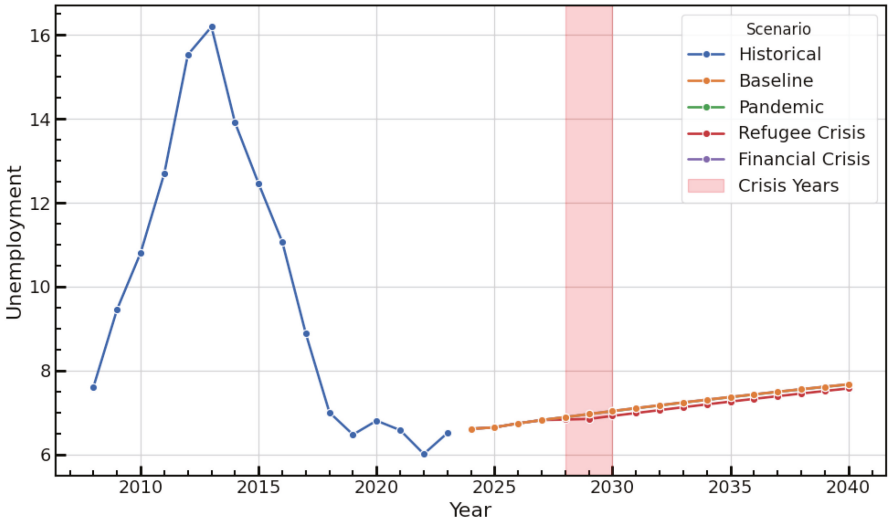
Female ASFRs (Figure 7) follow nearly identical dynamics, as expected given the joint nature of fertility outcomes. Both sexes respond similarly to the scenarios, confirming the model's internal coherence.



(**a**) Germany

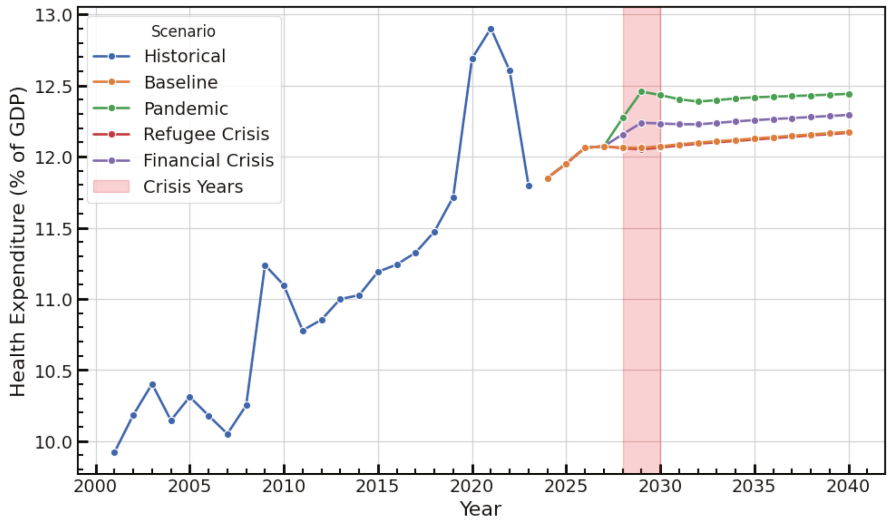


(**b**) Norway

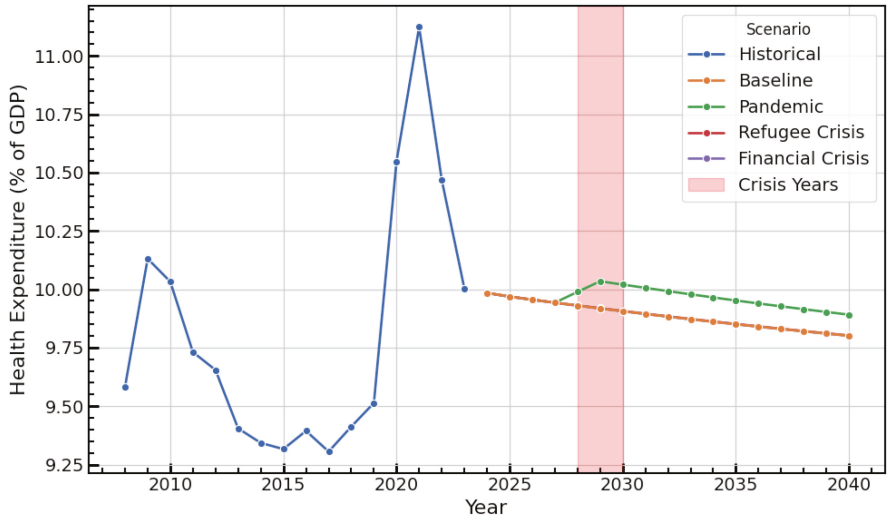


(**c**) Portugal

Figure 4. Historical and projected unemployment percentage under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).



(**a**) Germany



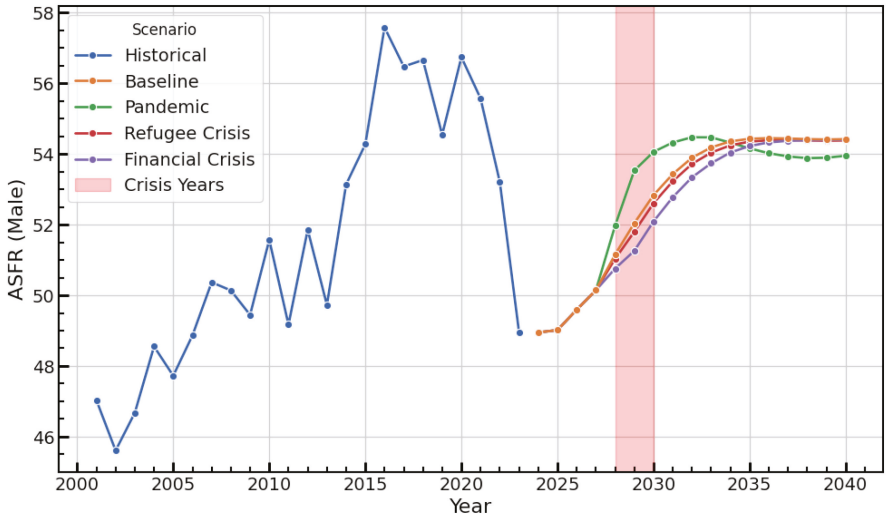
(**b**) Portugal

Figure 5. Historical and projected health expenditure as a share of GDP under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).

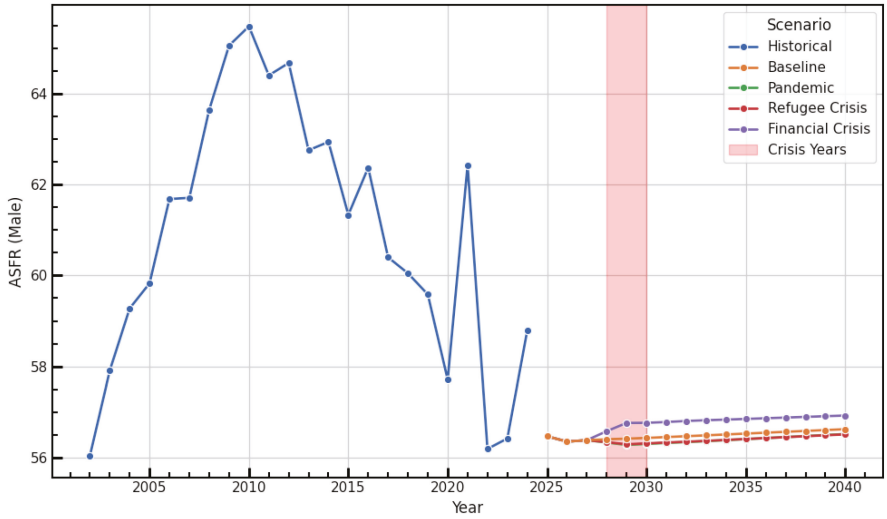
Mortality among older female cohorts (Figure 8) illustrates the same principle. In Germany, the downward trend persists with modest divergence across scenarios. Norway's series is noisier, with little scope for significant deviations. Portugal's mortality continues to decline steadily, with scenarios introducing only slight adjustments.

Immigration flows (Figure 9) demonstrate how strongly the model responds to significant historical shocks. Germany's 2015 spike is reflected in the refugee crisis scenario, resulting in a sustained upward shift. Norway's notable peak is only partly activated, as it did not surpass global thresholds. Portugal shows no clear separation, indicating its smoother pattern history.

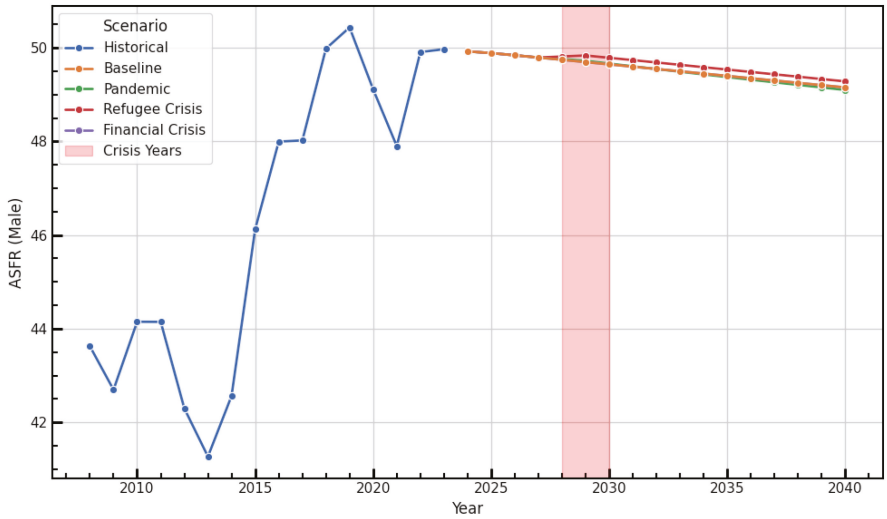
Emigration (Figure 10) reinforces this pattern. Germany experiences both financial crisis surges and refugee-related spikes, with the pandemic scenario dampening mobility. Norway exhibits modest increases during crises, but trajectories remain close to baseline. Portugal exhibits flat paths, reflecting limited historical precedent.



(**a**) Germany, Age 30

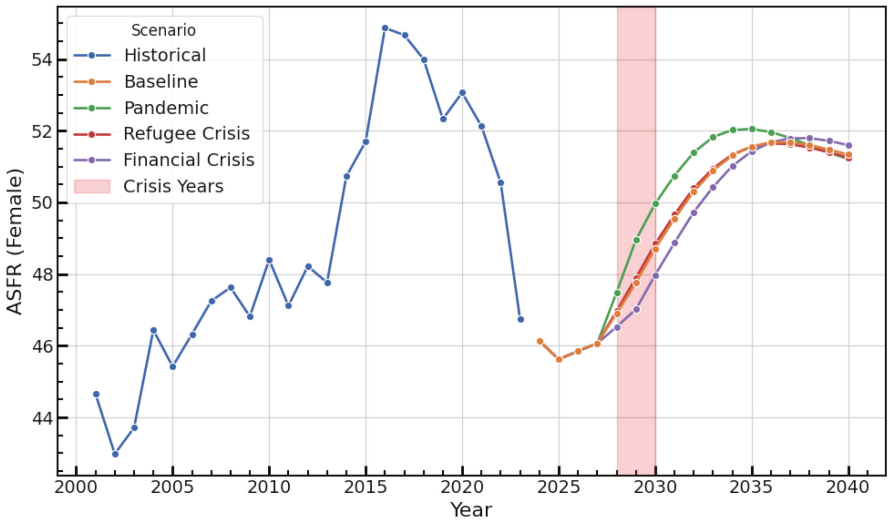


(**b**) Norway, Ages 30–34

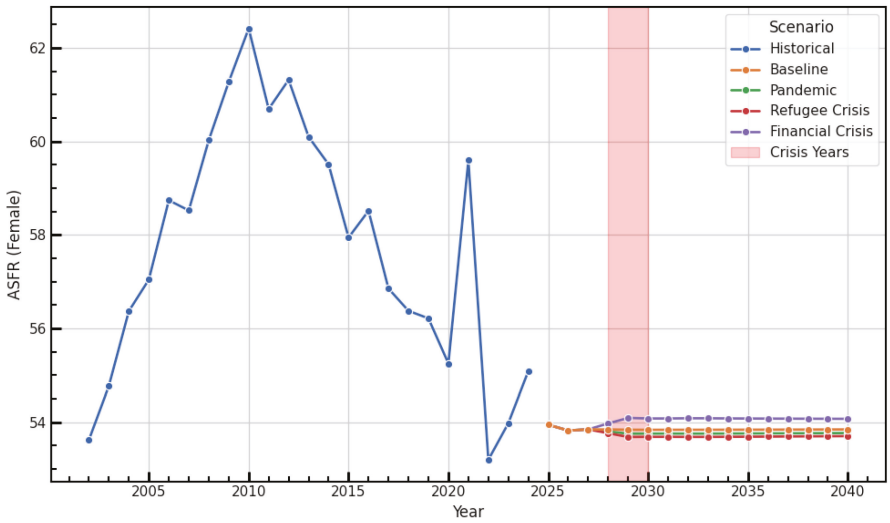


(**c**) Portugal, Ages 30–34

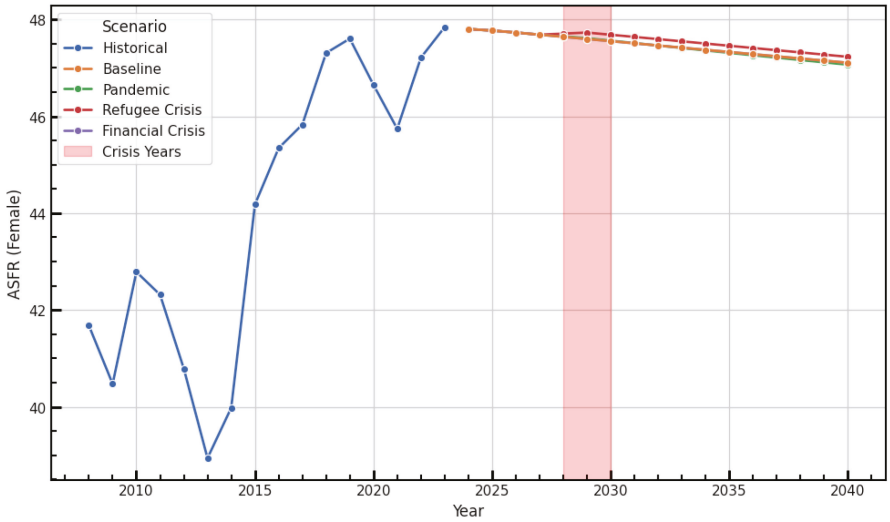
Figure 6. Historical and projected age-specific fertility rates (ASFRs, male) under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).



(**a**) Germany, Age 30

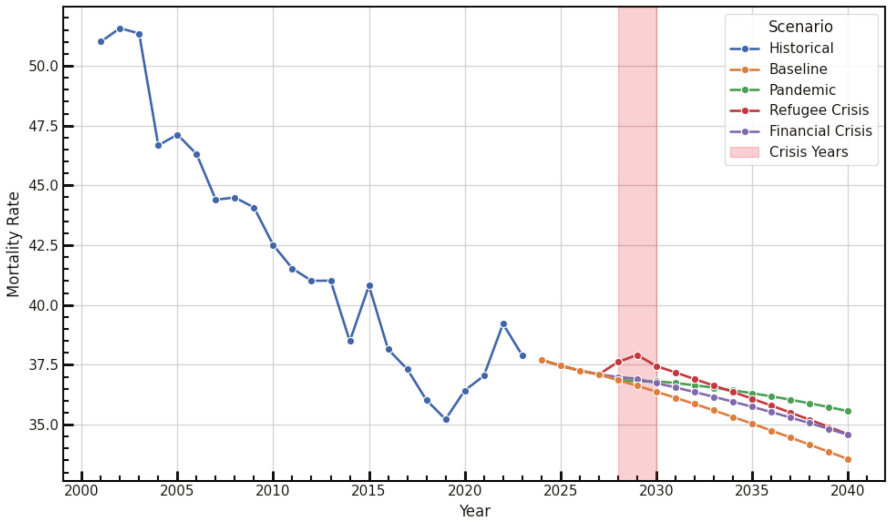


(**b**) Norway, Ages 30–34

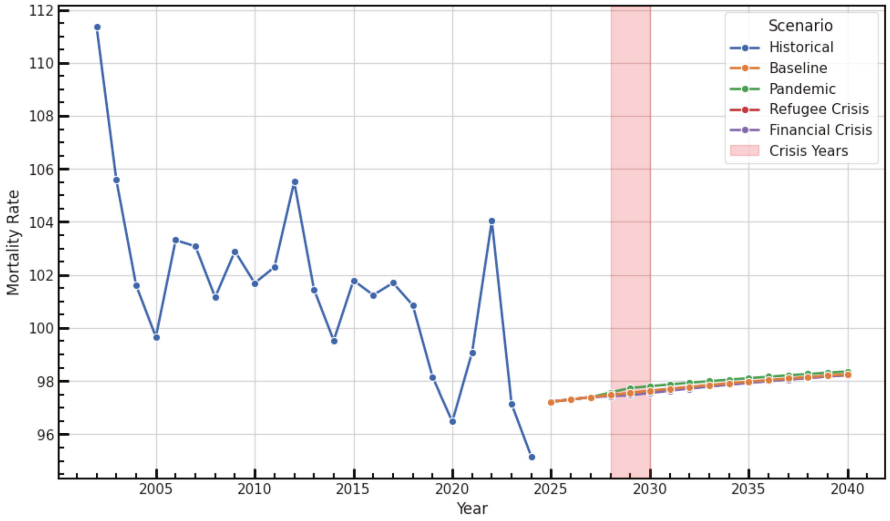


(**c**) Portugal, Ages 30–34

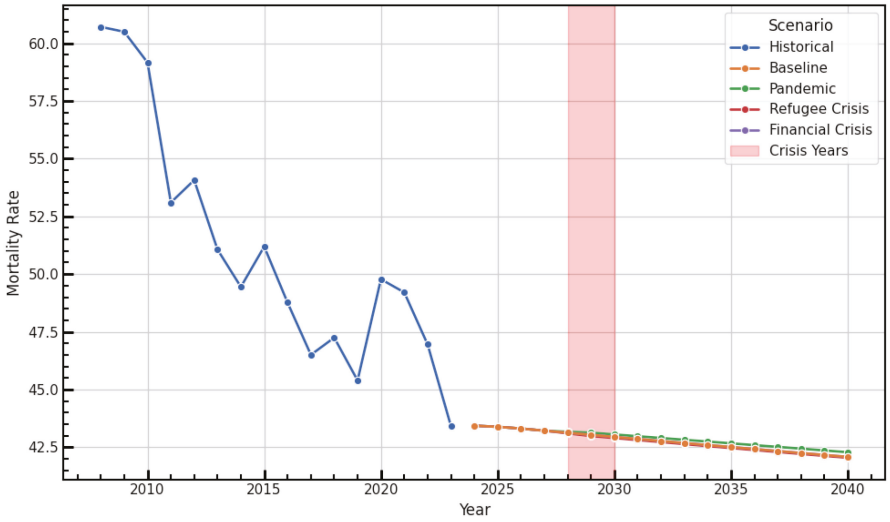
Figure 7. Historical and projected age-specific fertility rates (ASFRs, female) under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).



(**a**) Germany, Age 80 Females

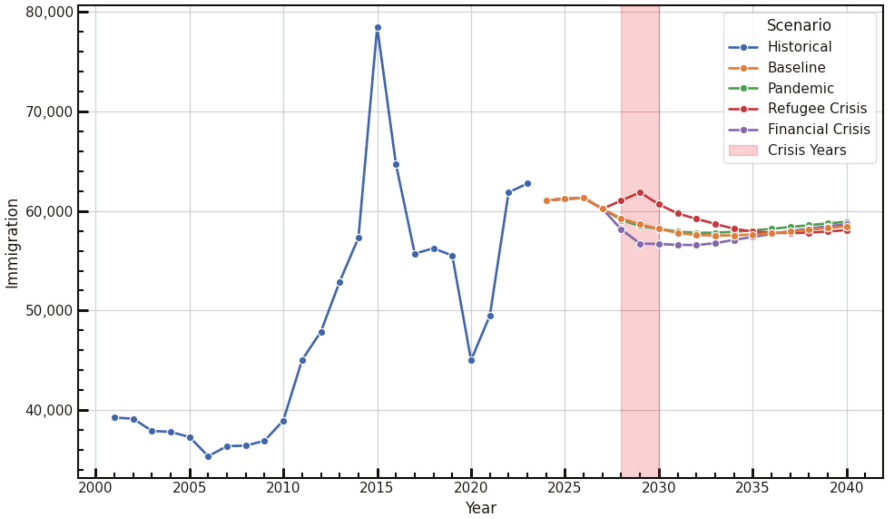


(**b**) Norway, Age 80+ Females

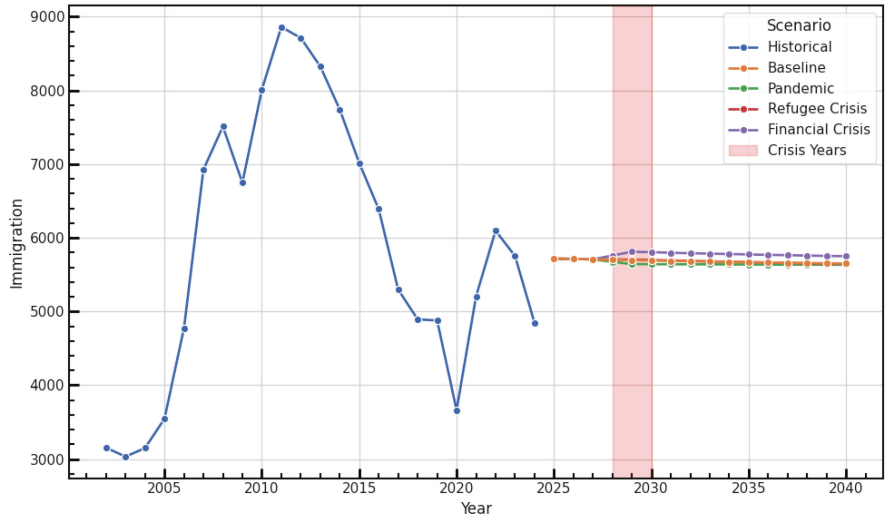


(**c**) Portugal, Ages 80–84 Females

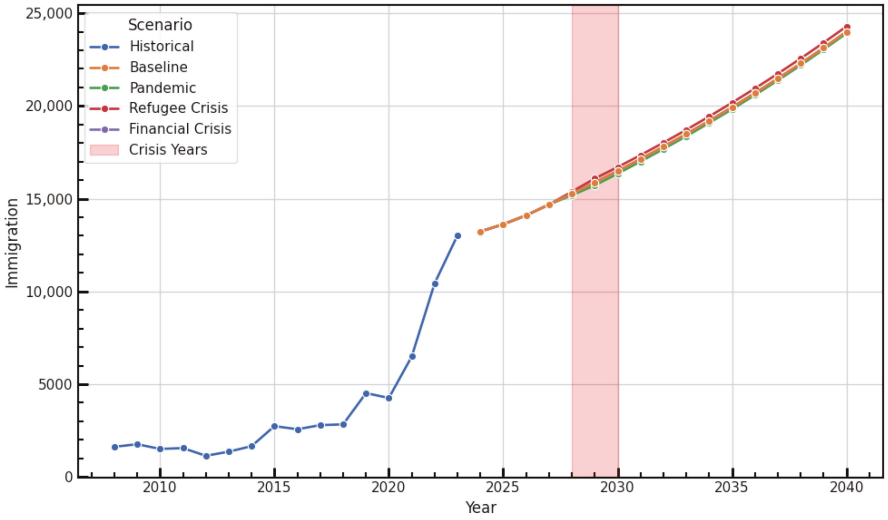
Figure 8. Historical and projected mortality rates under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).



(**a**) Germany, Age 25 Males

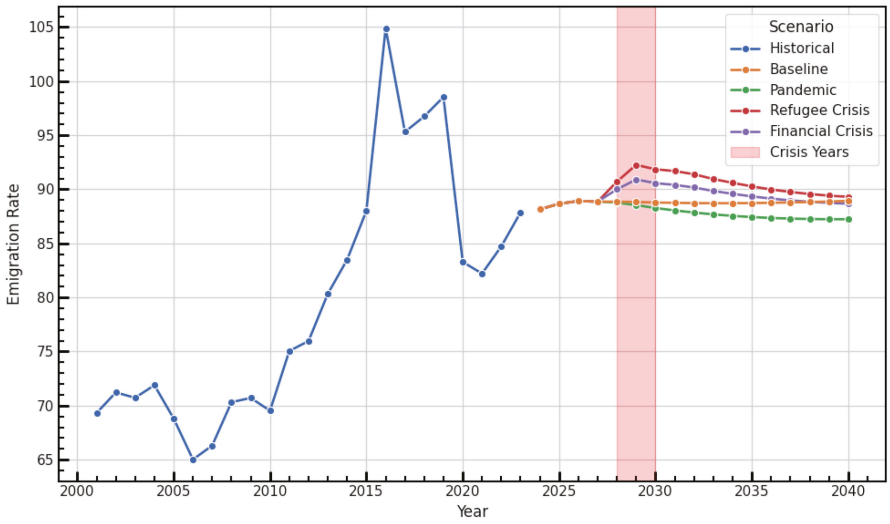


(**b**) Norway, Ages 25–29 Males

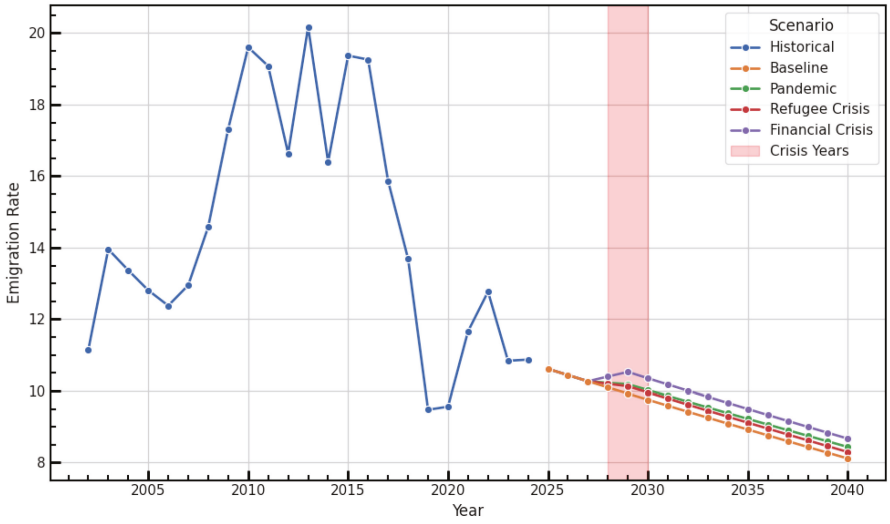


(**c**) Portugal, Ages 25–29 Males

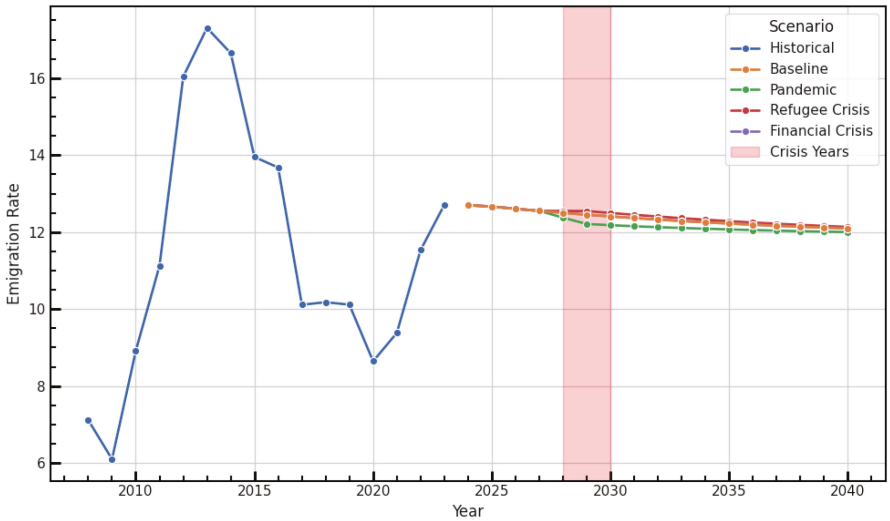
Figure 9. Historical and projected immigration flows under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).



(**a**) Germany, Age 25 Males



(**b**) Norway, Ages 25–29 Males



(**c**) Portugal, Ages 25–29 Males

Figure 10. Historical and projected emigration rates under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).

4.4. Population Projections

Finally, we aggregate cohort-specific rates into population outcomes.

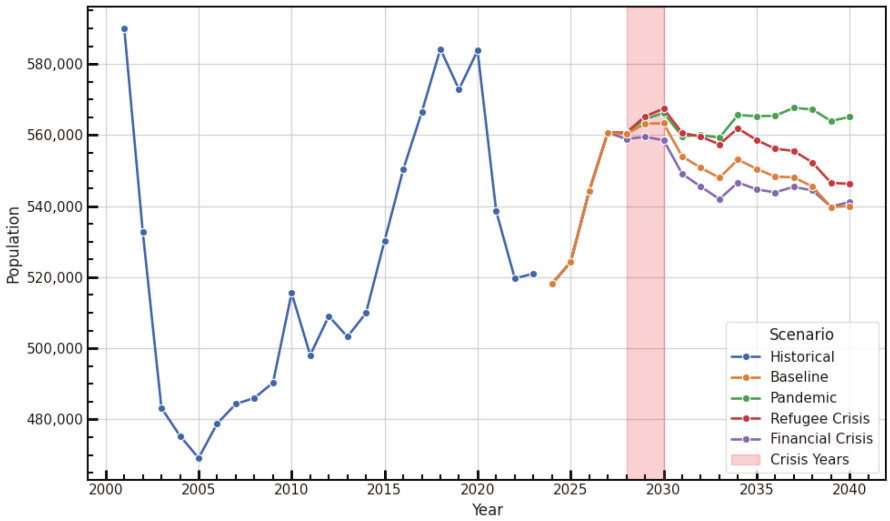
Figure 11 shows selected cohort projections. In Germany, shocks build up into lasting divergences: refugee crises raise levels through immigration, pandemics decrease emigration and also elevate trajectories, while financial crises lead to increased emigration and shift paths downward. Since these shocks influence flows, not just stocks, their impacts endure over time. Norway and Portugal, by contrast, exhibit subdued differentiation. Their scenarios largely mirror the baseline, reflecting the lack of significant past shocks in their series. Still, baseline demographic momentum remains well captured.

Total population outcomes are presented in Figure 12. The logic remains similar: in Germany, scenarios diverge more sharply, especially during the refugee crisis, leading to a noticeable long-term increase. Norway exhibits modest differences, aligning with its stable demographic structure. Portugal stands out with a steep projected rise, likely due to its recent immigration, as reflected in the rate models. Here, baseline momentum primarily influences the outcomes, with scenario variation having only a secondary effect.

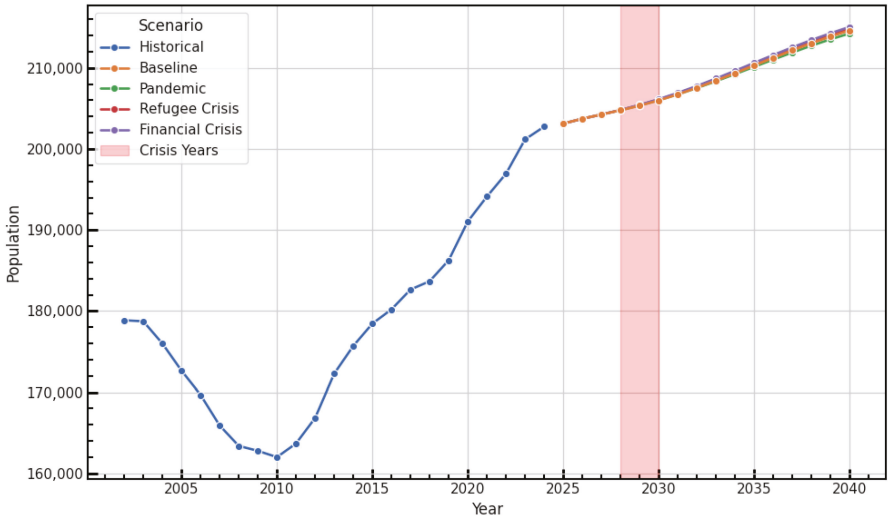
To illustrate how scenario flags lead to different futures, we present an overall visualisation for Germany (Figure 13). The main line shows the median of all scenario projections, while the shaded area covers the lowest and highest values. This demonstrates how different crisis flags cause varying effects on long-term population outcomes, emphasising points where scenario design results in significant divergence. In this way, we depict model uncertainty through the range of plausible scenario-driven outcomes.

The magnitude of scenario-induced deviations depends inherently on the strength and persistence of historical signals in each demographic component. When past behaviour exhibits limited volatility, the ensemble naturally produces conservative adjustments that remain within historical variation. This behaviour is by design: it ensures that the model does not extrapolate implausible crisis responses in components for which no strong historical shocks exist. Conversely, where historical shocks are well-documented—such as in Germany's migration series—the model produces more pronounced scenario-induced deviations. This feature reflects a fundamental principle of scenario-conditioned demographic modelling: responsiveness is anchored to known behavioural precedents, and caution is exercised when extrapolating beyond historically observed dynamics.

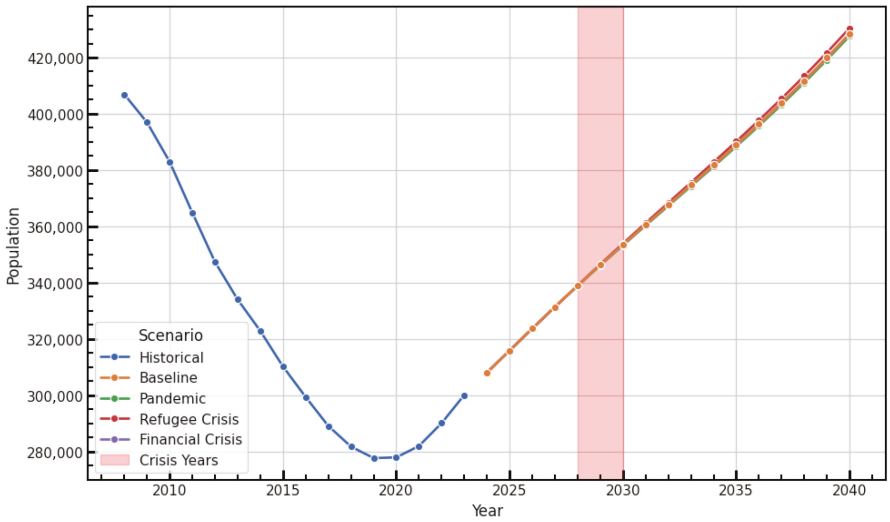
Taken together, the results show that the forecasting framework reliably reproduces recent demographic dynamics and translates historically grounded shocks into plausible future deviations. Scenario responsiveness is strongest where past crises left clear statistical traces, such as migration in Germany, and weakest where series are smoother or shorter, as in Portugal and Norway. Importantly, the ensemble balances stability and flexibility: it avoids overreaction in domains with little historical precedent, while preserving the capacity to project persistent effects of large shocks. The final section builds on these findings by examining their broader implications, assessing how methodological choices shape interpretability, and reflecting on the limits and policy relevance of scenario-based demographic forecasting.



(**a**) Germany, Males age 30

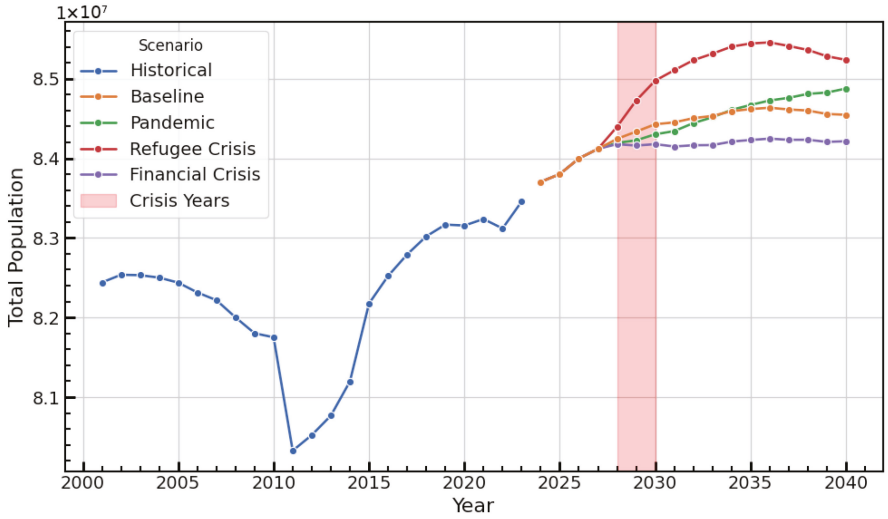


(**b**) Norway, Males age 30–34

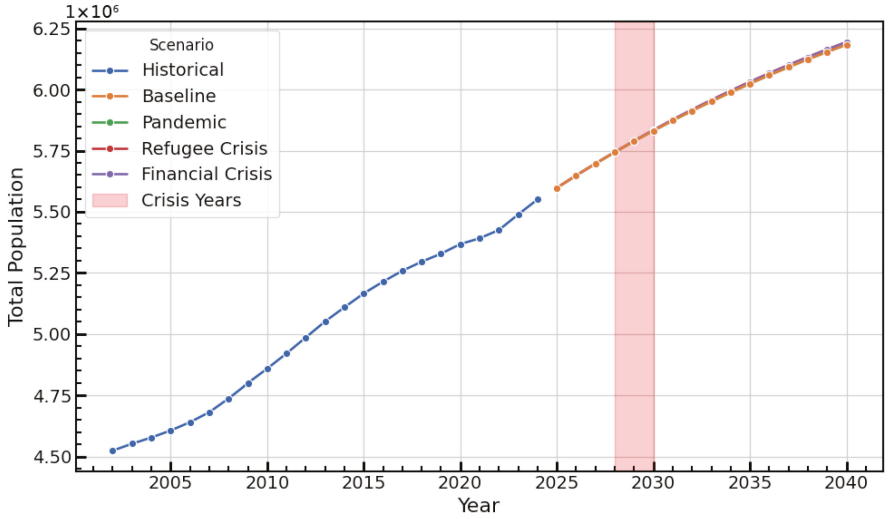


(**c**) Portugal, Males age 30–34

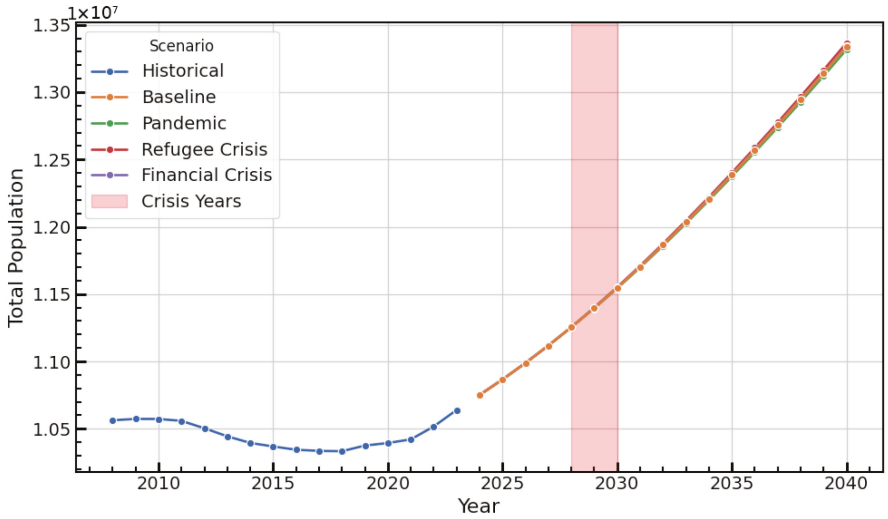
Figure 11. Historical and projected population counts for selected male cohorts under baseline and alternative scenarios. The shaded area denotes the crisis period (2028–2030).



(**a**) Germany, Total Population



(**b**) Norway, Total Population



(**c**) Portugal, Total Population

Figure 12. Historical and projected total population under baseline and alternative scenarios. The shaded area indicates the crisis period (2028–2030).

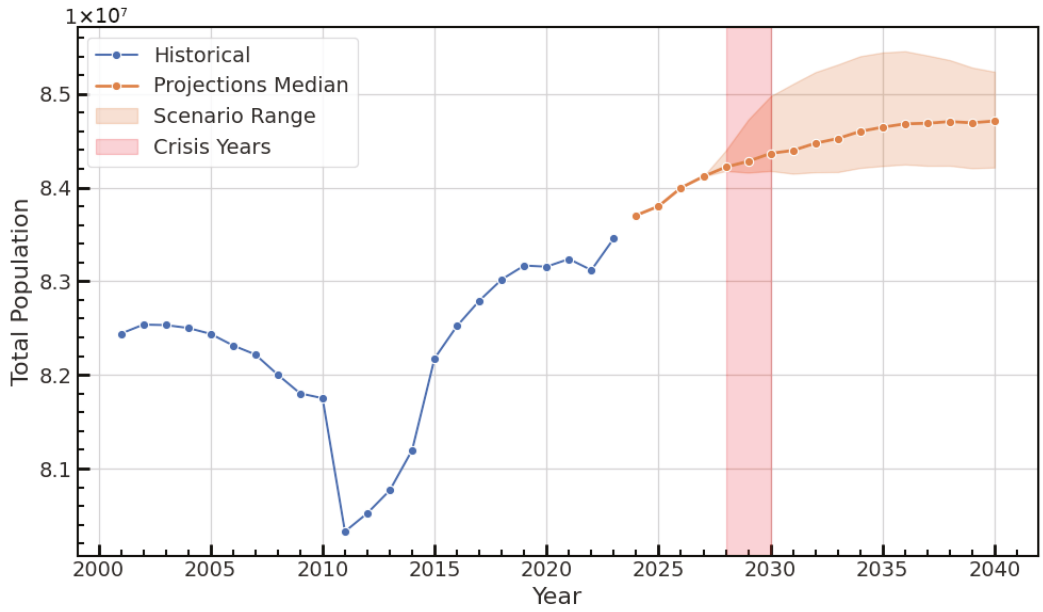


Figure 13. Historical and projected total population in Germany under alternative scenarios. The solid line indicates the median of all scenario projections, while the shaded area covers the minimum and maximum values. The shaded red area highlights the crisis period (2028–2030).

5. Conclusions

This paper has presented a scenario-conditioned population forecasting framework that joins a standard cohort-component projection core with a simple regression ensemble for demographic rates. The key feature of the approach is that population futures are expressed as conditional statements: each forecast is tied to explicit assumptions about the occurrence of shocks such as pandemics, refugee inflows, and financial crises. Instead of extending past trends in an unqualified way, the framework embeds these scenario switches in the rate models and then propagates their effects through age- and sex-structured cohort-component projections.

Methodologically, the framework combines three elements. First, fertility, mortality, and migration rates are modelled through a robust Huber regression that uses short autoregressive histories, socioeconomic covariates, and binary event indicators. This allows the models to absorb information from past shocks while limiting the influence of outliers. Second, an exponential moving-average slope is used as a trend anchor, providing a simple and transparent way to stabilise long-run trajectories when data are sparse and highly autocorrelated. Third, lag selection is guided by decorrelation analysis, which limits the number of annual lags and avoids overfitting when only 15–20 observations are available for each cohort. Together, these choices keep the forecasting machinery deliberately modest, in line with the information content of official annual series.

The empirical application to Germany, Norway, and Portugal illustrates how the framework behaves in practice. Across these three cases, the model reproduces recent dynamics of fertility, mortality, and migration reasonably well and generates annual projections to 2040 under a baseline and three stylised crisis scenarios. A consistent pattern is that scenario effects are more pronounced where historical shocks are strong and recurrent, and weaker where past series are relatively smooth. For instance, Germany's migration history includes clearly visible peaks associated with the Syrian refugee crisis and the war in Ukraine; when the refugee scenario is activated, the ensemble translates these precedents into higher future immigration, which in turn lifts specific working-age cohorts and the

total population compared with the baseline. In Norway and Portugal, where shocks in the observed period have been smaller or less regular, the scenario paths tend to remain closer to the baseline, and the framework behaves more like a cautious trend extrapolator.

Another important finding is that the impact of scenarios depends on the level of aggregation. National population totals tend to absorb short-lived or offsetting shocks in fertility, mortality, and migration, so the differences between scenarios are often modest at this scale. Age-structured projections reveal sharper divergences, especially for cohorts directly exposed to migration surges or crisis-related changes in mortality. This is consistent with demographic intuition and suggests that the framework can be particularly useful for applications that rely on the composition of the population by age and sex, such as labour-market planning or health-care demand, rather than only on total counts.

The study also highlights the constraints imposed by short and imperfect data. With roughly two decades of annual observations, the temporal memory that can be exploited in a statistically meaningful way is limited. This motivates the use of robust regression and simple trend anchoring, rather than more complex machine learning architectures that would require longer time series or high-frequency data. It also affects the treatment of macroeconomic variables, such as GDP per capita and unemployment, which are modelled here in a parsimonious autoregressive fashion. In several of the experiments, these aggregates change only slightly across scenarios, reflecting the fact that the available annual data do not contain a strong and regular link between the binary crisis flags and macroeconomic fluctuations. From a practical point of view, this behaviour is preferable to inventing large swings that are not supported by the historical record, but it also means that the current set-up is not intended to capture the full richness of macroeconomic dynamics.

Several limitations follow directly from these design choices. First, the short sample length restricts the complexity of both the rate models and the validation strategy. Out-of-fold one-year-ahead evaluation is a reasonable way to assess performance in this setting, but it does not fully reflect uncertainty over medium and long horizons. Second, the ensemble tends to attenuate shocks that do not exceed past peaks, which is a desirable property for robustness but may understate the consequences of unprecedented events. Third, age grouping and measurement issues in official statistics, including census revisions and changes in migration definitions, can introduce discontinuities that are difficult to separate from genuine demographic signals. The framework mitigates this through robust estimation and smoothing, but it does not attempt to resolve all measurement artefacts. Finally, the models are not intended to be causal: coefficients are used for prediction and scenario conditioning, not for identifying structural relationships.

Within these limits, the framework offers a practical tool for stress-testing demographic projections in environments with limited data. By combining a familiar cohort-component backbone with a transparent regression ensemble and explicit event flags, the approach allows analysts to explore how different crisis narratives would play out in terms of age-structured populations, while keeping the modelling assumptions clear. The results for Germany, Norway, and Portugal show that this can be done in a way that is sensitive to the strength of historical shocks and yet conservative where the empirical record is quieter.

Future work can proceed in several focused directions. A first step is to move from binary event indicators to a simple description of shock intensity and duration, for example, by allowing graded values between zero and one and short persistence terms. This would permit richer scenarios, such as long, low-level crises or repeated waves of disruption. A second priority is to add a basic probabilistic layer, for instance, through bootstrap ensembles or regularised Bayesian regressions, so that forecasts can be accompanied by uncertainty bands at both cohort and aggregate levels. A third direction is empirical rather

than methodological and concerns the extension of the framework to additional countries or to subnational regions where coherent annual data exist. Such applications would help to assess how far the present findings generalise and would make it possible to adapt the scenario design to different institutional and demographic contexts.

In summary, the paper argues that useful scenario-based population projections can be constructed from relatively short official series by combining simple, robust rate models with an explicit representation of crises. The emphasis is on clarity, internal demographic consistency and the ability to link qualitative narratives about shocks to quantitative trajectories. While the framework does not remove the fundamental uncertainty surrounding long-term population futures, it provides a structured way to organise that uncertainty and to examine how alternative assumptions about crises and stability might affect demographic outcomes.

Author Contributions: Conceptualization, N.C., Z.D.P. and D.D.; methodology, M.P., N.C., Z.D.P. and D.D.; formal analysis, M.P., N.C., Z.D.P. and D.D.; investigation, M.P. and N.C.; software, M.P. and N.C.; writing, M.P., N.C., Z.D.P. and D.D.; contribution to the discussion M.P., N.C., Z.D.P. and D.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used for this study are available through the public repository at https://github.com/nchrkis/demographic-forecast-data accessed on 10 October 2025.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. EMA-Slope Stationarity and Error Propagation

Appendix A.1. Stationarity of the EMA and Its Slope

Let $\{y_t\}$ be covariance-stationary. Define $\mathrm{EMA}_t = \alpha y_t + (1-\alpha)\mathrm{EMA}_{t-1}$ for $\alpha \in (0,1)$, with EMA_0 finite. This is a linear time-invariant filter with absolutely summable impulse response $\{\alpha(1-\alpha)^j\}_{j\geq 0}$, so $\{\mathrm{EMA}_t\}$ is stationary. The slope $\Delta_t = \mathrm{EMA}_t - \mathrm{EMA}_{t-1}$ is a stationary linear filter of $\{y_t\}$ as well. If $\{y_t\}$ is ARIMA(p,d,q) with $d \geq 1$, then applying $abla^d$ yields a stationary input and the same conclusion applies.

Appendix A.2. Variance Under AR(1)

If $y_t = \phi y_{t-1} + \varepsilon_t$ with $|\phi| < 1$ and $\varepsilon_t \sim (0, \sigma_\varepsilon^2)$, then

$$\mathrm{Var}(\Delta_t) = \frac{\alpha^2}{1-(1-\alpha)^2\phi^2}\,\sigma_\varepsilon^2,$$

which illustrates the trade-off between responsiveness (large α) and variance.

Appendix A.3. First-Order Error Propagation in CCM

Let $\widehat{r}_{a,s,t}$ be a forecast rate and $r_{a,s,t}$ the truth, with error $\varepsilon_{a,s,t} = \widehat{r}_{a,s,t} - r_{a,s,t}$. Under the cohort update, a first-order expansion yields

$$\frac{\widehat{P}_{a,s,t+h} - P_{a,s,t+h}}{P_{a,s,t+h}} \approx -\sum_{\tau=1}^{h} \frac{\varepsilon_{a+\tau,s,t+\tau}}{1-r_{a+\tau,s,t+\tau}},$$

assuming bounded rates and independent errors. This implies approximate linear growth of variance with horizon h, and larger sensitivity when survival probabilities are high.

References

1. Bureau, U.C. What Is the Cohort-Component Method? Available online: https://www.census.gov/programs-surveys/popproj/about.html (accessed on 20 April 2025).
2. United Nations, Department of Economic and Social Affairs, Population Division. *World Population Prospects 2022*; United Nations: New York, NY, USA, 2022.
3. United Nations, Department of Economic and Social Affairs; *World Population Prospects 2019*; United Nations: New York, NY, USA, 2019.
4. European Union. *Population on 1 January by Age, Sex and Type of Projection (demo_pjan)*; European Commission: Brussels, Belgium, 2023.
5. Statistisches Bundesamt (Destatis) Federal Statistical Office of Germany. Population by Territory and Average Age. 2024. Available online: https://www.destatis.de/EN/Themes/Society-Environment/Population/Current-Population/Tables/population-by-territory-and-average-age.html (accessed on 25 April 2025).
6. Statistisches Bundesamt (Destatis) Federal Statistical Office of Germany. 2025. Available online: https://www.destatis.de/DE/Home/_inhalt.html (accessed on 19 June 2025).
7. Denmark, S. Population in Denmark (April 2025). 2025. Available online: https://www.statbank.dk (accessed on 1 May 2025).
8. Statistics Norway. Statistics Norway Official Site. 2025. Available online: https://www.ssb.no/en (accessed on 19 June 2025).
9. Instituto Nacional de Estatística (INE). *Population Statistics*; World Bank: Washington, DC, USA, 2023.
10. Instituto Nacional de Estatística. Statistics Portugal (INE). 2025. Available online: https://www.ine.pt/xportal/xmain?xpgid=ine_main&xpid=INE (accessed on 19 June 2025).
11. Bijak, J. *Forecasting International Migration in Europe: A Bayesian View*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2010; Volume 24.
12. Lutz, W.; Butz, W.P.; Samir, K.E. *World Population and Human Capital in the Twenty-First Century*; OUP Oxford: Oxford, UK, 2014.
13. Vanella, P.; Deschermeier, P.; Wilke, C.B. An Overview of Population Projections—Methodological Concepts, International Data Availability, and Use Cases. *Forecasting* **2020**, *2*, 346–363. [CrossRef]
14. Alexander, M.; Alkema, L. A Bayesian cohort component projection model to estimate adult populations at the subnational level in data-sparse settings. *arXiv* **2021**. [CrossRef]
15. Caswell, H.; Sánchez Gassen, N. The sensitivity analysis of population projections. *Demogr. Res.* **2015**, *33*, 801–840. [CrossRef]
16. Madubueze, C.E.; Onwubuya, I.O.; Nkem, G.N.; Chazuka, Z. The transmission dynamics of the monkeypox virus in the presence of environmental transmission. *Front. Appl. Math. Stat.* **2022**, *8*, 1061546. [CrossRef]
17. Reher, D. Population and the Economy During the Demographic Transition. *Econ. Aff.* **2012**, *32*, 10–16. [CrossRef]
18. Gleditsch, R.F.; Syse, A.; Thomas, M.J. Fertility Projections in a European Context: A Survey of Current Practices among Statistical Agencies. *J. Off. Stat.* **2021**, *37*, 547–568. [CrossRef]
19. Thomas, M.; Syse, A.; Rogne, A.; Gleditsch, R. Assessing the Accuracy of National Population Projections: A Case Study of Norway. *Finn. Yearb. Popul. Res.* **2022**, *56*, 31–64. [CrossRef]
20. Hoerl, A.E.; Kennard, R.W. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* **1970**, *12*, 55–67. [CrossRef]
21. Box, G.E.P.; Jenkins, G.M. *Time Series Analysis: Forecasting and Control*; Holden-Day: Port Melbourne, Australia, 1970.
22. Pankratz, A. *Forecasting with Univariate Box-Jenkins Models: Concepts and Cases*; Wiley: Hoboken, NJ, USA, 1983.
23. Politis, D.N.; Romano, J.P. The Stationary Bootstrap. *J. Am. Stat. Assoc.* **1994**, *89*, 1303–1313. [CrossRef]
24. Singer, J.D.; Willett, J.B. *Applied Longitudinal Data Analysis: Modeling Change and Event Occurrence*; Oxford University Press: Oxford, UK, 2003.
25. Brown, R.G. *Statistical Forecasting for Inventory Control*; McGraw-Hill: New York, NY, USA, 1959.
26. Lee, R.D.; Carter, L.R. Modeling and Forecasting U.S. Mortality. *J. Am. Stat. Assoc.* **1992**, *87*, 659–671. [CrossRef]
27. Wiśniowski, A.; Smith, P.W.F.; Bijak, J.; Raymer, J.; Forster, J.J. Bayesian Population Forecasting: Extending the Lee-Carter Method. *Demography* **2015**, *52*, 1035–1059. [CrossRef] [PubMed]
28. Raftery, A.E.; Alkema, L.; Gerland, P. Bayesian Population Projections for the United Nations. *Stat. Sci.* **2014**, *29*, 58–68. [CrossRef] [PubMed]
29. Liu, P.; Raftery, A.E. Accounting for uncertainty about past values in probabilistic projections of the total fertility rate for most countries. *Ann. Appl. Stat.* **2020**, *14*, 685–705. [CrossRef] [PubMed]
30. Girosi, F.; King, G. *Demographic Forecasting*; Princeton University Press: Princeton, NJ, USA, 2008.
31. Tebaldi, C.; West, M. Bayesian Inference on Network Traffic Using Link Count Data. *J. Am. Stat. Assoc.* **1998**, *93*, 557–573. [CrossRef]
32. Keilman, N. Uncertainty in Population Forecasts for the Twenty-First Century. *Annu. Rev. Resour. Econ.* **2020**, *12*, 449–470. [CrossRef]

33. Hoeting, J.A.; Madigan, D.; Raftery, A.E.; Volinsky, C.T. Bayesian model averaging: A tutorial (with comments by M. Clyde, David Draper and E. I. George, and a rejoinder by the authors. *Stat. Sci.* **1999**, *14*, 382–417. [CrossRef]
34. Ševčíková, H.; Raftery, A.E. bayesPop: Probabilistic Population Projections. *J. Stat. Softw.* **2016**, *75*, 5. [CrossRef]
35. Ševčíková, H.; Li, N.; Kantorová, V.; Gerland, P.; Raftery, A.E. Age-Specific Mortality and Fertility Rates for Probabilistic Population Projections. *arXiv* **2015**. [CrossRef]
36. Abel, G.; Bijak, J.; Forster, J.J.; Raymer, J.; Smith, P.W.F.; Wong, J.S. Integrating uncertainty in time series population forecasts: An illustration using a simple projection model. *Demogr. Res.* **2013**, *29*, 1187–1226. [CrossRef]
37. Azose, J.J.; Ševčíková, H.; Raftery, A.E. Probabilistic population projections with migration uncertainty. *Proc. Natl. Acad. Sci. USA* **2016**, *113*, 6460–6465. [CrossRef]
38. Welch, N.G.; Raftery, A.E. Probabilistic forecasts of international bilateral migration flows. *Proc. Natl. Acad. Sci. USA* **2022**, *119*, e2203822119. [CrossRef]
39. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; MIT Press: Cambridge, MA, USA, 2012.
40. Friedman, J.H.; Hastie, T.; Tibshirani, R. *The Elements of Statistical Learning*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2009.
41. Opitz, D.W.; Maclin, R. Popular Ensemble Methods: An Empirical Study. *J. Artif. Intell. Res.* **2019**, *11*, 169–198. [CrossRef]
42. Del-Vecchio, C.; Verrilli, F.; Glielmo, L. Modelling and stability analysis in human population genetics with selection and mutation. *Math. Methods Appl. Sci.* **2018**, *41*, 1492–1508. [CrossRef]
43. Januschowski, T.; Georgiadis, C.; Salinas, D.; Flunkert, S.; Gasthaus, J. Criteria for classifying forecasting methods. *Int. J. Forecast.* **2020**, *36*, 167–173. [CrossRef]
44. Stiegelmeier, E.W.; Bressan, G.M. A fuzzy approach in the study of COVID-19 pandemic in Brazil. *Res. Biomed. Eng.* **2021**, *37*, 263–271. [CrossRef]
45. Dai, K.; Shen, S.; Cheng, C. Evaluation and analysis of the projected population of China. *Sci. Rep.* **2022**, *12*, 3644. [CrossRef] [PubMed]
46. Husain, M.J.; Datta, B.K.; Kostova, D. Disease and demography: A systems-dynamic cohort-component population model to assess the implications of disease-specific mortality targets. *BMJ Open* **2021**, *11*, e043313. [CrossRef]
47. Bontempi, G.; Taieb, S.B.; Borgne, Y.L. *Machine Learning Strategies for Time Series Forecasting*; European Business Intelligence Summer School: Wadern, Germany, 2013; pp. 62–77.
48. Abu-Musa, A.A.; Kobeissi, L.; Hannoun, A.B.; Inhorn, M.C. Effect of war on fertility: A review of the literature. *Reprod. Biomed. Online* **2008**, *17*, 43–53. [CrossRef]
49. Zaccone, A.; Trachenko, K. 2064 global population crisis scenario predicted by the most general dynamic model. *arXiv* **2025**. [CrossRef]
50. Hajirahimova, S.; Aliyeva, A.S. Development of a Prediction Model on Demographic Indicators Based on Machine Learning Methods: Azerbaijan Example. *Int. J. Educ. Manag. Eng.* **2023**, *13*, 1–9. . [CrossRef]
51. Teja, M.D.; Rayalu, G.M. Hybrid time series and machine learning models for forecasting cardiovascular mortality in India: An age specific analysis. *BMC Public Health* **2025**, *25*, 2150. [CrossRef] [PubMed]
52. Sravya, S.; Babu, C. World Population Analysis Using Machine Learning. *Int. J. Manag. Res. Rev.* **2025**, *15*, 219–223.
53. Tibshirani, R. Regression Shrinkage and Selection Via the Lasso. *J. R. Stat. Soc. Ser. B (Methodol.)* **2018**, *58*, 267–288. [CrossRef]
54. Zou, H.; Hastie, T. Regularization and Variable Selection Via the Elastic Net. *J. R. Stat. Soc. Ser. Stat. Methodol.* **2005**, *67*, 301–320. [CrossRef]
55. MacKay, D.J. Bayesian Interpolation. *Neural Comput.* **1992**, *4*, 415–447. [CrossRef]
56. Obi, C. Demand Forecasting in Retail Business Using the Ensemble Machine Learning Framework—A Stacking Approach. *Am. Sci. Res. J. Eng. Technol. Sci.* **2024**, *98*, 309–329.
57. Chambers, J.M.; Hastie, T.J. *Statistical Models in S*, 1st ed.; Routledge: London, UK, 2017. [CrossRef]
58. Wang, C.Y.; Lee, S.J. Regional Population Forecast and Analysis Based on Machine Learning Strategy. *Entropy* **2021**, *23*, 656. [CrossRef]
59. Brunton, S.L.; Proctor, J.L.; Kutz, J.N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. USA* **2016**, *113*, 3932–3937. [CrossRef]
60. Horrocks, J.; Bauch, C.T. Algorithmic discovery of dynamic models from infectious disease data. *Sci. Rep.* **2020**, *10*, 7061. [CrossRef]
61. Doak, D.F.; Gross, K.; Morris, W.F. Understanding and Predicting the Effects of Sparse Data on Demographic Analyses. *Ecology* **2005**, *86*, 1154–1163. [CrossRef]
62. Baffour, B.; Raymer, J. Estimating multiregional survivorship probabilities for sparse data: An application to immigrant populations in Australia, 1981–2011. *Demogr. Res.* **2019**, *40*, 463–502. [CrossRef]
63. Goes, J. Bayesian Forecasting of Mortality Rates for Small Areas Using Spatiotemporal Models. *Demography* **2024**, *61*, 439–462. [CrossRef] [PubMed]

64. Bravo, J.; Coelho, E. Forecasting Subnational Demographic Data using Seasonal Time Series Methods. In Proceedings of the CAPSI 2019 Proceedings, Lisbon, Portugal, 11–12 October 2019.
65. Kemei, Z.; Rotich, T.; Bitok, J. Modelling population dynamics using age-structured system of partial differential equations. *Int. J. Appl. Math. Res.* **2024**, *13*, 110–116. [CrossRef]
66. Saldaña-Núñez, V.; Córdova-Lepe, F.; Moreno-Gómez, F.N. epcc: An R package to assess and simulate the vulnerability of ectothermic populations when facing climate change. *Methods Ecol. Evol.* **2022**, *13*, 955–962. [CrossRef]
67. Granberry, P.; Kim, C.; Resseger, M.; Lee, J.; Lima, A.; Kang, K. Who Is at Risk of Migrating? Developing Synthetic Populations to Produce Efficient Domestic Migration Rates Using the American Community Survey. *Urban Sci.* **2018**, *2*, 80. [CrossRef]
68. Dockery, A.M.; Harris, M.N.; Holyoak, N.; Singh, R.B. A methodology for projecting sparse populations and its application to remote Indigenous communities. *J. Geogr. Syst.* **2021**, *23*, 37–61. [CrossRef]
69. Chi, G.; Wang, D. Population projection accuracy: The impacts of sociodemographics, accessibility, land use, and neighbour characteristics. *Popul. Space Place* **2018**, *24*, e2129. [CrossRef]
70. Schaub, M.; Gimenez, O.; Sierro, A.; Arlettaz, R. Use of Integrated Modeling to Enhance Estimates of Population Dynamics Obtained from Limited Data. *Conserv. Biol.* **2007**, *21*, 945–955. [CrossRef]
71. Merow, C.; Latimer, A.M.; Wilson, A.M.; McMahon, S.M.; Rebelo, A.G.; Silander, J.A., Jr. On using integral projection models to generate demographically driven predictions of species' distributions: Development and validation using sparse data. *Ecography* **2014**, *37*, 1167–1183. [CrossRef]
72. Clarke, P.; Wheaton, B. Addressing Data Sparseness in Contextual Population Research: Using Cluster Analysis to Create Synthetic Neighborhoods. *Sociol. Methods Res.* **2007**, *35*, 311–351. [CrossRef]
73. McKelvey, K.S.; Pearson, D.E. Population estimation with sparse data: The role of estimators versus indices revisited. *Can. J. Zool.* **2001**, *79*, 1754–1765. [CrossRef]
74. Maunder, M.N.; Piner, K.R. Dealing with data conflicts in statistical inference of population assessment models that integrate information from multiple diverse data sets. *Fish. Res.* **2017**, *192*, 16–27. . [CrossRef]
75. Pohl, P.; Slepecki, P.; Spielauer, M. STATSIM: Statistics Austria's dynamic microsimulation model for official population projections. *Stat. J. Iaos J. Int. Assoc. Off. Stat.* **2024**, *40*, 969–984. [CrossRef]
76. Dang, H.A.; Jolliffe, D.; Carletto, C. Data Gaps, Data Incomparability, and Data Imputation: A Review of Poverty Measurement Methods for Data-Scarce Environments. *J. Econ. Surv.* **2019**, *33*, 757–797. [CrossRef]
77. Furceri, D.; Pizzuto, P.; Yarveisi, K. The effect of pandemic crises on fertility. *J. Popul. Econ.* **2024**, *37*, 3. [CrossRef]
78. Winkler-Dworak, M.; Zeman, K.; Sobotka, T. Fertility decline in the later phase of the COVID-19 pandemic: The role of policy interventions, vaccination programmes, and economic uncertainty. *medRxiv* **2024**. [CrossRef]
79. Little, J.S.; Triest, R.K. Seismic shifts: The economic impact of demographic change. An overview. In Proceedings of the Federal Reserve Bank of Boston Conference, Boston, MA, USA, 14–16 November 2001.
80. Mourtada, R.; Melnikas, A.J. Crisis upon crisis: A qualitative study exploring the joint effect of the political, economic, and pandemic challenges in Lebanon on Syrian refugee women's fertility. *Confl. Health* **2022**, *16*, 35. [CrossRef]
81. Huber, P.J. Robust estimation of a location parameter. *Ann. Math. Stat.* **1964**, *35*, 73–101. [CrossRef]
82. Rousseeuw, P.J.; Leroy, A.M. *Robust Regression and Outlier Detection*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2005. [CrossRef]
83. Chatfield, C.; Yar, M. The Holt-Winters forecasting procedure. *J. R. Stat. Soc. Ser. C (Appl. Stat.)* **1978**, *27*, 264–279. [CrossRef]
84. World Bank. World Bank Open Data. 2025. Available online: https://data.worldbank.org/ (accessed on 19 June 2025).
85. Preston, S.H.; Heuveline, P.; Guillot, M. *Demography: Measuring and Modeling Population Processes*; Blackwell: Oxford, UK, 2000.
86. Hyndman, R.J.; Athanasopoulos, G. *Forecasting: Principles and Practice*; OTexts: Melbourne, Australia, 2018. Available online: https://otexts.com/fpp3/ (accessed on 20 June 2025).
87. UNHCR. Syria Emergency. 2015. Available online: https://www.unhcr.org/sy/ (accessed on 25 June 2025).
88. UNHCR. Ukraine Refugee Situation. 2022. Available online: https://data.unhcr.org/en/situations/ukraine (accessed on 24 June 2025).
89. World Health Organization. Timeline: WHO's COVID-19 Response. 2020. Available online: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/interactive-timeline (accessed on 5 June 2025).
90. OECD. The Effect of the Global Financial Crisis on OECD Potential Output. 2014. Available online: https://www.oecd.org/en/publications/the-effect-of-the-global-financial-crisis-on-oecd-potential-output_5jxwtl8h75bw-en.html (accessed on 23 June 2025).
91. Fund, I.M. Portugal: Exiting the Adjustment Program. 2014. Available online: https://www.imf.org/en/news/articles/2015/09/28/04/52/mcs110514 (accessed on 23 June 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

mathematics

MDPI

Article

Modeling and Forecasting U.S. Outbound Travel Demand Across Regions Using Time Series Model and Machine Learning: A Comparative Study

Shengkun Xie * and Chinwendu Onungwe

Global Management Studies, Ted Rogers School of Management, Toronto Metropolitan University, Toronto, ON M5B 2K3, Canada; chinwendu.onungwe@torontomu.ca
* Correspondence: shengkun.xie@torontomu.ca; Tel.: +1-416-979-5000 (ext. 543474)

Abstract

Forecasting of travel demand has become increasingly important in the context of evolving mobility patterns and structural disruptions, including economic fluctuations and public health crises. Classical time series models, although well established in travel-demand analysis, are often limited in their ability to capture non-linear dependencies or adapt to abrupt regime shifts. This study develops and evaluates forecasting techniques drawn from both traditional statistical modeling and machine learning approaches. Their predictive performance and adaptability are benchmarked for U.S. outbound air travel demand across eight global destination regions, Europe, the Caribbean, Asia, South America, Central America, Oceania, the Middle East, and Africa, respectively. Using historical outbound passenger data, six forecasting models are constructed and assessed through multiple forecasting accuracy measures, including the Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE). Empirical results demonstrate that machine-learning-based models, particularly those incorporating adaptive learning components, consistently outperform conventional approaches in modeling structural changes in travel demand data. The study further contributes a generalizable methodological framework that enhances robustness under uncertainty and offers broad applicability to forecasting problems in transportation, tourism, and related domains.

Keywords: travel demand forecasting; machine learning; time series models; data science

MSC: 62P25; 62M10

1. Introduction

The volume of U.S. citizens traveling abroad carries profound economic, political, and cultural implications, not only for the United States but also for destination countries worldwide. International travel demand directly influences the tourism and hospitality sector, shaping both inbound and outbound transportation flows [1]. For stakeholders such as government policy-makers, transportation authorities, service providers, and international marketers, the ability to understand volumes, shifts and uncertainties in travel demand is essential for informed strategic decision-making. In an era defined by increasing global connectivity [2] and the rising contribution of international tourism to national and global GDP [3], accurate forecasting of outbound travel has become critical for

maintaining competitiveness in the market, and ensuring its sustainable growth. Capturing the underlying structure of historical travel demand data often requires advanced statistical and computational tools that are capable of identifying latent patterns. These patterns may be deterministic, stochastic, or a combination of both. Effectively modeling the inherent complexity of the time series is therefore essential for producing reliable forecasts of future travel demand. This highlights the need for modern modeling techniques that can be used to uncover complex data patterns and effectively inform future travel demand forecasting.

Despite its importance, forecasting international travel demand remains challenging. Demand patterns are highly sensitive to structural shifts, including economic downturns and geopolitical events. Global disruptions such as the COVID-19 pandemic have also severely altered mobility and reshaped tourism dynamics [4–8]. Traditional time series models, although effective for modeling linear dependencies and seasonal cycles, often fall short in accommodating the non-linear, non-stationary, and multidimensional dynamics that characterize real-world travel behavior [9,10]. Their reliance on stationarity assumptions, limited handling of exogenous shocks, and inability to efficiently scale with high-frequency or high-volume data pose substantial risks of bias, particularly during periods of rapid change and uncertainty. This is why forecasting based on the non-stationary time series data is challenging and requires much of the attention on better understanding the major patterns obtained from the historical data and their impact to the unobserved future pattern. It is fundamentally important to employ forecasting methods that can both capture historical data patterns and adequately reflect future uncertainty. Compared with modern machine learning (ML) approaches, traditional time series forecasting methods exhibit several limitations. Their inherently linear structure restricts their ability to model non-linear and complex temporal dependencies. Moreover, these methods often rely heavily on stationarity assumptions, which typically require substantial data pre-processing. For instance, Auto Regressive Integrated Moving Average (ARIMA) model incorporates external variables only in a limited manner, making it less suitable for handling high-dimensional data.

Recent advances in data science provide a promising pathway to overcome these limitations. ML and deep learning (DL) methods offer the ability to capture non-linear relationships among variables [11,12], uncover latent patterns through automated feature extraction [13,14], and adapt dynamically by modeling long-term temporal dependencies [15]. A growing body of research highlights the strong performance of algorithms such as Random Forest (RF) and Gradient Boosting, as well as recurrent neural networks (NNs) including Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs), in improving forecasting accuracy in tourism domains [16–19]. This suggests the need to further investigate the applicability of these advanced ML methods to forecasting problems in travel research, in particular, systematically comparing the performance of different forecasting models when applied to travel demand data. This study is motivated by the urgent need for resilient and flexible forecasting models capable of addressing the dynamic and interconnected nature of global travel demand data [20,21]. By conducting a systematic comparison of traditional time series methods (e.g., ARIMA), classical ML models (e.g., Random Forest, Gradient Boosting), and advanced DL architectures (e.g., LSTM, GRU), this research contributes to identifying the most effective and adaptive forecasting strategies for travel demand. Unlike much of the existing literature, which tends to focus on inbound flows or domestic mobility, this study emphasizes outbound U.S. travel across global regions, a dimension that remains under-explored yet critical for understanding international tourism flows, making it a significant and original contribution to tourism research.

From a practical perspective, the findings of this study have important implications for transportation planners, tourism authorities, and policy-makers concerned with managing

https://doi.org/10.3390/math14050758

international mobility under uncertainty. Accurate forecasts of outbound travel demand can support infrastructure investment, airline capacity planning, marketing allocation, and policy formulation related to visa processing and international coordination. By providing region-specific forecasting insights, this study offers decision-makers actionable evidence to support data-informed planning in an increasingly volatile global travel environment. From a methodological standpoint, this study contributes a structured comparative framework for evaluating forecasting performance across heterogeneous regions using multiple modeling models. Rather than advocating a single universal approach, the analysis emphasizes how model suitability depends on regional data characteristics, including scale, volatility, and temporal structure. By systematically benchmarking statistical, ML, and DL models within a unified experimental design, this work advances empirical understanding of model selection in outbound travel demand forecasting and establishes a foundation for future extensions by incorporating additional data sources and hybrid modeling strategies.

2. Literature Review

The economic shocks and subsequent structural shifts in global mobility patterns have underscored the need for robust, adaptive forecasting approaches capable of addressing both short-term shocks and long-term transformations. In response, recent studies have increasingly turned to advanced time series models ML methods to capture the complex, non-linear, and non-stationary dynamics inherent in time series data. For example, Bontempi et al. [22] document how tree-based ensembles and neural networks adapt to evolving data-generating processes without requiring strict parametric assumptions. Similarly, Lai et al. [23] demonstrate that DL architectures, including recurrent and convolutional neural networks, substantially outperform traditional models when time series exhibit non-linearity and non-stationarity.

A prominent research direction centres on the integration of DL architectures to improve forecasting accuracy. For example, Chen et al. [24] proposed a spatial–temporal transformer network that simultaneously models temporal evolution and spatial interactions, while Zhang et al. [25] introduced a hybrid BiLSTM transformer framework to account for both short-term volatility and long-term structural patterns. Similarly, ensemble DL strategies, such as the bagging-based multivariate approach by Sun et al. [26], demonstrate how combining multiple learners can significantly enhance robustness and reduce predictive variance. In addition, Lim and Zohren [27] show that attention-based DL models can capture complex temporal dependencies and regime shifts that are difficult to model using classical approaches. These studies underscore a broader shift in the time series literature toward flexible, data-driven modeling frameworks that explicitly accommodate non-linearity, temporal dependence across multiple scales, and evolving data-generating mechanisms. This growing body of work highlights the potential of modern ML approaches not only to improve predictive accuracy, but also to enhance model robustness, thereby motivating their adoption in increasingly complex real-world time series forecasting.

Another stream of research emphasizes the fusion of multi-source data to enrich forecasting models. For instance, Lee [20] incorporated web search trends as exogenous variables into an SARIMAX framework, highlighting the utility of behavioral information as leading indicators of demand shifts. Complementarily, Colladon et al. [28] applied semantic and social network analysis to online travel forums, showing that unstructured digital footprints can provide early-warning signals for changes in travel demand behavior. These studies collectively highlight the inadequacy of traditional forecasting methods, which often fail to capture the richness and heterogeneity of contemporary travel demand

 https://doi.org/10.3390/math14050758

dynamics. Addressing the challenge of structural breaks and crisis-induced volatility has also become a central focus. For example, Liu et al. [29] introduced the BayesBag method, integrating bootstrap aggregation with Bayesian inference to improve model stability under uncertainty. Hybrid CNN–LSTM approaches, such as those used by [30] in post-pandemic recovery forecasting for Vietnam, further demonstrate the potential of adaptive architectures to outperform conventional models in turbulent contexts. These contributions underscore the importance of resilient methods capable of adjusting to abrupt disruptions.

More recently, advances in data augmentation and transformer-based architectures have further expanded methodological horizons. Diao et al. [31] leveraged spatio-temporal GANs to generate virtual samples for transformer models, effectively mitigating data sparsity issues. Similarly, Li et al. [32] proposed a hybrid framework integrating time series decomposition with a temporal fusion transformer optimized via Bayesian search, achieving superior accuracy and interpretability. Expanding on this line, Yi et al. [33] integrated calendar-based encodings into a transformer encoder–decoder architecture, enhancing both interpretability and predictive performance. Together, these innovations reflect a growing trend toward combining generative, predictive, and interpretable frameworks to address both data limitations and practical decision-making needs.

As evidenced by [24–26], recent literature shows rapid application of advanced ML and DL methods to time series forecasting, particularly in the context of tourism demand prediction. Yet, two critical gaps remain: (1) most studies concentrate on inbound tourism or regional-level demand, leaving outbound travel demand comparatively underexplored; and (2) although ensemble and hybrid models have shown strong potential, their application to outbound U.S. travel demand forecasting is still limited. By addressing these gaps, the present study advances the literature through systematically evaluating a range of forecasting approaches for U.S. outbound travel. It provides both methodological contributions and actionable insights for policy-makers, transportation authorities, and the tourism industry.

3. Materials and Methods

The methodological framework is designed to capture the complex and dynamic patterns inherent in U.S. outbound travel demand data, including linear trends, seasonal effects, and non-linear interactions driven by exogenous factors. The study design and its key components are summarized in the flowchart shown in Figure 1. To facilitate a meaningful comparative analysis, the following sections describe the data and model formulations, providing a clear overview of the proposed methodological framework.

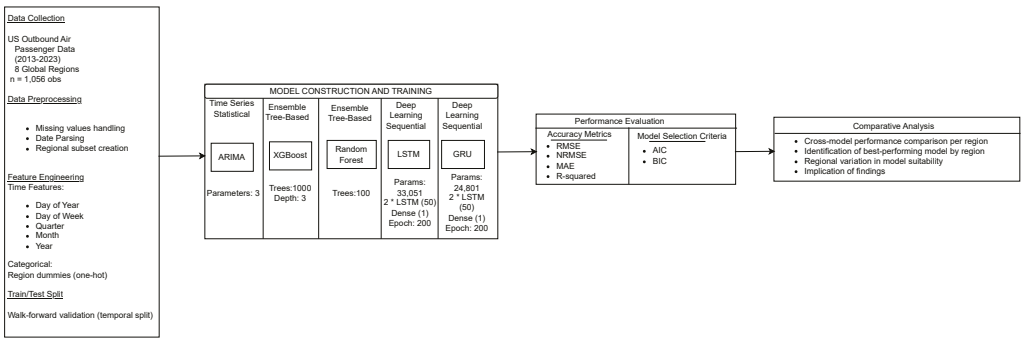


Figure 1. The flowchart of time series processing, modelling, and model evaluation via comparative analysis.

https://doi.org/10.3390/math14050758

3.1. Data

The dataset used in this study was obtained from the U.S. Department of Homeland Security's Customs and Border Protection (CBP) through the Advance Passenger Information System (APIS), an open-source resource that systematically tracks international air passenger flows. A key data summary is given in Table 1. The dataset records monthly counts of U.S. outbound air travelers, disaggregated by geographic destination region, thus offering a granular view of travel flows over time. The key variables include:

- Region—categorical labels representing the major global regions where U.S. travelers are destined.
- Date—temporal identifiers specifying the month and year of each observation.
- Passengers—the total number of U.S. tourists traveling to each destination region in a given month.

These attributes collectively enable the construction of robust time series suitable for forecasting tasks. To ensure analytical rigor, the research framework incorporates several steps: data preprocessing (to handle missing values and ensure consistency), exploratory data analysis (EDA) (to identify underlying patterns), and model development (to evaluate predictive performance across multiple approaches).

Table 1. Summary of dataset characteristics.

Metric	Value
Total observations	1056 (excluding header)
Regions	8
Time points per region	132 months (January 2013–December 2023)
Variables	Date, Region, Passengers
Train/Test split	Varies by region (walk-forward validation)

3.2. Statistical and Machine Learning Methods

Five forecasting models, ARIMA, RF, XGBoost, LSTM, and GRU, are employed in this study. These models were selected to represent three complementary classes of forecasting approaches, providing comprehensive methodological coverage across a broad spectrum of available techniques. The rationale for selecting these models is discussed in the following section.

3.2.1. Model Selection Rationale

ARIMA assumes linear autoregressive structure and is well-suited for stationary or trend-stationary series. Tree-based ensembles (RF, XGBoost) impose no linearity constraints and excel at capturing complex feature interactions without requiring explicit specification. Recurrent architectures (LSTM, GRU) are designed to learn arbitrary temporal dependencies through gating mechanisms. This makes them particularly suitable for time series exhibiting long-range dependences or regime changes. Each model class has demonstrated strong performance in tourism and travel demand forecasting. Foe instance, ARIMA remains a widely used benchmark in tourism forecasting literature [5,6]. Tree-based methods have shown competitive accuracy in recent comparative studies [8,10]. DL approaches, particularly LSTM networks, have emerged as state-of-the-art techniques for capturing complex seasonal patterns in tourism demand data [7,8,12].

On the other hand, ARIMA provides a lightweight, interpretable baseline, requiring little computational effort. Random Forest and XGBoost offer a middle ground with moderate training times and built-in regularization to prevent overfitting. LSTM and GRU, while

https://doi.org/10.3390/math14050758

computationally intensive, provide a capacity to model temporal dynamics of time series. By evaluating models from each class, this study enables a rigorous comparison across different model types rather than within a single model family. This design choice allows us to identify whether the additional complexity of DL architectures yields meaningful improvements over simpler methods for U.S. outbound travel demand forecasting, and whether performance varies systematically across regions with different data characteristics (e.g., high-volume destinations such as Europe versus lower-volume destinations such as Africa).

3.2.2. Auto Regressive Integrated Moving Average (ARIMA)

The ARIMA model [34] is a classical and widely used time series model that combines three components: autoregression (AR), differencing (I), and moving average (MA). The general ARIMA(p, d, q) model can be formally expressed as follows:

$$\Phi_p(B)(1-B)^d D_t = \Theta_q(B)\varepsilon_t, \tag{1}$$

where D_t denotes the observed time series at time t, B is the backshift operator such that $BD_t = D_{t-1}$, ε_t represents a white noise error term with zero mean and constant variance, and the polynomials are defined as:

$$\Phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p, \tag{2}$$

$$\Theta_q(B) = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q, \tag{3}$$

where p is the order of the autoregressive part, d is the degree of differencing used to ensure stationarity, and q is the order of the moving average part. ARIMA explicitly models temporal dependence through a global, linear autoregressive and moving-average structure, relying on assumptions of linearity and stationarity (after differencing) and capturing dynamics with fixed parameters across time. The ARIMA framework served a primary role within this research to provide a robust baseline against which more advanced forecasting approaches could be evaluated.

Note that, the organization of input data differs fundamentally between ARIMA model and ML approaches. The ARIMA model is formulated for a single univariate time series and is therefore estimated separately for each region using only the historical passenger counts of that region. In contrast, ML models require the time series to be transformed into a supervised learning format, where lagged passenger volumes are used as input features, along with additional categorical and calendar-based variables. This allows ML models to be trained globally across regions, capturing both temporal dynamics and cross-sectional heterogeneity within a unified framework.

3.2.3. Input Feature Vector for ML/DL

For each observation at time t, we construct an input feature vector $\mathbf{x} \in \mathbb{R}^{13}$ comprising temporal and categorical features:

$$\mathbf{x} = [d_t^{(y)}, d_t^{(w)}, q_t, m_t, y_t, r_1, r_2, \ldots, r_8]^\top \tag{4}$$

where $d_t^{(y)} \in 1, \ldots, 366$ denotes day of year; $d_t^{(w)} \in 0, \ldots, 6$ denotes day of week; $q_t \in 1, 2, 3, 4$ denotes quarter; $m_t \in 1, \ldots, 12$ denotes month; $y_t \in \mathbb{Z}^+$ denotes year; $r_j \in 0, 1$ for $j = 1, \ldots, 8$ are one-hot encoded region indicators. The target variable is passenger volume at time t, that is $D_t \in \mathbb{R}^+$. Note that, temporal features have been extracted from timestamps and encoded as integer values. While cyclical encoding via sine–cosine

https://doi.org/10.3390/math14050758

transformation (e.g., $m_{\sin} = \sin(2\pi m/12)$, $m_{\cos} = \cos(2\pi m/12)$) can better represent periodic features by preserving adjacency relationships (e.g., December and January), we employ linear integer encoding given the monthly aggregation level of our data. Tree-based models (XGBoost, Random Forest) can implicitly learn seasonal splits through recursive partitioning, and neural networks can capture periodicity through sufficient training examples across multiple seasonal cycles.

3.2.4. Random Forest

RF is an ensemble machine learning algorithm that builds a collection of decision trees and aggregates their outputs to enhance predictive accuracy and stability [35]. For regression tasks, the prediction of the RF model for an input vector $\mathbf{x}$ is given by:

$$\hat{D}(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} h_t(\mathbf{x}), \tag{5}$$

where T is the total number of trees and $h_t(\mathbf{x})$ is the prediction of the t-th tree, which recast time series forecasting as a supervised learning problem by using lagged observations as predictors and learning a nonlinear, data-driven mapping from past values to future outcomes. Each tree is trained on a bootstrap sample of the dataset, and at each split, only a random subset of predictors of size m (with $m < p$, where p is the total number of features) is considered. The best split is chosen by minimizing the mean squared error (MSE):

$$\mathrm{MSE}(S) = \frac{1}{|S|}\sum_{i\in S}(D_i - \bar{D}_S)^2, \tag{6}$$

where S is the set of training observations in a node, $|S|$ is the size of set S and $\bar{y}_S$ is the average of their response values. In this study, RF was applied to effectively model the potentially complex and non-linear relationships between the number of passengers and the time. By aggregating across multiple regression trees, RF reduces variance and produces more stable predictions than a single regression tree. Compared to individual regression trees, RF is less prone to overfitting due to its ensemble structure.

3.2.5. Extreme Gradient Boosting (XGBoost)

XGBoost is an advanced implementation of the Gradient Boosting framework, designed for speed, accuracy, and scalability [36]. It constructs an ensemble of regression trees in a sequential manner, where each new tree is trained to minimize the residual errors of the existing ensemble. Given a training dataset $\{(\mathbf{x}_i, D_i)\}_{i=1}^{n}$ with each $\mathbf{x}_i \in \mathbb{R}^p$ having the same data structure as Equation (4) and response $D_i \in \mathbb{R}$ for each time i, XGBoost models the prediction at iteration t as:

$$\hat{D}_i^{(t)} = \hat{D}_i^{(t-1)} + f_t(\mathbf{x}_i), \tag{7}$$

where $f_t \in \mathcal{F}$ is a regression tree from the function space $\mathcal{F}$ of the trees, and $\hat{y}_i^{(0)}$ is typically initialized as the mean of the response. $\mathbf{x}_i$ is the input vector that consists of the lagged predictors generated from time series D_t. The algorithm minimizes a regularized objective function:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(D_i, \hat{D}_i^{(t-1)} + f_t(\mathbf{x}_i)\right) + \Omega(f_t), \tag{8}$$

where $l(\cdot)$ is a differentiable convex loss function (e.g., squared error loss for regression), and $\Omega(f_t)$ is a regularization term that penalizes model complexity.

https://doi.org/10.3390/math14050758

This iterative refinement enables XGBoost to capture complex non-linear relationships in the time series data while maintaining robustness through regularization. XGBoost is widely recognized for its effectiveness across a variety of machine learning tasks, particularly when dealing with high-dimensional datasets and intricate feature interactions. Its built-in mechanisms for regularization (L_1 and L_2) help prevent overfitting and contribute to predictive stability.

3.2.6. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks are a specialized type of recurrent neural network (RNN) designed to capture sequential dependencies in time series data, including both short- and long-term patterns [37]. LSTMs address the vanishing gradient problem common in traditional RNNs by incorporating a memory cell c_t and hidden state h_t, which allow the network to retain relevant information across long sequences.

In a simplified mathematical view, the LSTM updates its hidden state at each time step t as a function of the current p dimensional input vector $\mathbf{x}_t$ consists of lagged predictors from time series D_t, and the previous hidden state h_{t-1}:

$$h_t = \mathcal{F}(\mathbf{x}_t, h_{t-1}; \Theta), \tag{9}$$

where $\mathcal{F}(\cdot)$ represents the non-linear transformation performed by the LSTM cell and Θ denotes the set of learnable parameters with the network. This abstraction highlights that LSTMs iteratively refine predictions by learning complex, non-linear temporal relationships without specifying detailed gate operations. Its ability to automatically learn temporal structure makes it well-suited for forecasting scenarios where historical context strongly influences future values.

To complement the LSTM, we also implemented the Gated Recurrent Unit (GRU) [38], which can be viewed mathematically in a similar form as LSTM:

$$h_t = \tilde{\mathcal{F}}(\mathbf{x}_t, h_{t-1}; \tilde{\Theta}), \tag{10}$$

where $\tilde{\mathcal{F}}(\cdot)$ is a streamlined recurrent function with fewer parameters than a standard LSTM, which means that $\tilde{\Theta}$ is a subspace of Θ. GRU retains most of the temporal modeling capability while being computationally more efficient, making it suitable for resource-constrained settings. By including both LSTM and GRU, this study evaluates the trade-off between predictive performance and computational cost, providing a comprehensive assessment of deep learning approaches for forecasting U.S. outbound passenger volumes.

The above-mentioned methods are applied to evaluate forecasting performance for US outbound travel demand data, with the objective of identifying a robust and suitable approach for similar prediction tasks in economic and business contexts. To achieve this, a series of experimental studies is conducted to demonstrate how these methods are implemented and to assess their performance under the data characteristics and scenarios considered in this study.

3.2.7. Hyperparameter Tuning for Machine Learning

The RF model was configured with *n_estimators* = 100 trees, using default splitting criteria (max_depth = None and min_samples_split = 2), and a fixed random_state = 42 to ensure reproducibility. The selected hyperparameter values for XGBoost, along with their corresponding justifications, are summarized in Table 2. For the LSTM and GRU models, the hyperparameter values, reported in Table 3, were chosen based on commonly adopted defaults in the machine learning literature and preliminary experimentation, rather than

https://doi.org/10.3390/math14050758

through systematic grid search or Bayesian optimization. This strategy prioritizes reproducibility and computational efficiency; however, we acknowledge that region-specific hyperparameter tuning may further enhance individual model performance and identify this as a direction for future research. The total number of tunable parameters for each selected model is summarized in Table 4.

Table 2. Hyperparameter values configured for XGBoost.

Parameter	Value	Rationale
n_estimators	1000	Sufficient ensemble size with early stopping
max_depth	3	Prevents overfitting; promotes generalization
learning_rate	0.1	Standard value balancing convergence speed and accuracy
early_stopping_rounds	50	Halts training when validation performance plateaus

Table 3. Neural network hyperparameter values for LSTM and GRU.

Parameter	Value
Hidden units per layer	50
Number of layers	2 (stacked)
Optimizer	Adam (learning rate = 0.001)
Loss function	Mean Squared Error
Epochs	200
Batch Size	32
Input normalization	Min-Max scaling [0, 1]
Sequence shuffling	Disabled (shuffle = False)

Table 4. Model Parameters Summary.

Model	Number of Trainable Parameters
ARIMA	3 (1 AR + 1 MA + 1 constant/variance)
XGBoost	$\sim$1000 trees $\times$ splits per tree
Random Forest	100 trees $\times$ features
LSTM	33,051
GRU	24,801

3.2.8. Model Validation Strategy

For the ARIMA model, we implement walk-forward (expanding window) validation: at each time step t in the test period, the model is refit using all available observations up to $t-1$, a one-step-ahead forecast is generated. The actual observation is then appended to the training history before proceeding to $t+1$. This approach mimics realistic forecasting conditions where the model is updated as new data becomes available. For ML and DL models (XGBoost, Random Forest, LSTM, GRU), a single temporal split is employed, where models are trained once on the full training set and evaluated on the held-out test period. While walk-forward validation would provide more robust performance estimates, the computational cost of iteratively retraining neural networks (each requiring 200 epochs) and ensemble models (1000+ trees) for each forecast horizon renders this approach impractical within the scope of the current study. While the ARIMA model employs walk-forward validation with expanding windows, the ML and DL models are evaluated using a single train–test split. This asymmetry arises from computational

https://doi.org/10.3390/math14050758

constraints: iteratively retraining neural networks and large ensemble models for each forecast step would substantially increase computational requirements. Consequently, the reported performance metrics for ML/DL models reflect a single temporal split rather than averaged performance across multiple forecast origins.

3.2.9. Methodological Limitations

Several methodological constraints should be acknowledged. First, the DL models (LSTM, GRU) were trained with fixed hyperparameters across all regions without region-specific tuning. Given the heterogeneity in passenger volumes and seasonal patterns across destination regions, individualized hyperparameter optimization could potentially enhance model performance, particularly for regions with lower sample variability (e.g., Africa, Oceania). Second, no explicit regularization mechanisms (dropout layers, L2 weight decay) were incorporated into the neural network architectures. While the shallow network depth partially mitigates overfitting, future work could investigate the impact of regularization on generalization performance. Third, the current implementation uses a fixed training epoch count (i.e., 200) without early stopping for neural networks. Implementing patience-based early stopping could improve computational efficiency and potentially reduce overfitting, particularly for regions exhibiting faster convergence. These choices reflect a balance between methodological rigor and practical constraints, with the primary objective of establishing baseline comparative performance across model families rather than achieving maximally optimized predictions for each individual model.

4. Experimental Study

4.1. Data Preparation

The raw dataset underwent a systematic preprocessing pipeline to ensure data quality and analytical consistency. The following steps were applied sequentially:

4.1.1. Missing Value Handling

An initial data audit identified missing passenger counts for December 2023 across all eight regions. To preserve temporal continuity while avoiding information leakage from future observations, missing values were imputed using linear interpolation with a time-based method. Specifically, for each region r, missing values were estimated as:

$$\hat{D}_{t,r} = D_{t-1,r} + \frac{D_{t+1,r} - D_{t-1,r}}{2}, \tag{11}$$

where $D_{t,r}$ denotes passenger volume for region r at time t. This approach maintains the temporal structure of the series while providing reasonable estimates based on adjacent observations. The interpolation was applied independently within each regional subset to preserve region-specific trends.

4.1.2. Date Parsing and Temporal Feature Extraction

The original date field, recorded in YYYY-MM format, was parsed into a date time index to enable time series operations. From this temporal index, the following features were extracted: Year, Month, Quarter, Day of Week, and Day of Year. These temporal features were encoded as integer values rather than cyclical (sine–cosine) transformations. This design choice reflects the monthly aggregation level of the data, where tree-based models can implicitly learn seasonal splits through recursive partitioning, and neural networks can capture periodicity through exposure to multiple seasonal cycles during training.

https://doi.org/10.3390/math14050758

4.1.3. Regional Subset Creation

The dataset was stratified by destination region to enable region-specific model training and evaluation. Eight regional subsets were created based on the categorical region labels provided in the source data. Each regional subset contains 132 monthly observations. Region identifiers were subsequently transformed into one-hot encoded dummy variables (8 binary features) to serve as categorical predictors for machine learning models.

4.1.4. Train–Test Split

Data partitioning followed a temporal split strategy to prevent data leakage. For each region, observations from 2013–2019 (approximately 80%) were allocated to the training set, with subsequent observations reserved for testing. This chronological split ensures that model evaluation reflects genuine out-of-sample forecasting performance.

4.2. Computational Environment

All experiments were conducted using Python 3.11.5 on MacOS. The hardware configuration consisted of Apple M2 with 16 GB of RAM. The key software libraries and their versions are summarized in Table 5.

Table 5. Software environment specifications.

Library	Version	Purpose
TensorFlow	2.19.0	Deep learning framework
Keras	3.10.0	Neural network API
XGBoost	3.1.3	Gradient boosting implementation
scikit-learn	1.6.1	Random Forest and preprocessing
statsmodels	0.14.6	ARIMA implementation
pandas	2.2.2	Data manipulation
numpy	2.0.2	Numerical computation

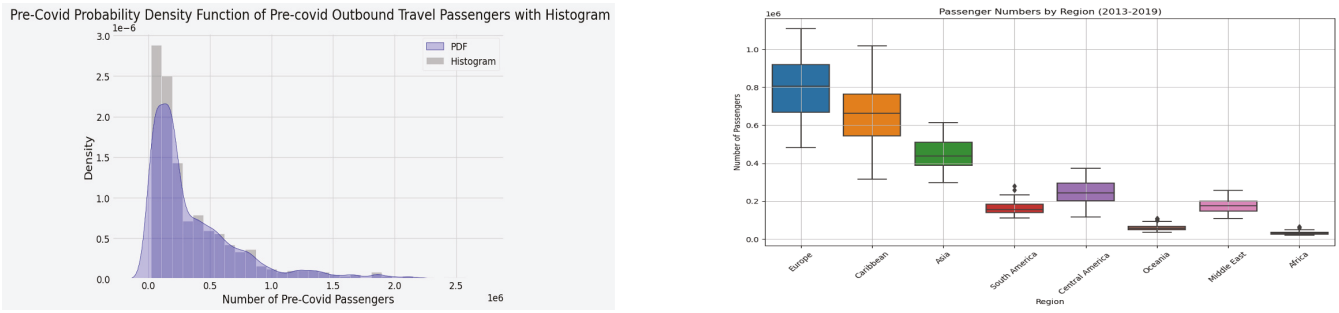
4.3. Exploratory Data Analysis

To explore the distributional characteristics of outbound travel volumes, a distribution was obtained using the full dataset, which is displayed in Figure 2a. The resulting distribution reveals a clear right-skewed pattern, with a high frequency of observations concentrated at lower passenger counts. This suggests that a majority of routes involve relatively modest passenger volumes, potentially reflecting either lower travel demand or less frequent flight availability. Conversely, the long tail of the distribution highlights the presence of extreme values, routes or regions with exceptionally high travel volumes. These outliers stretch the passenger count up to approximately 2.7 million and are likely attributable to major international hubs or popular tourism destinations. The distributional analysis thus underscores the unequal distribution of air traffic among different destinations and the importance of identifying these high-impact routes in travel demand forecasting. This result underscores the importance of conducting region- or destination-specific analyses to effectively account for heterogeneity in the data.

To examine regional disparities, we conducted a comparative analysis of total passenger volumes. The results, presented in Figure 2b, reveal distinct differences in central tendency across regions. The results indicate that Europe and the Caribbean consistently account for the highest outbound passenger traffic, both before and after the pandemic. This suggests sustained demand for these regions, driven by a combination of tourism, historical ties, and well-developed airline connectivity. Regions such as Asia and North America also record substantial passenger volumes, though they rank slightly below the

 https://doi.org/10.3390/math14050758

leading regions. These findings point to the important but secondary role these areas play in U.S. outbound travel networks.



(**a**) (**b**)

Figure 2. Data distribution function of outbound travel passengers and passenger numbers by regions (based on 2013–2018 data).

4.4. Designed Experiments

A series of experiments were conducted to assess the performance of various modeling approaches. The objective was to determine the most accurate and computationally efficient models discussed above for forecasting outbound passenger numbers, thereby supporting informed decision-making in sectors such as tourism, transportation management, and logistics.

- Experiment 1: ARIMA Modeling. The first experiment implemented the ARIMA model as a benchmark. The model was trained on the training data and evaluated using Root Mean Squared Error (RMSE), to assess goodness-of-fit and forecast accuracy.
- Experiment 2: Machine Learning Models (RF, Gradient Boosting,LSTM and GRU). In the second experiment, RF and XGBoost (Gradient Boosting) models were trained using the Year and Month features as inputs. These models were tuned via hyperparameter optimization to enhance performance.
- Experiment 3: Comparative Analysis. The third experiment focused on a comparative evaluation of the ARIMA model and the machine learning methods. Performance was assessed on the same test dataset using the same performance metrics to determine which approach provided superior predictive accuracy and reliability for passenger number forecasting.

4.5. Performance Evaluation

Model performance was evaluated using out-of-sample predictive accuracy metrics. Three complementary measures were employed:

- Root Mean Squared Error (RMSE): Measures the average magnitude of prediction errors, with lower values indicating greater accuracy. RMSE is particularly sensitive to large errors due to the squaring operation.
- Normalized RMSE (NRMSE): Computed as RMSE divided by the mean of observed values, enabling cross-regional comparison by accounting for differences in passenger volume scales.
- Mean Absolute Error (MAE): Measures the average absolute deviation between predicted and observed values, providing a more robust measure less sensitive to outliers than RMSE.

In addition, Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) are used exclusively to evaluate the ARIMA model. These likelihood-based criteria

https://doi.org/10.3390/math14050758

are theoretically grounded in maximum likelihood estimation with well-defined parameter dimensionality, which does not naturally extend to ML ensemble methods or neural network architectures where the concept of "effective parameters" differs fundamentally. Consequently, model comparison for ML/DL approaches relies exclusively on out-of-sample predictive accuracy metrics (i.e., RMSE, NRMSE, MAE).

5. Results

In this section, we analyze the obtained results. Table 6 summarizes the model performance across all evaluation metrics. Overall, the LSTM model achieves the highest predictive accuracy, as evidenced by its lowest RMSE, MAE, and normalized RMSE values. This suggests a strong capability in capturing complex, non-linear temporal dependencies. However, these gains in accuracy come with increased model complexity (excessive model parameter to be trained), highlighting a trade-off between fit and parsimony, particularly relevant in contexts where interpretability and computational efficiency are critical.

Table 6. Comparison of performance metrics for the models we consider across all regions.

Model	Mean	RMSE	NRMSE	MAE
ARIMA(1, 1, 1)	375,939	509,181	1.454	328,948
XGBoost	375,939	117,384	0.312	63,896
Random Forest	375,939	117,762	0.313	63,932
LSTM	375,939	101,999	0.271	61,392
GRU	375,939	107,763	0.286	68,763

Notably, simpler models such as ARIMA and Random Forest demonstrate competitive accuracy while maintaining lower complexity. We also note that modeling the aggregated data without separating regions would introduce substantial heterogeneity, as evidenced by the markedly different mean levels of outbound passenger volumes across regions. Because of this reason, we further analyze model performance across eight regions: Europe, the Caribbean, Asia, South America, Central America, Oceania, the Middle East, and Africa. Given the heterogeneity in travel demand patterns driven by seasonal, economic, cultural, and geopolitical factors, this regional breakdown offers insight into the adaptability of each model.

Model performance varied considerably across regions, reflecting differences in data scale, variability, and underlying travel dynamics. Overall, the comparative analysis highlights the importance of aligning model complexity with data characteristics and operational constraints.

For Africa, where the results are reported in Table 7, the ARIMA(1, 1, 1) model achieved the best performance across all key error metrics, demonstrating a strong capacity to capture the underlying linear structure of passenger flows. Both XGBoost and Random Forest showed moderate results, with the latter slightly outperforming the former. Deep learning models, particularly LSTM, performed poorly, as evidenced by high RMSE, and MAE. This suggests overfitting due to limited data volume and variability, emphasizing the limitations of data-intensive models in low-sample contexts. While ARIMA offered simplicity and accuracy, it lacked flexibility to account for non-linear effects, whereas tree-based models provided adaptability at the cost of computational efficiency.

In contrast, the results for Asia, shown in Table 8, exhibited distinct patterns favoring deep learning approaches. Both LSTM and GRU achieved the best predictive accuracy, benefiting from their ability to capture non-linear temporal dependencies and long-term relationships inherent in the data. ARIMA remained a robust yet simpler alternative,

 https://doi.org/10.3390/math14050758

performing adequately for relatively stable time series but failing to respond effectively to non-linear shifts. XGBoost and Random Forest produced intermediate results, suggesting that their performance could improve with feature augmentation or hyperparameter tuning. These results illustrate how model suitability depends on the complexity and temporal structure of regional data.

For the Caribbean, shown in Table 9, Random Forest delivered the lowest RMSE and normalized RMSE, confirming its superior ability to model non-linear interactions without substantial overfitting. Deep learning models such as LSTM and GRU provided moderate performance. ARIMA underperformed, reflecting its inability to represent pronounced seasonality and volatility in regional travel patterns. These findings highlight the efficacy of ensemble tree-based methods in small- to medium-scale datasets where non-linearity is present but data are insufficient for deep architectures.

In Europe, where the obtained results are given in Table 10, LSTM achieved the lowest RMSE (219,030), while GRU obtained the lowest normalized RMSE (0.13), indicating that both models effectively captured complex seasonal and trend components in the data. However, their computational demands and tuning requirements limit their scalability for operational use. ARIMA, by comparison, underperformed due to its rigid linear assumptions, underscoring its limited adaptability to intricate time series dynamics. The results suggest that while deep learning models excel in high-variability environments, practical applications must balance predictive power against resource and interpretability considerations.

For the Middle East, XGBoost emerged as the top-performing model, yielding the lowest RMSE, normalized RMSE, and MAE values, which are shown in Table 11. Its gradient boosting framework effectively modeled non-linear dependencies and complex regional variations. Random Forest performed moderately well, outperforming ARIMA but trailing XGBoost. LSTM, on the other hand, recorded the weakest performance, likely due to overfitting in a relatively small dataset. While XGBoost achieved the highest accuracy, its computational intensity and limited interpretability suggest that marginal gains over simpler alternatives should be carefully evaluated in applied settings.

In Oceania, from the results shown in Table 12, we can see that ARIMA achieved the lowest RMSE and MAE, demonstrating strong predictive accuracy and reliable tracking of observed data trends. Random Forest also performed competitively, while GRU and XGBoost showed higher errors and lower stability. Although ARIMA's linear structure effectively captured the dominant temporal trends, its inability to model complex dependencies may restrict its usefulness in more data-rich or volatile contexts. Nonetheless, for small or stable datasets, ARIMA remains a parsimonious and effective forecasting tool.

Finally, the comparative results corresponding to South America are displayed in Table 13. In this table, we see that XGBoost outperformed all other models, achieving the lowest RMSE (23,702.39), normalized RMSE (0.17), and MAE (18,465.63). ARIMA provided a consistent baseline with clear interpretability but lagged in predictive precision. Both LSTM and GRU underperformed, likely due to limited temporal depth and data complexity. These outcomes reaffirm the balance required between accuracy, model complexity, and interpretability when selecting predictive approaches for regional forecasting. Taken together, the results reveal that model performance is context-dependent. ARIMA remains effective for stable or low-variability series, tree-based ensemble methods (XGBoost, Random Forest) excel in moderately complex and nonlinear contexts, and deep learning models (LSTM, GRU) are most suitable when sufficient data are available to support their representational capacity. The comparative findings underscore the necessity of tailoring model selection to regional data characteristics, ensuring that predictive accuracy is achieved without sacrificing scalability or interpretability.

https://doi.org/10.3390/math14050758

Table 7. Comparison of performance metrics for the models we consider for the Africa region.

Model	RMSE	NRMSE	MAE	Mean
ARIMA(1, 1, 1)	9126.21	0.23	7322.24	40,999.96
XGBoost	13,473.13	0.34	11,302.26	40,999.96
RandomForest	10,607.34	0.27	8440.06	40,999.96
LSTM	23,511.63	0.59	19,676.35	40,999.96

Table 8. Comparison of performance metrics for the models we consider for Asia.

Model	RMSE	NRMSE	MAE	Mean
ARIMA(1, 1, 1)	65,061.99	0.34	51,120.12	533,275.12
XGBoost	81,353.35	0.42	71,266.72	533,275.12
RandomForest	74,971.84	0.39	63,719.83	533,275.12
LSTM	56,747.39	0.29	48,897.79	533,275.12
GRU	64,776.38	0.34	48,019.96	533,275.12

Table 9. Comparison of performance metrics for the models we consider for the Caribbean region.

Model	RMSE	NRMSE	MAE	Mean
ARIMA(1, 1, 1)	155,508.54	0.26	131,744.02	752,918.58
XGBoost	122,773.46	0.21	101,616.95	752,918.58
RandomForest	95,221.44	0.16	80,707.53	752,918.58
LSTM	139,538.48	0.23	106,493.79	752,918.58
GRU	136,793.17	0.23	107,670.03	752,918.58

Table 10. Comparison of performance metrics for the models we consider for Europe.

Model	RMSE	NRMSE	MAE	Mean
ARIMA(1, 1, 1)	343,121.6	0.18	284,167.11	1,532,984.42
XGBoost	290,914.61	0.15	238,561.1	1,532,984.42
RandomForest	304,987.31	0.16	268,880.74	1,532,984.42
LSTM	227,320.09	0.12	186,343.97	1,532,984.42
GRU	238,337.36	0.13	200,376.47	1,532,984.42

Table 11. Comparison of performance metrics for the models we consider for Middle East.

Model	RMSE	NRMSE	MAE	Mean
ARIMA(1, 1, 1)	27,616.97	0.3	23,450.8	207,847.71
XGBoost	16,556.52	0.18	14,404.71	207,847.71
RandomForest	25,293.01	0.27	21,994.26	207,847.71
LSTM	43,638.85	0.47	36,282.11	207,847.71
GRU	41,797.86	0.45	35,623.91	207,847.71

Table 12. Comparison of performance metrics for the models we consider for Oceania.

Model	RMSE	NRMSE	MAE	Mean
ARIMA(1, 1, 1)	15,775.95	0.28	13,029.46	75,791
XGBoost	19,673.66	0.35	15,029.42	75,791
RandomForest	16,308.87	0.29	13,535.49	75,791
LSTM	21,295.64	0.38	17,198.77	75,791
GRU	28,450.97	0.51	23,537.89	75,791

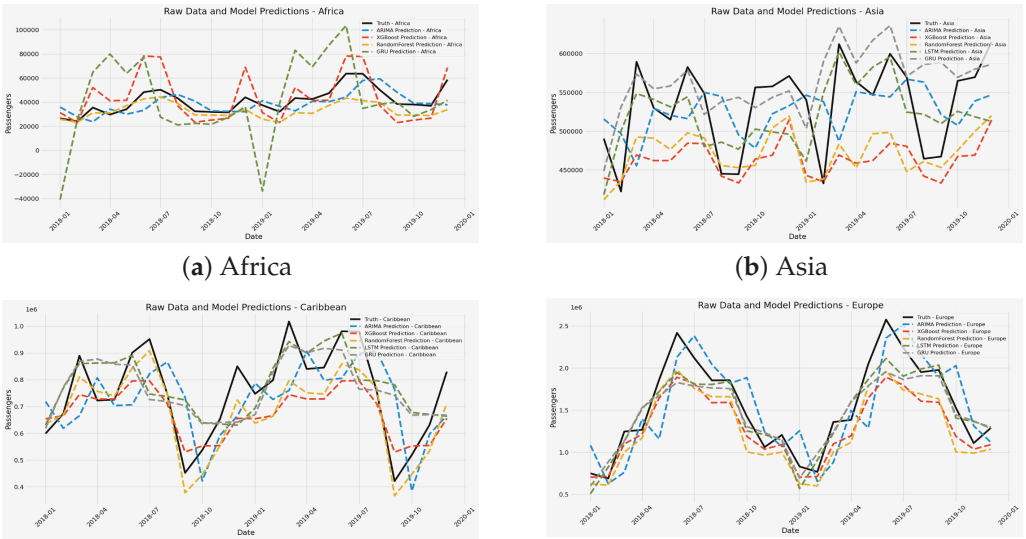
https://doi.org/10.3390/math14050758

Table 13. Comparison of performance metrics for the models we consider for South America.

Model	RMSE	NRMSE	MAE	Mean
ARIMA(1, 1, 1)	37,131.99	0.26	27,842.25	185,103.38
XGBoost	23,702.39	0.17	18,465.63	185,103.38
RandomForest	29,000.86	0.21	23,552.66	185,103.38
LSTM	36,307.37	0.26	30,483.38	185,103.38
GRU	36,881.12	0.26	32,838.02	185,103.38

Figures 3 and 4 illustrate substantial regional heterogeneity in U.S. outbound travel demand and corresponding differences in model performance. Across all regions, outbound passenger volumes exhibit distinct magnitudes, volatility levels, and temporal patterns, underscoring the importance of region-specific analysis. No single forecasting model consistently outperforms others across all regions, highlighting the limitations of a uniform modeling strategy. In regions characterized by higher variability and more complex demand dynamics, such as Europe and Asia, machine learning and deep learning models more effectively track observed fluctuations, while in regions with lower passenger volumes or more stable patterns, simpler models such as ARIMA remain competitive and, in some cases, superior.

The comparative results further demonstrate that model effectiveness depends critically on data richness and structural complexity. Deep learning architectures, particularly LSTM and GRU, tend to provide improved predictive accuracy in regions with sufficient data and pronounced non-linear temporal dependencies, whereas ensemble tree-based methods offer strong performance in moderately complex settings. Conversely, in data-sparse or relatively stable regions, traditional time series models yield robust and interpretable forecasts with lower computational cost. Together, these findings emphasize that outbound travel demand forecasting benefits from a flexible, region-adaptive modeling framework that balances predictive accuracy, computational efficiency, and interpretability rather than relying on a single universal approach.

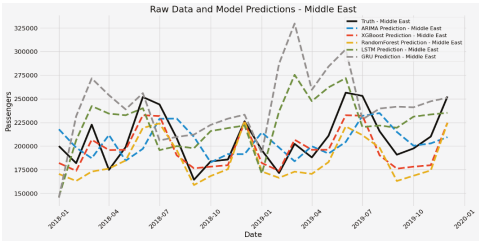


(**a**) Africa (**b**) Asia

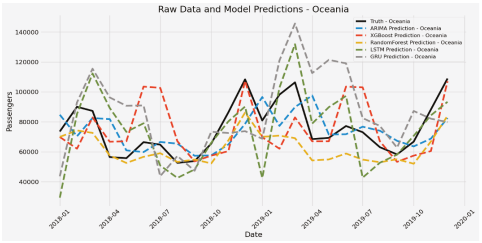
(**c**) Caribbean (**d**) Europe

Figure 3. U.S. outbound passengers vs. model predictions for the regions of Africa, Asia, Caribbean and Europe.

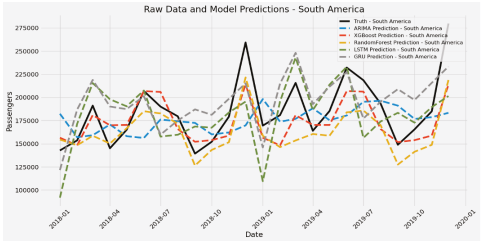
https://doi.org/10.3390/math14050758



(**a**) Middle East



(**b**) Oceania



(**c**) South America

Figure 4. U.S. outbound passengers vs. model predictions for the regions of Middle East, Oceania and South America.

6. Discussions

The empirical results reveal substantial heterogeneity in model performance across regions, underscoring that forecasting accuracy is strongly conditioned on data scale, variability, and temporal structure. Ensemble tree-based models, such as XGBoost and Random Forest, consistently performed well in regions characterized by moderate data volume and nonlinear seasonal patterns, while deep learning models exhibited advantages primarily in high-traffic regions with richer temporal dynamics. These findings indicate that no single modeling approach dominates across all contexts, highlighting the importance of aligning model complexity with regional data characteristics.

Deep learning architectures, including LSTM and GRU, demonstrated improved predictive accuracy in regions with sufficient historical depth, where long-term temporal dependencies and evolving demand patterns are more pronounced. Conversely, their weaker performance in data-sparse regions suggests sensitivity to sample size and increased risk of overfitting. This shows the practical limits of highly parameterized models in constrained settings. Tree-based ensemble methods offered a more stable balance between flexibility and robustness in several regions, making them good alternatives when deep architectures are not ideal for the data.

Feature design also played a critical role in shaping model performance. The inclusion of temporal indicators and lagged passenger volumes enhanced forecast accuracy by enabling models to better capture seasonal persistence and trend evolution. These results highlight that careful data organization and feature engineering remain essential, even when advanced learning algorithms are employed. From an applied perspective, the region-specific insights derived from this comparative analysis can support more targeted decision-making in capacity planning, marketing allocation, and policy design. By demonstrating how forecasting performance varies across regional contexts, this study provides practical guidance for selecting appropriate modeling strategies in real-world travel demand applications.

https://doi.org/10.3390/math14050758

7. Conclusions and Future Work

This study demonstrates that the suitability of forecasting models for outbound travel varies significantly across regions, depending largely on the volume of data and the complexity of underlying travel patterns. For high-traffic regions such as Europe and the Caribbean, advanced machine learning models, particularly XGBoost and Random Forest, consistently outperformed traditional approaches. These models effectively captured seasonal variations and abrupt changes, delivering high predictive accuracy. In contrast, for regions with lower passenger volumes, simpler models like ARIMA proved more reliable, offering stable forecasts while mitigating the risk of overfitting in data-sparse environments.

The results emphasize the importance of region-specific model selection, where the balance between model complexity and data availability must be carefully managed. While sophisticated models are advantageous in data-rich contexts, simpler, more interpretable models offer better generalizability in regions with limited data. This trade-off between bias and variance is critical for ensuring robust and practical forecasting outcomes.

The findings of this study show direct implications for industry practitioners and policymakers. For airline revenue management, the demonstrated superiority of adaptive machine learning models enables more responsive capacity planning, allowing carriers to optimize seat inventory and route frequency in anticipation of demand fluctuations across destination regions. Airport authorities and ground handlers can leverage regional forecasts to allocate staffing and terminal resources more efficiently, particularly during seasonal peaks.

From a policy perspective, destination countries can utilize these forecasting frameworks to anticipate U.S. tourist inflows, informing visa processing capacity, border control staffing, and tourism infrastructure investments. Also, national tourism organizations can apply region-specific forecasts to allocate marketing budgets strategically, targeting promotional efforts toward periods and regions where demand elasticity is highest. At the macroeconomic level, improved forecasting accuracy supports more reliable projections of tourism-related foreign exchange earnings, informing fiscal planning in tourism-dependent economies.

Looking ahead, future research can build on these findings in several ways. First, incorporating additional features, such as economic indicators, political events, and environmental factors like weather patterns, could significantly enhance predictive performance, especially in regions characterized by high variability. Second, exploring ensemble learning techniques, such as stacking or blending, offers the potential to combine the strengths of various models. Such hybrid approaches could adaptively apply simpler models in data-scarce settings and more complex algorithms in data-rich contexts to yield consistently strong results.

Author Contributions: Conceptualization, S.X. and C.O.; methodology, S.X. and C.O.; software, C.O.; validation, S.X. and C.O.; formal analysis, S.X. and C.O.; investigation, S.X. and C.O.; writing—original draft preparation, S.X. and C.O.; writing—review and editing, S.X.; visualization, C.O.; supervision, S.X.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article material. For further inquiries, please contact the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

https://doi.org/10.3390/math14050758

References

1. Agbola, F.W.; Dogru, T.; Gunter, U. Tourism demand: Emerging theoretical and empirical issues. *Tour. Econ.* **2020**, *26*, 1307–1310. [CrossRef]
2. Tuite, A.R.; Bhatia, D.; Moineddin, R.; Bogoch, I.I.; Watts, A.G.; Khan, K. Global trends in air travel: Implications for connectivity and resilience to infectious disease threats. *J. Travel Med.* **2020**, *27*, taaa070. [CrossRef]
3. Richter, L.K. International tourism and its global public health consequences. *J. Travel Res.* **2003**, *41*, 340–347. [CrossRef]
4. Ceccato, R.; Rossi, R.; Gastaldi, M. Travel demand prediction during COVID-19 pandemic: Educational and working trips at the University of Padova. *Sustainability* **2021**, *13*, 6596. [CrossRef]
5. Baghestani, A.; Tayarani, M.; Mamdoohi, A.R.; Habibian, M.; Gao, O. Travel demand management implications during the COVID-19 pandemic: The case study of Tehran. *Sustainability* **2023**, *15*, 1209. [CrossRef]
6. Shemer, L.; Shayanfar, E.; Avner, J.; Miquel, R.; Mishra, S.; Radovic, M. COVID-19 impacts on mobility and travel demand. *Case Stud. Transp. Policy* **2022**, *10*, 2519–2529. [CrossRef]
7. Miller, E.J. Travel demand models, the next generation: Boldly going where no-one has gone before. In *Mapping the Travel Behavior Genome*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 29–46.
8. Zhang, T.; Zhang, Z.; Xue, G. Mitigating the disturbances of events on tourism demand forecasting. *Ann. Oper. Res.* **2024**, *342*, 1019–1040. [CrossRef]
9. Lim, C.; McAleer, M. Time series forecasts of international travel demand for Australia. *Tour. Manag.* **2002**, *23*, 389–396. [CrossRef]
10. Jha, K.; Sinha, N.; Arkatkar, S.S.; Sarkar, A.K. A comparative study on application of time series analysis for traffic forecasting in India: Prospects and limitations. *Curr. Sci.* **2016**, *110*, 373–385. [CrossRef]
11. Peng, H.; Yan, J.; Yu, Y.; Luo, Y. Time series estimation based on deep learning for structural dynamic nonlinear prediction. *Structures* **2021**, *29*, 1016–1031. [CrossRef]
12. Chen, R.; Jin, X.; Laima, S.; Huang, Y.; Li, H. Intelligent modeling of nonlinear dynamical systems by machine learning. *Int. J. -Non-Linear Mech.* **2022**, *142*, 103984. [CrossRef]
13. Li, L.; Wu, Y.; Ou, Y.; Li, Q.; Zhou, Y.; Chen, D. Research on machine learning algorithms and feature extraction for time series. In *Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*; IEEE: Piscataway, NJ, USA, 2017; pp. 1–5.
14. Herff, C.; Krusienski, D.J. Extracting features from time series. In *Fundamentals of Clinical Data Science*; Springer: Cham, Switzerland, 2019; pp. 85–100.
15. Ubal, C.; Di-Giorgi, G.; Contreras-Reyes, J.E.; Salas, R. Predicting the long-term dependencies in time series using recurrent artificial neural networks. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 1340–1358. [CrossRef]
16. Law, R.; Li, G.; Fong, D.K.C.; Han, X. Tourism demand forecasting: A deep learning approach. *Ann. Tour. Res.* **2019**, *75*, 410–423. [CrossRef]
17. Yu, N.; Chen, J. Design of machine learning algorithm for tourism demand prediction. *Comput. Math. Methods Med.* **2022**, *2022*, 6352381. [CrossRef]
18. Li, Y.; Lin, Z.; Xiao, S. Using social media big data for tourist demand forecasting: A new machine learning analytical approach. *J. Digit. Econ.* **2022**, *1*, 32–43. [CrossRef]
19. Zhang, Y.; Li, G.; Muskat, B.; Law, R. Tourism demand forecasting: A decomposed deep learning approach. *J. Travel Res.* **2021**, *60*, 981–997. [CrossRef]
20. Lee, G.C. A Data-Driven Approach to Tourism Demand Forecasting: Integrating Web Search Data into a SARIMAX Model. *Data* **2025**, *10*, 73. [CrossRef]
21. Giannoukou, I.; Kougia, A. Innovative Digital Strategies in Crisis Management for Tourism Enterprises: Enhancing Resilience and Sustainable Growth in the Digital Era. In *International Conference of the International Association of Cultural and Digital Tourism*; Springer: Berlin/Heidelberg, Germany, 2025; pp. 47–87.
22. Bontempi, G.; Ben Taieb, S.; Le Borgne, Y.A. Machine learning strategies for time series forecasting. In *European Big Data Management and Analytics Summer School*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 62–77.
23. Lai, G.; Chang, W.C.; Yang, Y.; Liu, H. Modeling long-and short-term temporal patterns with deep neural networks. In *Proceedings of the 41st international ACM SIGIR Conference on Research & Development in Information Retrieval*; Association for Computing Machinery: New York, NY, USA, 2018; pp. 95–104.
24. Chen, J.; Li, C.; Huang, L.; Zheng, W. Tourism demand forecasting: A deep learning model based on spatial-temporal transformer. *Tour. Rev.* **2025**, *80*, 648–663. [CrossRef]
25. Zhang, Y.; Tan, W.H.; Zeng, Z. Tourism Demand Forecasting Based on a Hybrid Temporal Neural Network Model for Sustainable Tourism. *Sustainability* **2025**, *17*, 2210. [CrossRef]

 https://doi.org/10.3390/math14050758

26. Sun, S.; Li, Y.; Guo, J.; Wang, S. Tourism Demand Forecasting: An Ensemble Deep Learning Approach. *arXiv* **2021**, arXiv:2002.07964. [CrossRef]
27. Lim, B.; Zohren, S. Time-series forecasting with deep learning: A survey. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2021**, *379*, 20200209. [CrossRef] [PubMed]
28. Colladon, A.F.; Guardabascio, B.; Innarella, R. Using social network and semantic analysis to analyze online travel forums and forecast tourism demand. *Decis. Support Syst.* **2019**, *123*, 113075. [CrossRef]
29. Liu, X.; Liu, A.; Chen, J.L.; Li, G.; Song, H. Tourism Demand Forecasting in Normal and Crisis Times: Combining Bootstrap-Aggregating and Bayesian Approaches. *J. Hosp. Tour. Res.* **2025**, 10963480251313492. [CrossRef]
30. Nguyen-Da, T.; Li, Y.M.; Peng, C.L.; Cho, M.Y.; Nguyen-Thanh, P. Tourism demand prediction after COVID-19 with deep learning hybrid CNN–LSTM—Case study of Vietnam and provinces. *Sustainability* **2023**, *15*, 7179. [CrossRef]
31. Diao, T.; Wu, X.; Yang, L.; Xiao, L.; Dong, Y. A novel forecasting framework combining virtual samples and enhanced Transformer models for tourism demand forecasting. *arXiv* **2025**, arXiv:2503.19423. [CrossRef]
32. Li, X.; Xu, Y.; Law, R.; Wang, S. Enhancing tourism demand forecasting with a transformer-based framework. *Ann. Tour. Res.* **2024**, *107*, 103791. [CrossRef]
33. Yi, S.; Chen, X.; Tang, C. Time series transformer for tourism demand forecasting. *Sci. Rep.* **2025**, *15*, 29565. [CrossRef] [PubMed]
34. Box, G.E.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
35. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
36. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In *KDD'16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; Association for Computing Machinery: New York, NY, USA, 2016; pp. 785–794.
37. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
38. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/math14050758

mathematics

MDPI

Article

Geospatial Clustering of GNSS Stations Using Unsupervised Learning: A Statistical Framework to Enhance Deformation Analysis for Environmental Risk Management

Daniel Álvarez-Ruiz, Alberto Sánchez-Alzola * and Andrés Pastor-Fernández

School of Engineering, University of Cádiz, 11519 Puerto Real, Cádiz, Spain;
daniel.alvarezruiz@alum.uca.es (D.Á.-R.); andres.pastor@uca.es (A.P.-F.)
* Correspondence: alberto.sanchez@gm.uca.es; Tel.: +34-956-483472

Abstract

The global expansion of continuous GNSS networks has generated large-scale spatiotemporal datasets whose analysis requires robust mathematical and statistical tools. This study introduces a geospatial, multivariate statistical framework for classifying 21,548 GNSS stations from the University of Nevada repository. The methodology integrates harmonic regression, stochastic noise modeling, quality assessment, and slope estimation into a unified feature space suitable for high-dimensional analysis. Using unsupervised learning clustering computed with our custom-developed code, based entirely on free and open-source software, we identify homogeneous station groups that reflect dominant signal properties—periodicity, noise structure, data quality, and long-term velocity—together with their spatial context. The resulting clusters exhibit strong mathematical coherence and reveal continental-scale patterns driven by seasonal forcing, tectonic regime, climatic variability, and monument stability. By grouping stations with similar statistical behavior, the proposed framework improves reference-site selection, enhances deformation-field interpretation, and supports the detection of anomalous or hazard-related behavior. Overall, this approach provides a scalable, data-driven mathematical tool for analyzing complex spatiotemporal signals and contributes to more reliable deformation modeling and environmental risk assessment.

Keywords: machine learning; GNSS; time series analysis; harmonic models

MSC: 62H30; 62M10; 86A32

1. Introduction

In recent years, the deployment of permanent GNSS positioning networks has undergone a remarkable development [1,2]. Since the first permanent stations of international networks such as International GNSS Service (IGS) and European Reference Frame (EUREF) at the end of the 20th century, more and more countries have denser and more widespread positioning networks [3]. Currently, the registration of stations exceeds the number of 20,000 and their data are available in free databases on the Internet of institutions and governments around the world. These stations have an extensive range of accuracy and are used for make reference frames for applications in geosciences [4].

The widespread use of permanent GNSS networks has been vital to geodynamic advances in determining crustal deformation, earthquake study, and active volcanism [1,5,6].

In areas with large numbers of permanent stations such as North America, Europe and Japan, dense networks of permanent GNSS stations have been of great help in the modeling of seismic and volcanic risk by institutions and countries [5,6].

Accurate characterization of GNSS time series is essential for obtaining reliable deformation models, as their quality is influenced by multiple factors including noise level, periodicity, data gaps, time-series length, and station stability [7,8]. Stations affected by horizon obstructions often exhibit elevated noise levels [9], while those installed on buildings or elevated structures may be impacted by thermal expansion or local monument deformation. These site-specific effects must be carefully assessed to ensure that geodynamic interpretations derived from GNSS observations remain robust and scientifically defensible [10].

Recent research has placed considerable emphasis on improving the stochastic modeling of continuous GNSS (CGPS) time series. Several studies have demonstrated the importance of accounting for temporally correlated and heteroscedastic noise to enhance the reliability of velocity estimates and the detection of subtle geophysical signals [11]. More sophisticated stochastic frameworks, such as autoregressive (AR) and generalized autoregressive conditional heteroscedasticity (GARCH) models, have been proposed to better capture the complex noise behavior observed in GNSS data [2]. Additionally, the spatial variability of noise characteristics and the influence of monument stability and environmental conditions on the stochastic properties of GNSS time series have been increasingly recognized [12].

Beyond these stochastic considerations, continuous GNSS time series are also affected by strong periodic signals, equipment-related discontinuities (e.g., antenna or receiver changes), and local site instabilities. Focusing on noise, this can be classified according to its spectral properties: white noise exhibits no temporal correlation, whereas real GNSS observations commonly contain colored noise components such as flicker and random walk. Flicker noise reflects long-term temporal correlations, while random-walk noise introduces cumulative deviations that may bias long-term displacement estimates [13]. Data gaps and local deformations further distort the temporal structure of the signal, complicating the extraction of geophysical information. Proper identification and modeling of these effects are therefore crucial to ensure the robustness and geodynamic reliability of GNSS-based interpretations [14].

Statistics provides essential tools for the analysis of geodetic time series [15]. Multivariate statistical approaches, including unsupervised machine-learning techniques, can be applied to group and classify stations according to the quality of their time series [16–18]. Such classifications offer valuable support to users by enabling a clearer assessment of station quality based on the cluster to which each station is assigned.

Cluster analysis and discriminant analysis are widely used statistical tools in the processing of GNSS time series, aimed at identifying patterns, classifying stations, and improving geodynamic interpretations [19,20]. Cluster analysis enables the grouping of geodetic stations that exhibit similar behavior in terms of displacement, noise levels, or periodicities, without requiring prior information about existing categories [14]. This is particularly useful for detecting regions with homogeneous tectonic responses or identifying stations affected by similar local conditions [21]. In contrast, discriminant analysis constructs functions that separate previously defined groups, facilitating the classification of new stations or validating groupings obtained through other methods [16]. When applied together, these methods provide valuable insights into the spatial coherence of geodetic motion, the quality of the time series, and the characterization of local or regional geophysical phenomena [22].

https://doi.org/10.3390/math14050855

Despite the significant advances achieved by existing statistical and machine-learning approaches applied to GNSS time series, several limitations remain when addressing large-scale and heterogeneous networks. Many state-of-the-art studies focus on specific aspects of the signal—such as stochastic noise modeling [8], seasonal signal extraction [23], or velocity estimation—often relying on sophisticated or computationally demanding methods applied to restricted regions or limited numbers of stations. While these approaches provide high-quality results at local or regional scales, their complexity can reduce transparency, reproducibility, and scalability when extended to continental or global datasets.

Furthermore, previous classification and clustering efforts frequently rely on a reduced set of variables or treat data quality, noise characteristics, and deterministic signals separately, limiting the construction of an integrated and interpretable representation of station behavior. This fragmentation complicates cross-regional comparisons and restricts the practical use of classification results for network management, reference-station selection, and large-scale geodynamic interpretation [8,11]. These considerations motivate the need for a unified, robust, and reproducible multivariate framework that deliberately favors well-established and interpretable statistical models, and that can be systematically applied to tens of thousands of GNSS stations using freely available software tools.

In this study, we develop and apply a scalable multivariate statistical framework for the systematic classification of 21,548 GNSS stations listed in the University of Nevada database (https://geodesy.unr.edu/, accessed on 24 February 2026). The methodology is based on four key parameters—noise, slope, periodicity, and quality—directly derived from the harmonic modeling of GNSS time series and representative of their mathematical and physical properties.

Python scripts were used to automatically retrieve and organize the stations by country, followed by a filtering procedure to remove sites with known issues, short observational records, or anomalous values. The periodicity, stochastic noise characteristics, quality indicators, and long-term velocity (slope) were then computed using custom R scripts specifically developed for this work. These descriptors were integrated into a unified feature space enabling large-scale unsupervised analysis. Clustering techniques were applied to identify coherent and physically interpretable station families, and discriminant analysis was used to evaluate cluster separability.

The principal contributions of this work include (i) the construction of quantitative variables capturing periodic behavior, noise structure, instrumental stability, and long-term deformation trends; (ii) the identification of statistically coherent cluster families reflecting dominant signal characteristics; and (iii) the demonstration that these clusters exhibit spatial patterns governed by geodynamic, climatic, and operational factors. Overall, the study shows that multivariate statistical classification provides a transferable framework for improving the interpretation, reliability, and practical use of GNSS time series in geophysical applications.

This research addresses the following questions:

- RQ1: Is it possible to classify GNSS stations into homogeneous groups using time-series slope, periodicity, noise, and quality indicators?
- RQ2: Is there a relationship between the geographic location of stations and their statistical classification?
- RQ3: Can the cluster profiles of CGNSS stations serve as a statistical framework to enhance deformation analysis and risk management?

https://doi.org/10.3390/math14050855

2. Background and Theoretical Context

Continuous GNSS time series contain a mixture of deterministic signals and stochastic noise whose proper characterization is essential for extracting reliable geophysical information. This section introduces the theoretical foundations required to interpret these components, covering the harmonic modeling of periodic signals, the main noise processes affecting CGNSS data, and the quality factors that influence the stability and reliability of long-term geodetic records.

2.1. Harmonic Regression Model for Periodic GNSS Signals

In this study, the periodic components of GNSS time series are modeled using a linear harmonic regression framework. This model is designed to represent regular oscillations in the signal associated with cyclic environmental and geophysical influences. The general form of the model for a GNSS coordinate component $y(t)$ is:

$$y(t) = \beta_0 + \beta_1 t + a_1 \sin(2\pi t) + b_1 \cos(2\pi t) + a_2 \sin(4\pi t) + b_2 \cos(4\pi t) + \delta \cdot J(t) + \varepsilon(t), \quad (1)$$

where

- t denotes time expressed in decimal years,
- β_0 and β_1 represent the intercept and the linear trend (velocity),
- a_1 and b_1 correspond to the annual harmonic component,
- a_2 and b_2 correspond to the semiannual harmonic component,
- $J(t)$ is an indicator function that takes the value 1 when a discontinuity (jump) occurs at time t, and 0 otherwise,
- δ represents the magnitude of the discontinuity,
- $\varepsilon(t)$ denotes the stochastic residual component (noise).

In practice, the jump indicator function $J(t)$ is implemented using a tolerance window around known change epochs, such as antenna replacements or monumentation modifications. When multiple discontinuities are present, the model is extended by introducing several indicator functions $J_1(t), J_2(t), \ldots$, each associated with a corresponding jump coefficient $\delta_1, \delta_2, \ldots$, allowing the model to account for multiple abrupt changes in the GNSS time series.

2.2. Noise in CGNSS Time Series

To properly model and interpret GNSS time series, it is essential to understand the different types of noise that may affect the data. These noise processes differ in their temporal correlation structure and spectral properties, which have important implications for velocity estimation, uncertainty quantification, and geophysical interpretation.

2.2.1. White Noise

Definition 1 (White Noise)**.** *Let $\{\varepsilon_t\}_{t\in\mathbb{Z}}$ be a sequence of random variables defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. The sequence $\{\varepsilon_t\}$ is said to be a white noise process if it satisfies the following conditions:*

1. *Zero mean:*
$$\mathbb{E}[\varepsilon_t] = 0, \quad \forall t \in \mathbb{Z}.$$
2. *Constant variance (homoscedasticity):*
$$Var[\varepsilon_t] = \sigma^2 < \infty, \quad \forall t \in \mathbb{Z}.$$

https://doi.org/10.3390/math14050855

3. No temporal correlation (lack of memory):

$$Cov(\varepsilon_t, \varepsilon_s) = \mathbb{E}[\varepsilon_t \varepsilon_s] = \begin{cases} \sigma^2, & \text{if } t = s, \\ 0, & \text{if } t \neq s. \end{cases}$$

Remark 1. *If, in addition, the variables ε_t are independent and identically distributed (i.i.d.), then the process is called i.i.d. white noise. In the particular case where the distribution is normal $\mathcal{N}(0, \sigma^2)$, the process is referred to as Gaussian white noise: ε_t i.i.d. $\sim \mathcal{N}(0, \sigma^2)$ [24].*

White noise is a weakly stationary stochastic process characterized by constant mean and variance, and zero autocorrelation between time points. Its power spectral density is flat, indicating equal energy across all frequencies and the absence of periodic or temporal structure. Due to the lack of correlation, white noise is inherently unpredictable. Figure 1 shows the CGNSS station LINO, located in Great Britain on the stable western margin of the Eurasian plate. The vertical component (Up) exhibits high white noise levels, minimal trend, and clear annual periodicity. We estimated a white noise amplitude of $(5.43 \pm 0.11) \times 10^{-3}$ m.

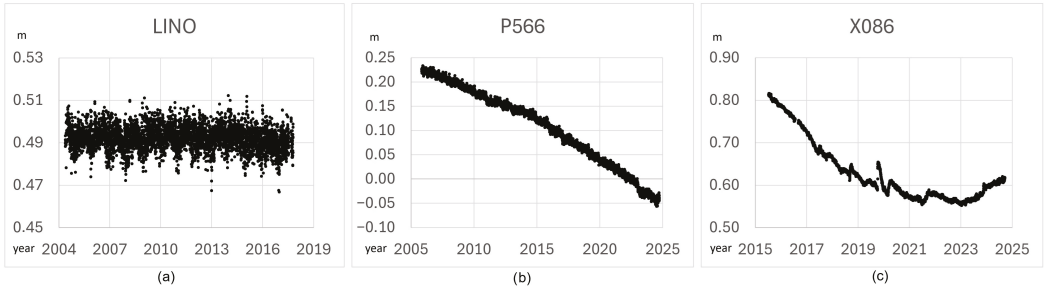


Figure 1. Types of noise observed in CGNSS time series are illustrated: (**a**) Up component with high white noise levels: CGNSS station LINO, Great Britain. (**b**) East component with high random walk noise: CGNSS station X086, Okinawa, Japan. (**c**) Up component with flicker noise CGNSS station P566, California USA.

2.2.2. Random Walk Process

Definition 2 (Random Walk Process)**.** *Let $\{X_t\}_{t \in \mathbb{Z}}$ be a stochastic process defined recursively by:*

$$X_t = X_{t-1} + \varepsilon_t, \quad \text{with } \varepsilon_t \sim WN(0, \sigma^2),$$

where $\{\varepsilon_t\}$ is a white noise process with zero mean and constant variance σ^2.

The process $\{X_t\}$ is called a random walk process, also known as cumulative noise. It can be expressed as:

$$X_t = X_0 + \sum_{k=1}^{t} \varepsilon_k.$$

The random walk process is a non-stationary stochastic model characterized by constant mean but linearly increasing variance over time. Its values exhibit persistent temporal dependence, as past perturbations accumulate and influence future states. Consequently, the autocovariance depends not only on the lag but also on the time index [25].

Spectrally, the process concentrates energy in low frequencies, with a power spectral density that decays approximately as the inverse square of frequency—typical of red or Brownian noise. This long-range memory implies that distant past values can still affect current observations. Such behavior is especially relevant in geodetic applications, where

https://doi.org/10.3390/math14050855

cumulative effects like tectonic drift or sensor instability can bias trend estimations if not properly accounted for. Identifying random walk noise is therefore essential for reliable GNSS time series analysis. Figure 1 shows station CGNSS X086, located in Okinawa (Japan) on an active tectonic boundary. The East component exhibits high random walk noise, with no clear linear trend and strong autocorrelation. We estimated a random walk noise amplitude of $(5.13 \pm 0.13) \times 10^{-2}$ m.

2.2.3. Flicker Noise

Definition 3 (Flicker Noise)**.** *Let $\{X_t\}_{t\in\mathbb{Z}}$ be a zero-mean stochastic process with finite variance over finite time intervals. Denote*

$$\gamma(\tau) = \mathbb{E}[X_t X_{t+\tau}],$$

where $\gamma(\tau)$ denotes the (formal) autocovariance function of the process, whenever it exists.

The power spectral density (PSD) of the process is defined as the discrete Fourier transform of the autocovariance function [26,27]:

$$f(\lambda) = \frac{1}{2\pi} \sum_{\tau=-\infty}^{\infty} \gamma(\tau)\, e^{-i\tau\lambda}, \qquad \lambda \in [-\pi, \pi].$$

The process $\{X_t\}$ is said to exhibit flicker noise (or $1/f$ noise) if its power spectral density satisfies the scaling law

$$f(\lambda) \propto \frac{1}{|\lambda|},$$

for frequencies λ within a finite interval $\lambda \in [\lambda_{\min}, \lambda_{\max}]$.

Flicker noise represents an intermediate stochastic behavior between white noise and random walk noise. Unlike white noise, it exhibits long-range temporal dependence, while, in contrast to random walk processes, its variance does not grow unbounded with time. From a theoretical perspective, an ideal $1/f$ spectrum is not integrable at the origin, implying that a pure flicker noise process is not strictly stationary. In practical applications, flicker noise is therefore interpreted as an approximate spectral behavior observed over a limited frequency band.

Spectrally, flicker noise concentrates more energy at low frequencies than white noise but less than random walk noise. This slow decay of the spectrum reflects persistent correlations across multiple time scales, making flicker noise particularly relevant in geodetic time series, where it can significantly bias velocity estimates if neglected. Figure 1 shows the CGNSS station P566, where the vertical component exhibits a clear $1/f$-type spectral decay, indicative of flicker noise. We estimated a flicker noise amplitude of $(3.71 \pm 0.07) \times 10^{-3}$ m.

2.2.4. Noise Estimation Methodology

The following noise estimates should be interpreted as empirical noise metrics derived from the residual series, rather than as strict identification of ideal stochastic processes. The stochastic component $\varepsilon(t)$ in Equation (1) is estimated from the residuals obtained after fitting the deterministic harmonic regression model using ordinary least squares (OLS), a standard approach in GNSS time-series analysis [28]. Three stochastic noise processes are quantified: white noise, flicker (low-frequency) noise, and random walk noise. Each component is estimated using complementary time-domain and frequency-domain methods applied to the regression residuals.

- White noise amplitude is computed as the standard deviation of the residual series, representing the high-frequency uncorrelated component.

https://doi.org/10.3390/math14050855

- Flicker (low-frequency) noise is evaluated in the spectral domain using the power spectral density (PSD) of the residuals. The noise level is approximated from the spectral energy concentrated in the lowest frequency band, capturing long-period correlated variability consistent with the $1/f$ behavior commonly observed in geodetic time series.
- Random walk noise is estimated by fitting an ARIMA(0, 1, 0) model to the residual series using maximum likelihood estimation. The square root of the innovation variance of the fitted model provides an estimate of the random walk noise amplitude.

We note that the adequacy of the noise characterization was assessed through inspection of the residual autocorrelation structure and spectral behavior. The identification of white noise, flicker noise, and random walk noise therefore relies on their expected statistical and spectral signatures. Several factors may affect the accuracy of noise quantification, including limited time-series length, data gaps, imperfect modeling of discontinuities, seasonal signal leakage into the residuals, and irregular temporal sampling. These effects may introduce bias in spectral estimates and influence the clustering results derived from noise-related features. Despite these limitations, this empirical characterization provides a consistent and scalable framework for comparing noise behavior across thousands of CGNSS stations.

2.3. Periodicity in CGNSS Time Series

GNSS time series typically exhibit both deterministic and stochastic components. Among the deterministic components, periodic signals play a central role in modeling the geophysical behavior of the stations. These signals are often associated with environmental effects such as seasonal loading (hydrological, atmospheric or snow-related), thermal expansion of the monumentation, and systematic errors in satellite orbits or equipment calibration [29]. From a mathematical perspective, periodicity in the GNSS signal is often represented using harmonic components, which are modeled as sinusoidal functions with specific frequencies and phases. The most prominent periodicities observed in CGNSS data are the annual and semiannual cycles [23].

In this study, only the annual and semiannual harmonic components were considered in the regression model because they represent the dominant and most stable seasonal signals consistently observed in CGNSS time series [30,31]. Higher-order harmonics generally exhibit smaller amplitudes, lower temporal stability, and greater sensitivity to data gaps and noise contamination. Including additional frequencies would therefore increase model complexity without significantly improving the characterization of periodic behavior in large-scale GNSS station datasets. This parsimonious representation is commonly adopted in GNSS time-series modeling.

Figure 2 illustrates representative examples of the different types of periodic behavior observed in the CGNSS time series. These cases highlight how stations may exhibit low periodic content, strong annual signals, or pronounced semiannual components depending on their geophysical and environmental context. In Equation (1), the sinusoidal terms of the model represent the annual and semi-annual periodic components of the time series.

https://doi.org/10.3390/math14050855

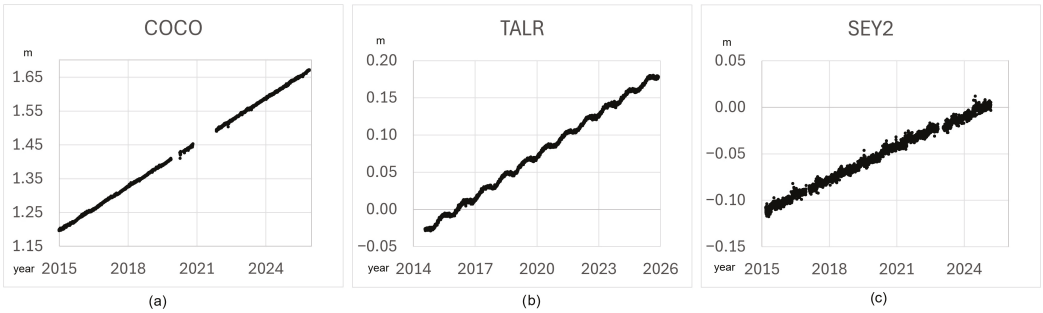


Figure 2. Types of periodicity observed in CGNSS time series are illustrated: (**a**) North component with low periodicity levels: CGNSS station COCOS, Cocos Keeling Island. (**b**) East component with high annual periodicity: CGNSS station TALR, Spain. (**c**) North component with high semiannual periodicity: CGNSS station SEY2, Seychelles.

2.4. Quality in CGNSS Time Series

In this study, the quality of continuous GNSS (CGNSS) time series is formally defined as a set of structural and observational descriptors that characterize the completeness, continuity, and instrumental stability of each station record. Specifically, the variables considered are time series length, data availability, and the presence of offsets. These quality descriptors are treated as indicators of dataset integrity and are mathematically independent from the stochastic noise metrics and deterministic signals used in other analyses.

Offsets arising from equipment changes, such as antenna or radome replacements, represent a major factor affecting CGNSS time series quality (Figure 3). These discontinuities introduce artificial jumps that, if not properly documented and corrected, may bias velocity estimates and long-term geophysical interpretations. Studies within the EUREF Permanent Network identify antenna-related changes as a primary source of offsets and highlight the importance of systematic metadata tracking and detection strategies [32,33].

Data gaps also reduce time series quality by limiting observational continuity and availability. Although missing observations can influence stochastic noise estimation—particularly colored noise components such as flicker and random walk [34]—they are considered here exclusively as indicators of observational completeness. Similarly, the length of the GNSS time series constrains the reliability of velocity estimation, seasonal signal separation, and stochastic interpretation, with longer records generally enabling more robust parameter estimation [34]. Consequently, time series length is incorporated as a fundamental descriptor of observational capacity.

To maintain consistency and comparability in the clustering framework, each quality descriptor is normalized and optionally weighted before integration into the multivariate analysis. Offsets are coded as the number of documented equipment changes, data availability is expressed as number of observations, and series length is normalized to a fixed scale. In this way, the quality feature contributes proportionally to cluster assignments without dominating or being dominated by noise metrics [35]. This ensures that the four primary feature domains—slope, periodicity, noise, and quality—remain conceptually distinct and separable in the feature space.

https://doi.org/10.3390/math14050855

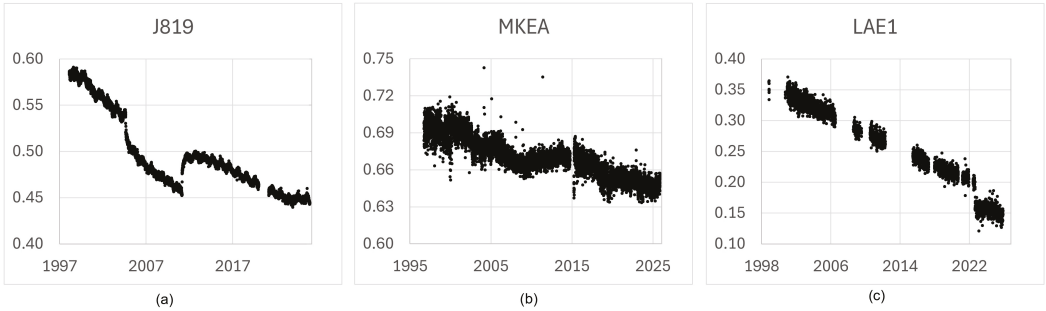


Figure 3. Types of low quality observed in CGNSS time series are illustrated: (**a**) North component with earthquake effect: CGNSS station J819, Japan. The time series show leaps in 2004 and 2011 earthquakes. (**b**) Up component with lots of antenna changes: CGNSS station MKEA, Hawaii (USA). (**c**) Up component with lots of gaps and clear change of antenna in 2022: CGNSS station LAE1, Papua New Guinea.

3. Materials and Methods

This study followed a systematic methodology encompassing the automated download, processing, filtering, and analysis of GNSS time series data. The steps carried out involved an initial automated download of the GNSS Data. Next, it was necessary to label the stations using the countries, followed by adaptation of the file to the format of the SARI program [36]. Afterwards we computed the Slopes, Periodicities, and Noise Levels of the stations with a data filtering and cleaning for a better analysis. Eventually we applied a statistical descriptive analysis with several multivariate techniques such as clustering and discriminant analysis. Below, we describe the whole procedure in detail.

3.1. Preprocessing and Download of the GNSS Data

Our work began with an automated download of the GNSS Data from (https://geodesy.unr.edu/, accessed on 24 February 2026), a public repository provided by the UNAVCO/Nevada Geodetic Laboratory. A total of 21,548 geodetic stations were downloaded globally. For that we used a Python script to read the time series files for East, North, and Up components. The script utilized web scraping techniques through the requests library to identify all accessible stations and systematically download the corresponding files. The downloaded files were adapted into a custom .tenv format to facilitate subsequent organization and processing.

Downloaded files were locally organized into country-specific folders, based on each station's identification code. This classification was performed using information data contained in each station's filename or download URL. Grouping by country enabled regional comparative analyses and facilitated quality control processes. The number of stations per country varies depending on the different development of GNSS networks in each of them. Regions such as Europe, North America, and active seismic zones boast denser networks than less populated and poor areas.

Later we considered the philosophy of analysis in a rigorous and widely used program for determining parameters in geodetic time series such as SARI (https://sari-gnss.github.io/, accessed on 24 February 2026). The ".tenv" files were converted into a format similiar to the used with this software. The conversion process involved rewriting the data into columns of decimal dates and displacements (East, North, Up) following the required order; ensuring uniformity in the number of columns and data formatting.

https://doi.org/10.3390/math14050855

3.2. Computation of Velocities, Periodicities, and Noise Levels

The computational workflow was implemented using both Python (version 3.12.4) and R-Studio (version 2026.01.1+403). Python was used for automated data retrieval and dataset construction. In particular, the libraries `requests`, `pandas`, `os`, and `time` were used to download GNSS time-series files from the Nevada Geodetic Laboratory repository, organize them into country-based directories, and generate structured input datasets for subsequent analysis. The estimation of velocities, seasonal components, noise metrics, filtering procedures, clustering analyses, and visualizations were implemented in R (version 4.4.2). Data manipulation and preprocessing were performed using the `tidyverse` ecosystem, while clustering and statistical analyses relied on the packages `stats`, `cluster`, `MASS`, `clusterSim`, `mclust`, and `e1071`. Visualization and mapping were carried out using `ggplot2`, `leaflet`, and `sf`. All scripts used in the workflow were developed by the authors to ensure reproducibility of the analysis.

Building on the workflow described above, we developed a second script in R to extract the key geodetic parameters from the GNSS time series. First, we fitted a linear model to each time series to estimate the velocities in the East, North, and Up components. In this model we also incorporated sinusoidal terms with annual and semi-annual frequencies to capture the seasonal behavior of the displacements following what was described in Section 2.1. This approach allowed us to characterize three main noise components: white noise, colored noise, and random walk noise, representing. Finally, we organized all the estimated parameters for each station into a structured table, ensuring the data was ready for subsequent analysis.

Following the parameter estimation, we compiled the results into a structured database to facilitate further analysis. In this table, each row represents an individual GNSS station, while each column corresponds to one of the calculated variables, including velocities, seasonal amplitudes, noise levels, and the length of the time series. This organization enables efficient statistical evaluations and comparative studies across the entire dataset.

To ensure the robustness and reliability of the analyses, a multi-step data filtering pipeline was implemented over our 21,548 stations downloaded. Initially, only records from 2020 onwards were considered to standardize the time series and ensure they have approximately the same length, staying 15,661 stations. Afterwards, we note that some records containing missing values in the velocity components (East, North, and Up) and only complete observations were considered. Also, stations with physically reasonable slope values (between -10 and 10 mm yr^{-1}) were retained. Finally, stations with time series shorter than three years were also excluded, since shorter observations do not allow for a reliable estimation of seasonal periodicities. After this basic filtering stage, 12,383 stations out of the 15,661 were retained.

Subsequently, multivariate outliers among the velocity components were identified using the Mahalanobis distance, computed from the covariance structure of the East, North, and Up trends. Stations exceeding the 99th percentile of the χ^2 distribution (with three degrees of freedom) were considered anomalous and removed. It is worth noting that the basic filtering step is univariate, as it only considers the velocity components and the length of the time series. However, this approach does not account for atypical relationships among the three velocity components (East, North, and Up). Therefore, an additional multivariate filtering using the Mahalanobis distance was applied. This method identifies anomalous observations based on the covariance structure of the dataset, allowing the detection of stations with inconsistent behavior among components, even if their individual values fall within acceptable limits. Consequently, this step ensures a more robust and internally consistent dataset for subsequent analyses. This step reduced the dataset from

 https://doi.org/10.3390/math14050855

12,383 to 12,290 stations, representing 78.48% of the initial set. In Table 1, the structure and variables of the four datasets can be visualized, providing an overview of the information available for subsequent analyses.

Table 1. Datasets and their corresponding fully expanded variables.

Dataset	Variables
Noise	Station, Latitude, Longitude, White Noise East, North and Up, Flicker Noise East North and Up, Random Walk Noise East, North and Up
Slope	Station, Latitude, Longitude, East Slope, North Slope, Up Slope
Period	Station, Latitude, Longitude, Annual East, North and Up Amplitudes and Phases, Halfyear East, North and Up Amplitudes and Phases.
Quality	Station, Latitude, Longitude, Time Series Length, Data Count, Offset Count

Note: The variables Station, Latitude, and Longitude are not used in any computation; they are included only as informational metadata.

Finally, an interquartile range (IQR) criterion was independently applied to each derived dataset (noise parameters, velocity components, periodic terms, and quality-related metrics) in order to identify and remove univariate outliers. In the case of the slope dataset, absolute values were used prior to the IQR filtering. After this filtering step, the retained samples consisted of 8816 records for the noise dataset (56.29% of the original data), 9197 for the slope dataset (58.73%), 8444 for the periodic terms dataset (53.92%), and 7921 for the quality dataset (50.58%). This additional filtering ensured that only statistically consistent and physically plausible values were preserved for subsequent analyses. In Figure 4, the percentage of data retained after each filtering stage for three datasets is shown. An additional figure related to the filtering procedure is provided in the Supplementary Materials (Figure S1).

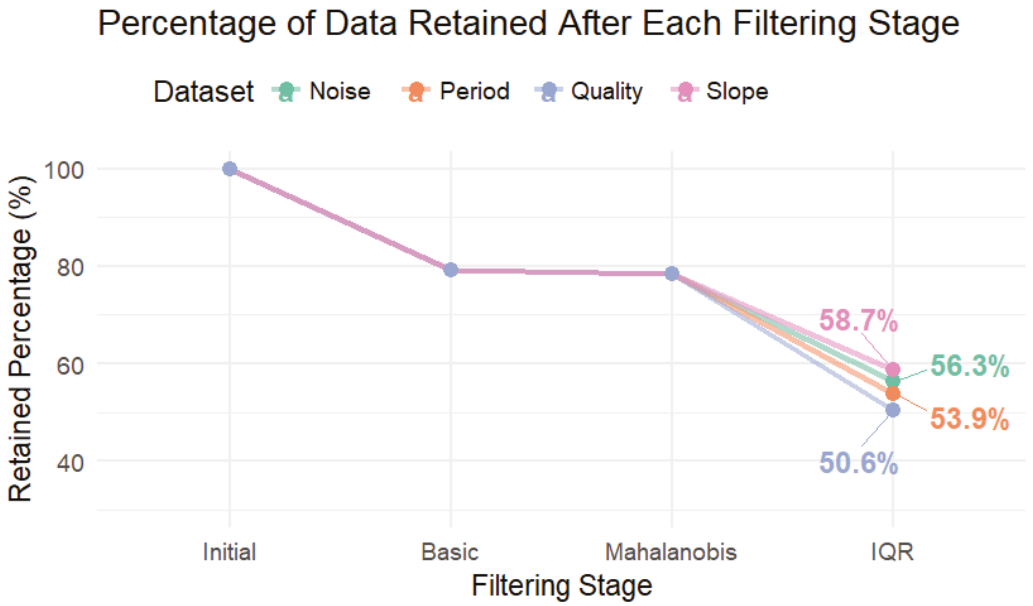


Figure 4. Percentage of data retained after each filtering stage for four datasets: `Noise`, `Quality`, `Period` and `Slope`.

https://doi.org/10.3390/math14050855

After the data filtering stages illustrated in Figure 4, the clustering process was prepared by selecting the optimal number of groups for each dataset. To this end, a robust version of the elbow method was implemented, which evaluates the within-cluster sum of squares (WSS) for increasing values of k. The optimal number of clusters (k_{opt}) was identified as the point with the maximum perpendicular distance from the straight line connecting the first and last WSS values. The elbow method was applied independently to the four datasets (periodicity, noise, quality and slope) to determine the optimal number of clusters. In Figure 5 we show the results of the elbow method in each dataset and the optimal number of clusters needed.

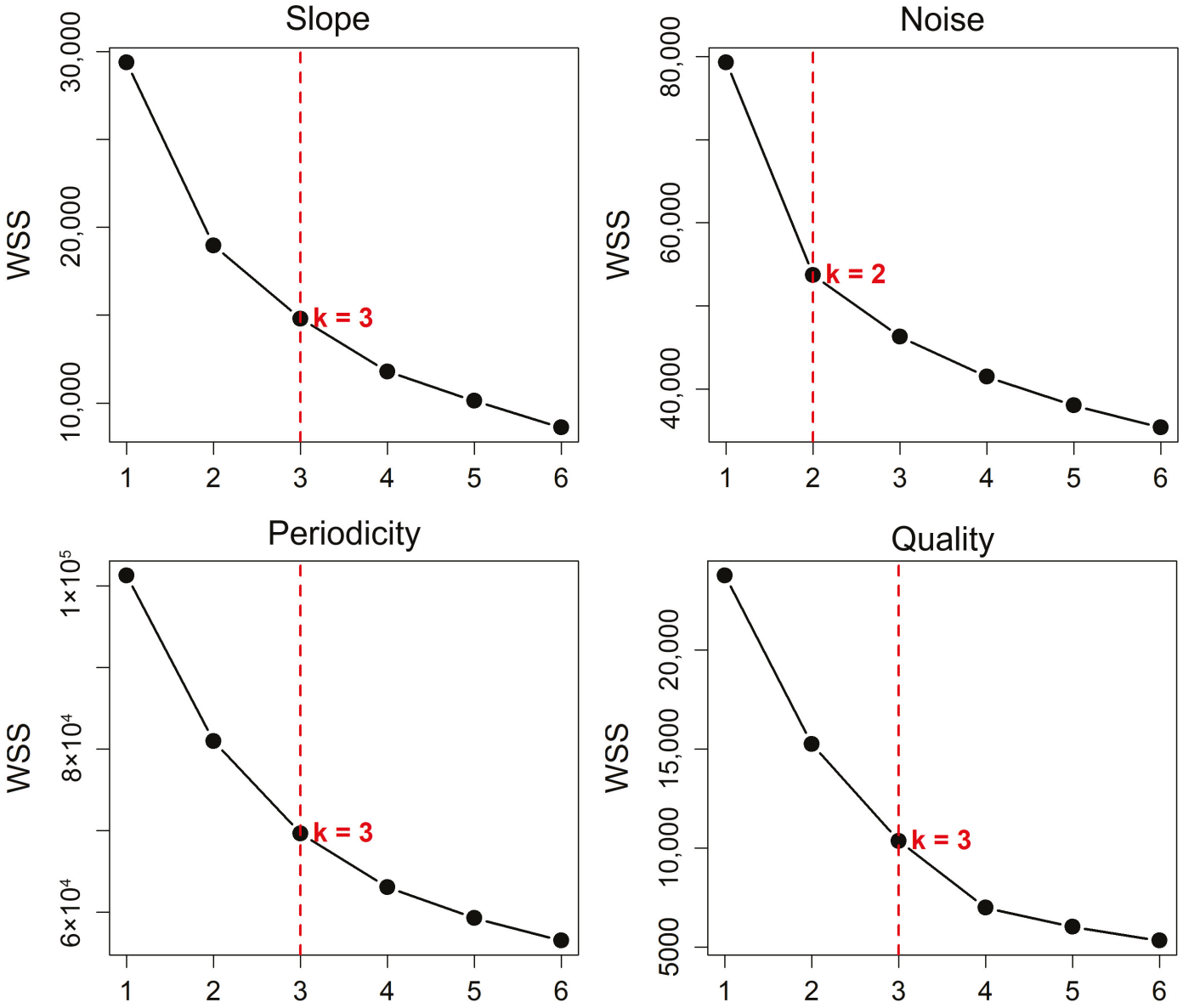


Figure 5. Elbow plots for the four GNSS signal components—slope, noise, periodicity, and quality—used to determine the suggested number of clusters.

We found that the appropriate number of clusters for grouping almost all the analysed datasets is three, with the exception of noise, for which the algorithm indicates that the optimal number of groups is two (stations with higher and lower noise levels). To evaluate the robustness of the clustering, we compared the $k = 2$ and $k = 3$ solutions for the noise

https://doi.org/10.3390/math14050855

dataset. While $k = 2$ captures the main contrast between low-noise and high-noise levels, it merges distinct geophysical behaviors into overly broad groups. The $k = 3$ configuration provides a clearer and more informative separation, revealing an intermediate cluster with coherent spatial and geophysical patterns. For this reason, and considering that the elbow method provides only a heuristic and orientative indication of the appropriate number of clusters, we adopt $k = 3$ as the preferred solution.

3.3. Clustering Analysis

To explore structural characteristics within the GNSS-derived datasets (noise parameters, periodic terms, general quality and velocity components), we applied five distinct clustering algorithms to each dataset independently. The clustering procedures were implemented in R, using standardized versions of the data matrices to ensure comparability across variables. The selected methods are detailed below.

In this study, several clustering approaches were applied and organized according to their hierarchical or non-hierarchical nature, as well as their partition-based or probabilistic formulation, in order to capture different latent structures in the data and to evaluate the robustness of the results.

Among the non-hierarchical partition-based methods, the k-means algorithm was implemented as a distance-based approach that minimizes the within-cluster sum of squares. To reduce sensitivity to initial conditions and to ensure convergence toward a stable solution, 50 random initializations were performed (*nstart* = 50). In addition, the Partitioning Around Medoids (PAM) algorithm was used as a robust alternative to k-means. By selecting actual observations as cluster centers (medoids), PAM improves interpretability and provides increased resistance to the influence of outliers.

A hierarchical clustering approach was also employed using Ward's minimum variance criterion (*ward.D2* option). This agglomerative method constructs a dendrogram based on pairwise distances between observations, which was subsequently cut at a predefined number of clusters. The resulting dendrograms were saved to allow for visual inspection of the hierarchical structure and relationships among clusters.

Finally, non-hierarchical fuzzy and probabilistic methods were considered. Fuzzy C-means clustering was applied to allow partial membership of each observation across clusters, with the fuzziness parameter set to $m = 2$, following the classical formulation of Bezdek (1981) [37], as it provides a well-established balance between membership fuzziness and centroid definition. The resulting membership matrices were exported for further analysis. In parallel, Gaussian Mixture Models (GMMs) were fitted as a fully probabilistic framework, assuming that the data were generated from a mixture of multivariate normal distributions. The optimal number of components was fixed at k_{opt}, and Bayesian Information Criterion (BIC) values were recorded to assess and compare model fit.

For each method, the cluster assignments and cluster centers (or medoids) were extracted and stored. These results enabled comparative evaluations of clustering consistency and facilitated subsequent geospatial and statistical interpretations.

3.4. Clustering Evaluation Criteria

To assess the quality and consistency of the clustering results, several internal validation metrics were computed for each method and dataset. These metrics provide complementary perspectives on the compactness, separation, and discriminative power of the resulting clusters. Following the recommendation of reference [38], three complementary cluster evaluation criteria were selected in order to determine the most suitable clustering method for the available data. These indices were chosen because they jointly

https://doi.org/10.3390/math14050855

assess cluster compactness, separation, and overall structural validity, providing a balanced framework for comparing alternative clustering solutions.

The Average Silhouette Width was used to quantify both cohesion and separation of the resulting clusters. Values close to 1 indicate well-separated and compact clusters, whereas values near 0 suggest overlapping or ambiguous cluster assignments. This metric allows an intuitive interpretation of clustering quality at both the global and individual observation levels. In addition, the Davies–Bouldin Index (DBI) and the Calinski–Harabasz Index (CHI) were employed as complementary validation measures. Lower DBI values indicate superior clustering performance by reflecting low intra-cluster dispersion and high inter-cluster separation. Conversely, higher CHI values are associated with better-defined clusters, as this index favors solutions that maximize between-cluster variance while minimizing within-cluster variance.

All metrics and visualizations were computed using the standardized data matrices to ensure comparability across variables. The results were stored and exported for each clustering method and dataset, enabling a comprehensive comparison of clustering performance. To quantitatively assess the performance of the clustering algorithms applied to each GNSS signal component, we computed several internal validation metrics. The results are summarized in Table 2.

Table 2. Performance metrics for the clustering methods applied to each GNSS signal component. The table reports Silhouette, Davies–Bouldin, and Calinski–Harabasz indices for all tested algorithms.

Component	Method	Silhouette	Davies–Bouldin	Calinski–Harabasz
Slope	k-means	0.406	1.109	5170.972
	PAM	0.403	1.139	5125.477
	Hierarchical	0.336	1.260	3803.265
	Fuzzy C-m	0.399	1.162	5096.332
	G-M	0.227	1.648	1994.017
Quality	k-means	0.487	1.042	5113.207
	PAM	0.492	1.082	4725.828
	Hierarchical	0.442	1.089	4209.896
	Fuzzy C-m	0.489	1.028	5079.954
	G-M	0.251	1.444	1223.629
Noise	k-means	0.347	1.643	3147.169
	PAM	0.174	1.833	2864.923
	Hierarchical	0.257	1.819	2390.510
	Fuzzy C-m	0.187	2.003	2869.742
	G-M	0.151	2.842	1549.336
Period	k-means	0.199	1.971	1921.221
	Hierarchical	0.171	2.159	1602.448
	Fuzzy C-m	0.164	2.237	1641.621
	G-M	0.134	2.285	1430.125

Regarding the four dataframes used to evaluate the performance of each GNSS station, K-means clustering generally provided the most effective grouping results compared with the other methods tested. It yielded the highest-quality outcomes for the velocity, periodicity, and noise dataframes across the three evaluation criteria. In contrast, for the dataframe describing the overall station quality, the best-performing method was Fuzzy Means, although the K-means solution produced results that were very similar and only slightly inferior.

https://doi.org/10.3390/math14050855

Reducing the number of variables can substantially alter the behavior of clustering algorithms, often favoring soft approaches such as Fuzzy C-Means (FCM) over hard methods like K-means. Lower dimensionality typically yields more compact data structures and smoother group boundaries, enabling fuzzy-membership algorithms to better capture gradual transitions between clusters. JCR studies have shown that FCM consistently achieves higher Silhouette, Davies–Bouldin, and Calinski–Harabasz scores in settings with subtle group separation or partial overlap [39,40]. Moreover, FCM tends to be more stable than K-means when the variable set is small, as its fuzzy formulation accommodates structural uncertainty and avoids rigid assignments that may be unsuitable in low-dimensional spaces [41]. These findings support the observation that, under reduced dimensionality, FCM can emerge as the best-performing clustering method, as reflected in this study.

Based on the validation indices reported in Table 2 (Silhouette, Davies–Bouldin, and Calinski–Harabasz), the clustering method with the best overall performance was selected independently for each component. Specifically, k-means was retained for Slope (Velocity), Noise, and Period, while Fuzzy C-means was selected for the Quality component. The subsequent analysis and interpretation are therefore based on these optimal configurations rather than on a joint clustering of the four quantities.

3.5. Linear Discriminant Analysis of Cluster Separability

To evaluate the separability of the clusters obtained from each clustering method, a Linear Discriminant Analysis (LDA) was performed using the `lda()` function from the `MASS` package in R. The analysis was applied to the standardized numerical variables used for clustering, with cluster assignments treated as class labels. For each clustering solution, model performance was assessed using leave-one-out cross-validation (LOOCV), from which confusion matrices and classification accuracy were computed. Discriminant projections shown in the figures correspond to the first two linear discriminant components obtained from an LDA model fitted without cross-validation and are used solely for visualization of group separation in the reduced discriminant space. LDA projections were generated only when all clusters contained at least two observations to ensure numerical stability of the discriminant model. This analysis provides a complementary evaluation of clustering quality by quantifying the degree of linear separability among the identified groups in the feature space, alongside other internal validation metrics.

3.6. Geospatial Visualization of Clustering Results

To support the interpretation of clustering outcomes, interactive geospatial maps were generated for each clustering method and dataset. These maps display the spatial distribution of GNSS stations colored by their assigned cluster, enabling visual assessment of regional patterns and potential geophysical correlations. The maps were created using the `leaflet` library in R, which allows dynamic rendering of geographic data. Each station was plotted using its latitude and longitude coordinates, and clusters were distinguished using an adaptive color palette from the `viridis` package. Pop-up labels were included to display station identifiers and cluster assignments upon interaction.

For each clustering method applied to each dataset (noise parameters, periodic terms, and velocity components), a separate map was generated and exported in HTML format. These visualizations provide an intuitive overview of how stations group spatially, revealing regional similarities, tectonic behaviors, or anomalies that may not be evident from numerical analysis alone.

In addition to the interactive maps, the R workflow also produces geospatial files in standard GIS formats—specifically ESRI Shapefiles—so that the results can be further

 https://doi.org/10.3390/math14050855

explored, edited, or integrated within QGIS 3.40 software (https://qgis.org/, accessed on 24 February 2026). This ensures full interoperability with common geospatial analysis tools and facilitates more advanced spatial queries or cartographic representations. The use of clean basemaps (`CartoDB.Positron`) and consistent styling ensures clarity and comparability across maps. Legends were added to facilitate interpretation, and all outputs were saved for further exploration and presentation.

4. Results

In this section, we present the main outcomes of the clustering analysis applied to the GNSS time series. First, centroid profiles were computed for each cluster to characterize the dominant temporal behavior of the stations across the different signal components. We then generated Linear Discriminant Analysis (LDA) projections to evaluate the separability of the clusters and to visualize the discriminant structure captured by each dataset.

In addition, geospatial maps were generated to visualize the spatial distribution of GNSS stations according to their assigned clusters, enabling detailed regional comparisons and the identification of coherent geographic or geophysical patterns. These maps were produced using R and subsequently exported in standard GIS formats, allowing further inspection, refinement, and spatial analysis within QGIS software. This integration facilitates a more comprehensive interpretation of the clustering results, particularly in regions where local tectonics, environmental conditions, or data quality may influence station behavior. The resulting geospatial products form a key component of the analysis and are examined in detail in the following subsections.

To aid the interpretation of the clustering results, each GNSS signal component was grouped according to the dominant behavior observed in its time series. For the periodic component, clusters P1, P2, and P3 capture distinct patterns of seasonal variability: P1 includes stations with a strong annual signal in the East and Up components and a weaker response in the North; P2 groups stations dominated by an annual cycle in the North component together with a pronounced semiannual contribution in the horizontal components; and P3 comprises stations with intermediate annual amplitudes horizontally but a markedly enhanced semiannual signal in the vertical direction. A similar nomenclature is applied to the remaining components—Q1–Q3 for quality, N1–N3 for noise, and S1–S3 for slope—each summarizing the main structural characteristics of the stations within their respective cluster families. These patterns are detailed in Table 3.

To further characterize the structure and separability of the clusters obtained for each GNSS signal component, we examined the centroid profiles derived from the clustering results. These profiles summarize the representative behavior of each group and provide a direct visualization of the dominant temporal patterns related to slope, noise, periodicity, and quality. In addition, Linear Discriminant Analysis (LDA) projections were generated to evaluate the discriminant power of the clustering solutions and the degree of separation in a reduced feature space. Together, these graphical representations offer a clear view of the internal consistency of the clusters and their ability to capture meaningful differences across stations. The corresponding centroid profiles and LDA projections are shown in Figures 6 and 7. Additional figures related to the cluster box plots are provided in the Supplementary Materials (Figures S2–S5).

https://doi.org/10.3390/math14050855

Table 3. Summary of the main characteristics of the clusters identified for each GNSS signal component (periodicity, quality, noise, and slope), including the number of stations, their relative proportion, and the defining features of each cluster profile.

Compon.	Cluster	N	%	Profile
Period	P1	2013	23.84%	Strong annual periodicity in East and Up; weak in North
	P2	3574	42.33%	Strong annual periodicity in North; marked semiannual in horizontal
	P3	2857	33.83%	Moderate annual periodicity in horizontal; strong semiannual in Up
Quality	Q1	956	12.07%	Very few offsets (highest quality)
	Q2	2540	32.07%	More frequent offsets
	Q3	4425	55.86%	High data completeness
Noise	N1	1692	19.19%	High random-walk and white noise
	N2	5628	63.84%	Generally low noise levels
	N3	1496	16.97%	Elevated flicker noise
Slope	S1	4334	47.12%	Low velocities in all components
	S2	1507	16.39%	Intermediate velocities; high in Up
	S3	3356	36.49%	Strong horizontal velocities

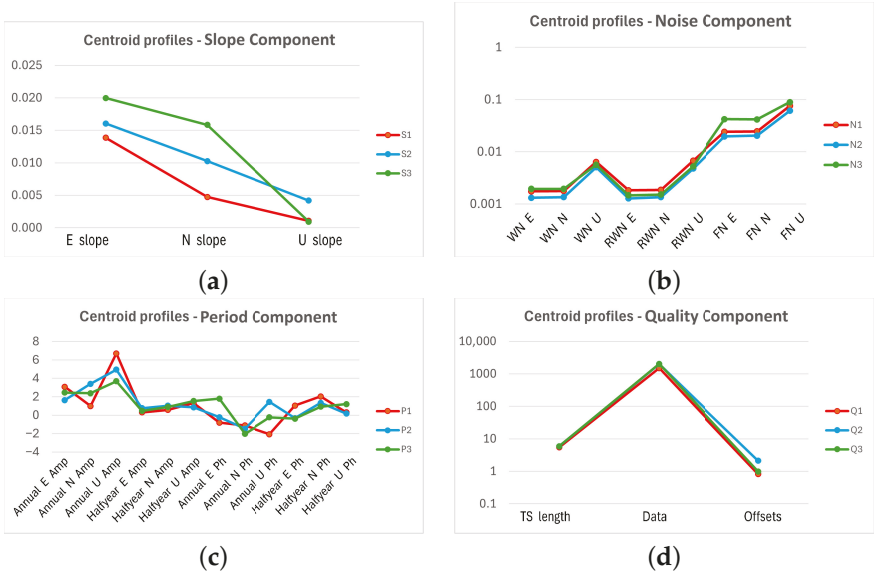


Figure 6. Centroid profiles of the clusters obtained for each GNSS signal component. Panels show the representative patterns for (**a**) slope, (**b**) noise, (**c**) period, and (**d**) quality, summarizing the characteristic behavior of each cluster.

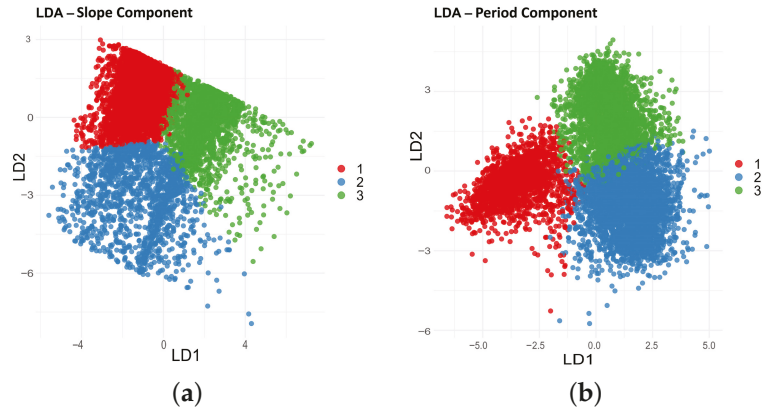
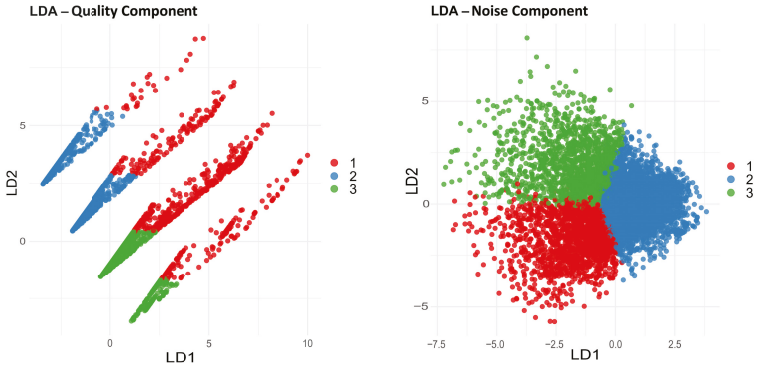


Figure 7. *Cont.*

https://doi.org/10.3390/math14050855

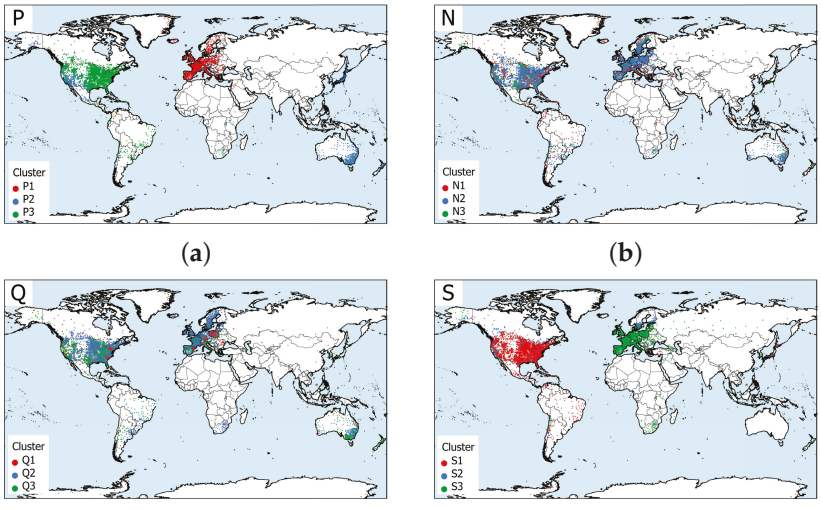


(**c**) (**d**)

Figure 7. Linear Discriminant Analysis (LDA) projections for the clustering results of each GNSS signal component: (**a**) slope, (**b**) period, (**c**) quality, and (**d**) noise.

5. Discussion

This section discusses the physical interpretation and geodynamical implications of the clustering results obtained from the analysis of GNSS time-series characteristics. By jointly considering periodicity, slope, quality, and noise, the clustering framework reveals coherent spatial patterns that reflect the combined influence of regional geophysical processes, tectonic setting, climatic forcing, and network-related factors. Rather than treating each metric in isolation, the discussion emphasizes how the identified groups capture systematic differences in crustal behavior across continents and tectonic domains. The following subsections examine each component in detail, relate the cluster distributions to established findings in the literature, and highlight how the observed patterns contribute to a more integrated understanding of global GNSS station behavior. Figure 8 displays the global station-clustering results for the four parameters analyzed. To facilitate interpretation, Figure 9 provides a detailed view focusing on specific regions, including Europe, the continental United States, Australia, New Zealand, and Japan.



(**a**) (**b**)

(**c**) (**d**)

Figure 8. Global distribution of GNSS stations classified by cluster analysis for four parameters: (**a**) periodicity, (**b**) noise, (**c**) quality, and (**d**) slope. In each map, stations are grouped into three clusters representing the dominant characteristics of the corresponding component, highlighting the spatial variability of GNSS time series behavior at the global scale. Colors are consistent across all components: the same color always represents the same cluster number (Cluster 1, Cluster 2, Cluster 3), while the P/Q/S/N labels indicate the component-specific cluster assignment.

https://doi.org/10.3390/math14050855

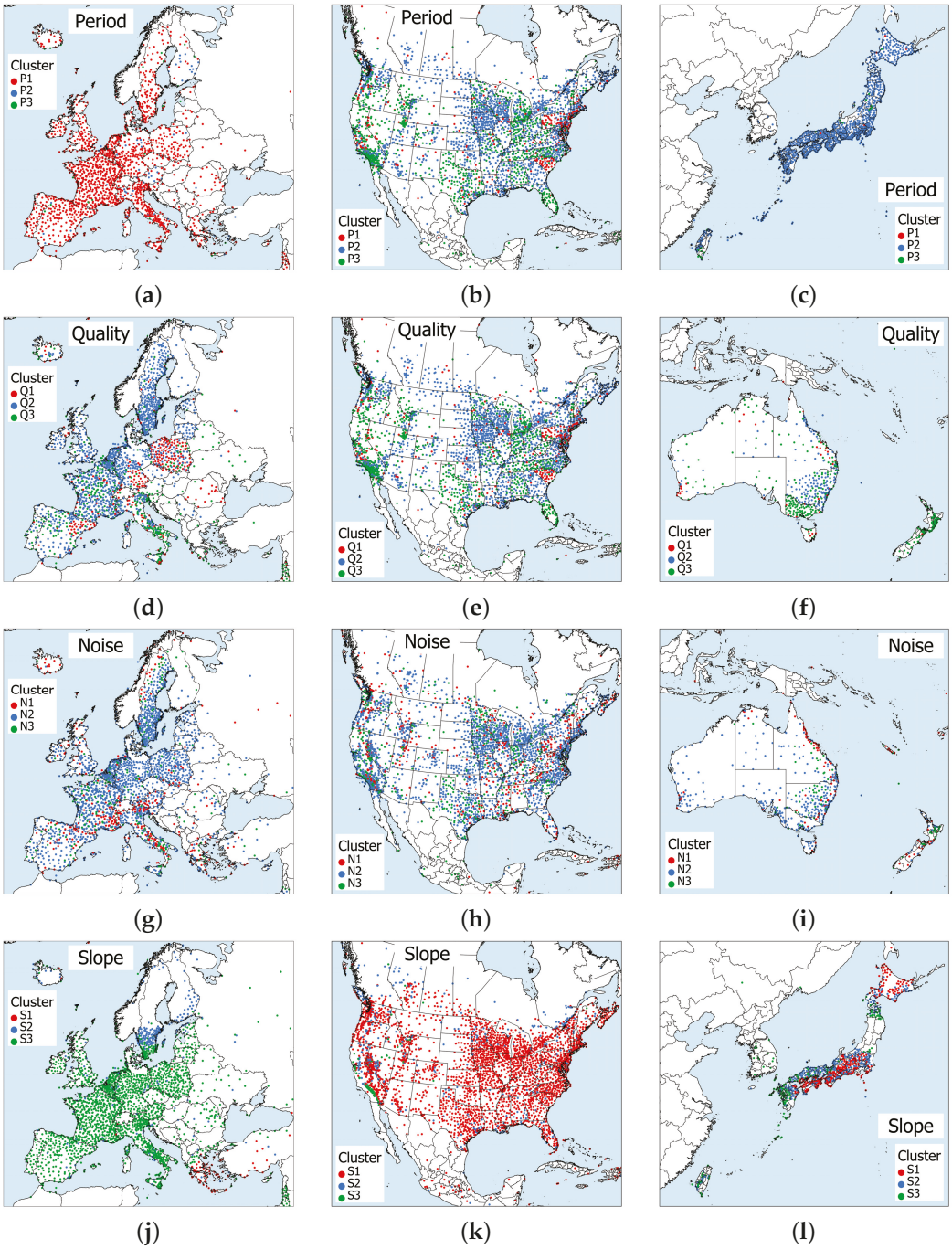


Figure 9. Regional distribution of GNSS stations classified by cluster analysis for four signal components: periodicity (P), quality (Q), noise (N), and slope (S). Each row corresponds to a component: (**a**–**c**) periodicity, (**d**–**f**) quality, (**g**–**i**) noise, and (**j**–**l**) slope. Columns represent regions: Europe, USA, and other areas (for periodicity, quality, and noise, 'Other' includes remaining global regions; for slope, 'Other' corresponds to Japan in panel (**l**) and Australia in panels (**f**,**i**)). Stations are classified into three clusters per component. This figure highlights how GNSS signal characteristics and data quality are spatially structured at regional and continental scales. Colors are consistent across all components: the same color always represents the same cluster number (Cluster 1, Cluster 2, Cluster 3), while the P/Q/S/N labels indicate the component-specific cluster assignment.

5.1. Periodicity

Regarding the periodic component, the analysis of the characteristics of groups P1 and P2 reveals a marked regional variability in the expression of seasonal signals, particularly in

https://doi.org/10.3390/math14050855

the horizontal components. In group P1, which encompasses much of Europe, the annual periodicity appears to be dominant in the East component. In contrast, in group P2, covering large areas of North and South America, the periodicity is predominantly expressed in the North component. Several studies have shown that in the Americas—particularly at IGS stations located in the United States and South America—the North component exhibits larger seasonal amplitudes and greater discrepancies with loading models, suggesting a dominance of this component in the annual periodicity. This trend has been documented in global analyses demonstrating a higher sensitivity of the North component to hydrological and atmospheric variations across the American continent, as well as a greater complexity in its modeling [42,43].

In contrast, in Europe, the specialized literature indicates that the East component tends to exhibit a more stable and coherent seasonal behavior across stations, with well-defined annual amplitudes and lower dispersion relative to loading models. Studies focusing on European GNSS networks have identified that the seasonal signal in the East component is particularly robust, suggesting a more direct control by regional geophysical processes such as continental hydrological loading and thermoelastic crustal variations [44,45]. Taken together, these results support the hypothesis that the dominant periodicity differs between regions, with a clear predominance of the North component in the Americas and the East component in Europe.

As a specific case within group P2, where semiannual periodicity is more prominent in the horizontal components, several global studies have shown that GNSS stations located in the American continent exhibit greater complexity in their horizontal seasonal signals, often characterized by a more pronounced semiannual component. This behavior has typically been linked to the combined influence of atmospheric, hydrological, and oceanic loading, whose interannual variability is particularly strong in the Americas. Recent analyses indicate that semiannual amplitudes in the North and East components are systematically larger at American stations compared with those in Europe, suggesting a more dynamic crustal response in this region, modulated by geophysical processes with higher temporal variability [23,42].

Regarding group P3, which includes coastal areas of the Pacific Ocean (Japan, Korea and Taiwan, Australia, New Zealand, and the Polynesian islands) as well as the southwestern United States, Mexico, and Alaska, the group profile reveals a stronger presence of semiannual periodicity in the vertical component. Studies focused on the southwestern Pacific indicate that the combined influence of oceanic and hydrological loading, together with tectonic processes, generates highly seasonal vertical motion patterns that are not purely annual, with significant subannual contributions at coastal stations on Pacific islands near Australia and New Zealand [46]. This behavior may be related to tidal effects associated with the semiannual solar declination cycle.

Consistently, global analyses of relative sea level and vertical motion have shown that in tectonically active margins—such as the Pacific Rim and high-latitude regions—the vertical component exhibits a nonlinear behavior dominated by loading variations that amplify subannual signals (including the semiannual one), particularly in areas such as Alaska and other sectors of the northern Pacific [47]. Additionally, regional studies integrating GNSS observations with hydrological loading models demonstrate that in areas subjected to strong seasonal climatic forcing, the vertical component robustly incorporates both annual and semiannual terms [48]. This is consistent with the presence of an enhanced semiannual vertical signal in the southwestern United States and Mexico, where combined oceanic, hydrological, and tectonic loading effects converge.

https://doi.org/10.3390/math14050855

As a specific example of group P3 within the European region, the case of southern Finland stands out. The detection of a semiannual signal in the vertical component in this area is consistent with glacial isostatic uplift mechanisms, which make the crust particularly sensitive to seasonal loading variations. The combined effects of snow accumulation and melt, the presence of lakes, changes in hydrological storage, and semiannual variations in atmospheric pressure and Baltic Sea ocean loading generate a bimodal vertical response that can be observed in GNSS time series.

5.2. Slope

The clustering of slope fields is strongly linked to tectonic context and reflects the grouping of stations according to the geodynamic setting in which they are embedded. Groups S1 and S3 exhibit contrasting characteristic profiles, with relatively low and high horizontal slopes, respectively. Group S1 is mainly located in the tectonically stable part of the Americas, as well as in regions of Turkey and Greece and parts of Japan and the Philippines. Group S3, in contrast, is predominant across Europe and Africa.

In regions where both groups converge, tectonically active and seismically dynamic areas emerge as a result of the relative slope differences between stations, as observed in California, Panama, Chile, Greece, and Turkey, as well as in volcanic regions such as Japan and the Philippines. The relationship between relative slope gradients and seismicity and volcanism has been extensively documented. In California, for example, the lateral interaction between the Pacific and North American plates generates strong slope gradients and highly seismic transform faults [49,50]. In subduction zones such as Chile, Japan, and the Philippines, rapid and heterogeneous plate convergence leads to deformation accumulation, seismic rupture, and magma generation through fluid-induced melting [51,52]. In the eastern Mediterranean, relative slope differences between Africa, Arabia, and Eurasia account for the intense distributed deformation, seismic activity, and volcanism associated with the Hellenic Arc [53].

In the case of group S2, characterized by a high relative slope in the vertical component (whether uplift or subsidence), its presence is predominant in regions undergoing post-glacial isostatic adjustment, such as northern Europe (Norway, Sweden, and Finland) and areas of Canada and Greenland where uplift is occurring. This group is also found in volcanic regions such as the Azores, Taiwan, and Japan. In the specific context of isostasy, high-precision geodetic measurements in these regions confirm this behavior [54–56]. In contrast, vertical variations—both positive and negative—in volcanic areas may be related to magmatic adjustments, intrusions, and deformation associated with subduction arcs, hydrological loading changes, or oceanic rift processes, as documented in studies from the Azores, Japan, and Taiwan [57].

To provide a preliminary quantitative validation of the clustering results, we compared the observed GNSS velocities with the plate-motion predictions of the GSRMv1 model [58]. The stations included in the analysis yielded correlation coefficients of 0.446 for the East component and 0.783 for the North component. These values indicate a systematically better agreement in the meridional direction, consistent with previous studies reporting larger residuals and higher noise levels in the East component of GNSS velocities. In addition, the spatial distribution of the residual vectors reveals differences of up to 40–50 mm/yr in major plate-convergence zones, such as western North America, the Hellenic–Anatolian region, and the eastern Eurasian margin, in line with the expected behavior in areas of strong tectonic interaction.

It is worth noting that the boundaries between the velocity-based clusters tend to coincide with regions of enhanced seismic, tectonic, or deformational activity. These

 https://doi.org/10.3390/math14050855

transition zones separate areas with distinct kinematic behavior and may therefore reflect local stress accumulation or deformation gradients. This pattern provides additional geophysical support for the clustering structure identified in this study.

5.3. Quality

For the quality component, although all stations generally exhibit good performance due to prior filtering and preprocessing, the algorithm identified three distinct clusters. Groups Q1 and Q2 show similar characteristics in terms of the percentage of available data in the time series. Q2 presents slightly lower relative quality compared with Q1, which is the group with the lowest average number of offsets and therefore the highest overall quality. Group Q3, in turn, also shows good quality, with a higher percentage of available data but without standing out in terms of offset occurrence. All groups have a comparable average time span, as the last five years from 2020 were selected for all stations.

The spatial distribution of the clusters is heterogeneous but shows a clear geographical pattern linked to the network to which each station belongs. The optimal-quality group Q1 is strongly represented in stations located in South Carolina and Pennsylvania (United States), Poland, Bavaria (Germany), Catalonia and Aragon (Spain), and Romania. For the good-quality group Q3, a higher concentration is observed in New Zealand, Japan, and Korea in Asia; in stations across the U.S. states of Texas, Alabama, Florida, California, and Alaska; in Chile; in New South Wales (Australia); in southern Italy; and in Israel. Several studies have shown that well-maintained continuous GNSS networks exhibit a very low incidence of jumps, discontinuities, and data gaps, resulting in more stable and reliable time series [25,59].

5.4. Noise

Regarding the noise component, most stations exhibit generally low noise levels, and the algorithm has classified them into group N2. Stations in group N1, by contrast, show higher values of random-walk noise and white noise. This group is predominantly distributed across equatorial and tropical regions, as well as northern Australia. It is also present in mountainous areas such as the Appalachian range in the United States and several European mountain chains, including the Alps, the Apennines, and the Scandinavian Mountains. Additionally, this group extends over large areas of Greenland and appears in seismically and volcanically active regions such as parts of Japan, the Hawaiian Islands, and New Zealand.

These patterns have been documented in equatorial and tropical regions, where atmospheric variability and hydrological loading increase the stochastic component of the time series [12]. Likewise, global studies have shown that stations located in mountainous areas exhibit higher noise levels due to local geomorphological and meteorological effects [60]. This behavior is also observed in high-latitude regions such as Greenland, where extreme atmospheric conditions intensify random-walk noise [61]. In addition, in seismically and tectonically active areas such as Japan and New Zealand, geodynamic activity has been shown to increase the stochastic variability of GNSS time series, leading to elevated noise levels [25].

Stations in group N3 exhibit higher levels of flicker noise. Their spatial distribution is more diffuse and complex, although notable concentrations appear in northern Sweden and in volcanic regions such as New Zealand and Japan. In the United States, this group is present in areas of western Texas, northern Minnesota, the region near Vancouver, and the Mammoth Lakes area in California. A significant presence is also observed in the states of Alaska and Indiana, as well as in central Mexico.

https://doi.org/10.3390/math14050855

Correlated noise in GNSS time series exhibits a marked geographical dependence, with stronger intensities in high-latitude regions such as northern Scandinavia and Alaska [9], where ionospheric and atmospheric conditions favor the emergence of flicker-type components. This effect has also been documented in studies focused on the polar ionosphere, where high-frequency variability increases correlated noise in GNSS observations [62]. Moreover, the greater presence of flicker noise in volcanic regions such as Japan, New Zealand, or Mammoth Lakes is consistent with studies showing that magmatic activity and associated deformation generate significant increases in non-white noise, including flicker noise [63]. Finally, the occurrence of this group in regions with active aquifers or strong hydrological loading, such as central Mexico or parts of the U.S. Midwest, aligns with the findings of [12], who demonstrated that hydrogeological dynamics introduce correlated-noise components into GNSS time series.

6. Conclusions

In this study, a comprehensive multivariate statistical framework was applied to the analysis of GNSS time series from 21548 permanent stations registered in the University of Nevada database. By combining harmonic regression, noise characterization, quality assessment, and slope estimation with clustering and discriminant analysis techniques, we demonstrate that large GNSS networks can be systematically classified into homogeneous groups that reflect both the dominant signal characteristics of the time series and their underlying geophysical context.

Regarding the first research question (RQ1), the results clearly show that GNSS stations can be robustly classified into well-defined and interpretable clusters using variables related to periodicity, noise, quality, and slope. The identified cluster families (P1–P3, Q1–Q3, N1–N3, and S1–S3) capture the main structural features of the time series and summarize the dominant behavior of the stations in a compact and meaningful way. The consistency of the cluster profiles across thousands of stations supports the suitability of multivariate statistical methods for large-scale GNSS network characterization.

In relation to the second research question (RQ2), a strong correlation is observed between the spatial distribution of the clusters and the geographical and geodynamic setting of the stations. Distinct regional patterns emerge in all analyzed components. Periodicity clusters reveal marked continental-scale differences in the dominance of seasonal signals, with contrasting behavior between Europe, North and South America, and the Pacific region. Slope clusters closely reflect tectonic regimes, distinguishing stable continental interiors, actively deforming plate boundaries, volcanic arcs, and regions affected by post-glacial isostatic adjustment. Similarly, noise clusters exhibit clear associations with climatic zones, topography, latitude, and tectonic activity, while quality clusters reflect differences in network management, monument stability, and operational standards. These results confirm that the statistical classification of GNSS time series is not arbitrary, but strongly controlled by physical and environmental factors.

Concerning the third research question (RQ3), the cluster profiles derived in this work provide a robust statistical framework for enhancing deformation analysis and geohazard assessment. By grouping stations with similar signal characteristics, the proposed approach facilitates the identification of reliable reference stations, the detection of anomalous behavior, and the interpretation of spatially coherent deformation patterns. This is particularly relevant for applications in seismic and volcanic risk analysis, where distinguishing between tectonic signals and noise-related artifacts is essential.

Considering the limitations of the proposed framework, the simplified representation of GNSS time-series behavior using a reduced set of features that, while capturing domi-

https://doi.org/10.3390/math14050855

nant statistical characteristics, cannot fully reflect the physical complexity of site-specific processes such as monument–environment interactions, nonlinear deformation, or transient geophysical events. Likewise, the noise characterization relies on empirical metrics derived from regression residuals rather than on a formal stochastic model selection, meaning that the estimated noise components should be viewed as descriptive indicators of temporal variability rather than strict identifications of ideal stochastic processes.

These limitations include the heterogeneity of data across the global network, which introduces additional constraints. Differences in time-series length, metadata completeness, monument stability, and maintenance practices may affect the extracted features and, consequently, the clustering results, even after applying normalization and filtering procedures. Furthermore, the unsupervised clustering approach is inherently sensitive to the choice of feature space and distance metric. Alternative feature definitions, weighting schemes, or clustering algorithms could lead to different groupings. For this reason, the resulting clusters should be interpreted as statistically meaningful patterns within the proposed analytical framework, rather than as definitive categories of GNSS station behavior.

Overall, this study demonstrates that multivariate statistical classification constitutes a powerful and scalable tool for extracting geophysical insight from large GNSS networks. The proposed methodology is transferable to different regions and temporal windows and can be readily extended by incorporating alternative stochastic models or additional geodetic observables. These results highlight the value of statistical clustering and discriminant analysis as complementary approaches for improving the interpretation, reliability, and practical use of GNSS time series in geodynamic studies.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/math14050855/s1, Figure S1: GNSS Stations Removed During the Filtering Steps for Each Dataset (S, N, P, Q). Figure S2: Distributions of quality features by cluster. Figure S3: Distributions of velocity features by cluster. Figure S4: Distributions of noise features by cluster. Figure S5: Distributions of periodicity features by cluster.

Author Contributions: Conceptualization, A.S.-A., A.P.-F. and D.Á.-R.; methodology, A.S.-A. and D.Á.-R.; software, D.Á.-R.; validation, A.S.-A. and D.Á.-R.; formal analysis, A.S.-A. and D.Á.-R.; investigation, A.S.-A., A.P.-F. and D.Á.-R.; resources, A.S.-A. and D.Á.-R.; data curation, D.Á.-R.; writing—original draft preparation, A.S.-A., A.P.-F. and D.Á.-R.; writing—review and editing, A.S.-A., A.P.-F. and D.Á.-R.; visualization, A.S.-A., A.P.-F. and D.Á.-R.; supervision, A.S.-A. and A.P.-F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data and code that support the findings of this study are available from the following link: https://github.com/DanielAlvarezRuiz/Mathematics_Alvarez_et_al_2026 (accessed on 30 December 2024).

Acknowledgments: The authors would like to thank their families for the support provided throughout the development of this research article, with special gratitude to Paula Álvarez Ruiz and to Gonzalo, Felipe, and Carlota Alés Sánchez.

Conflicts of Interest: The authors declare no competing interests.

References

1. Ge, M.; Gendt, G.; Dick, G.; Zhang, F.P.; Rothacher, M. A new data processing strategy for huge GNSS global networks. *J. Geod.* **2006**, *80*, 199–203. [CrossRef]
2. Wang, S.; Huang, S.; Fang, H. Ground GNSS station selection to generate the global ionosphere maps using the information content. *Space Weather* **2022**, *20*, e2020SW002675. [CrossRef]

3. Cao, Y.; Kuznetsov, E.; Miao, C.; Chen, S.; Meng, T. The current situation of Russia's GNSS continuous operating reference station network and thinking on future development. *Adv. Space Res.* **2024**, *73*, 3896–3908. [CrossRef]
4. Altamimi, Z.; Collilieux, X.; Métivier, L. Reference frames for applications in geosciences. In *International Association of Geodesy Symposia*; Springer Nature: Berlin/Heidelberg, Germany, 2013; Volume 138, pp. 137–145. [CrossRef]
5. Ohno, K.; Ohta, Y.; Takamatsu, N.; Munekane, H.; Iguchi, M. Real-time modeling of transient crustal deformation through the quantification of uncertainty deduced from GNSS data. *Earth Planets Space* **2024**, *76*, 140. [CrossRef]
6. Blewitt, G.; Hammond, W.C.; Kreemer, C. Harnessing the GPS data explosion for interdisciplinary science. *Eos* **2018**, *99*, e2020943118. [CrossRef]
7. Williams, S.D.P.; Penna, N.T. Non-tidal ocean loading effects on geodetic GPS heights. *Geophys. Res. Lett.* **2011**, *38*, L09314. [CrossRef]
8. Langbein, J. Noise in GPS displacement measurements from southern California and southern Nevada. *J. Geophys. Res.* **2008**, *113*, B05405. [CrossRef]
9. Gobron, K.; Rebischung, P.; Chanard, K.; Altamimi, Z. Anatomy of the spatiotemporally correlated noise in GNSS station position time series. *J. Geod.* **2024**, *98*, 34. [CrossRef]
10. Ren, Y.; Lian, L.; Wang, J. Analysis of seismic deformation from global three-decade GNSS displacements: Implications for a three-dimensional Earth GNSS velocity field. *Remote Sens.* **2021**, *13*, 3369. [CrossRef]
11. He, X.; Bos, M.S.; Montillet, J.P.; Fernandes, R.M.S. Investigation of the noise properties at low frequencies in long GNSS time series. *J. Geod.* **2019**, *93*, 1271–1282. [CrossRef]
12. He, Y.; Nie, G.; Wu, S.; Li, H. Analysis and discussion on the optimal noise model of global GNSS long-term coordinate series considering hydrological loading. *Remote Sens.* **2021**, *13*, 431. [CrossRef]
13. Dmitrieva, K.; Segall, P.; Bradley, A.M. Effects of linear trends on estimation of noise in GNSS position time-series. *Geophys. J. Int.* **2016**, *208*, 281–288. [CrossRef]
14. Kaczmarek, A.; Kontny, B. Identification of the noise model in the time series of GNSS station coordinates using wavelet analysis. *Remote Sens.* **2018**, *10*, 1611. [CrossRef]
15. Montillet, J.P.; Bos, M.S.; Fernandes, R.M.S.; Williams, S.D.P.; Yu, K. *Geodetic Time Series Analysis in Earth Sciences*; Springer Geophysics; Springer: Berlin/Heidelberg, Germany, 2020. [CrossRef]
16. Li, M.; Huang, G.; Wang, L.; Xie, W. Comprehensive classification assessment of GNSS observation data quality by fusing k-means and KNN algorithms. *GPS Solut.* **2024**, *28*, 21. [CrossRef]
17. Liu, Y.; Nagahata, H.; Uchiyama, H.; Taniguchi, M. Discriminant and cluster analysis of possibly high-dimensional time series data by a class of disparities. *Commun. Stat.—Simul. Comput.* **2017**, *46*, 7875–7892. [CrossRef]
18. Alonso, A.M.; Berrendero, J.R.; Hernández, A.; Justel, A. Time series clustering based on forecast densities. *Comput. Stat. Data Anal.* **2006**, *51*, 762–776. [CrossRef]
19. Barba, P.; Rosado, B.; Ramírez-Zelaya, J.; Berrocoso, M. Comparative analysis of statistical and analytical techniques for the study of GNSS geodetic time series. *Eng. Proc.* **2021**, *5*, 21. [CrossRef]
20. Bogusz, J.; Klos, A. On the significance of periodic signals in noise analysis of GPS station coordinates time series. *GPS Solut.* **2015**, *19*, 403–417. [CrossRef]
21. Rosado, B.; Plaza, S.; Páez, R.; Gárate, J.; Fernández-Palacín, F.; Berrocoso, M. Establishing models of surface deformation from geodetic time series GNSS in the southern region of the Iberian Peninsula and North Africa (Spain). In Proceedings of the 31st IUGG Conference on Mathematical Geophysics, Paris, France, 6–10 June 2016; p. 203.
22. Costantino, G.; Giffard-Roisin, S.; Dalla Mura, M.; Socquet, A. Denoising of geodetic time series using spatiotemporal graph neural networks: Application to slow slip event extraction. *arXiv* **2024**, arXiv:2405.03320. [CrossRef]
23. Klos, A.; Bogusz, J.; Bos, M.S.; Gruszczynska, M. Modelling the GNSS time series: Different approaches to extract seasonal signals. In *Geodetic Time Series Analysis in Earth Sciences*; Montillet, J.P., Bos, M.S., Eds.; Springer: Cham, Switzerland, 2018; pp. 211–234. [CrossRef]
24. Shumway, R.H.; Stoffer, D.S. *Time Series Analysis and Its Applications: With R Examples*, 3rd ed.; Springer: Berlin/Heidelberg, Germany, 2011.
25. Zhang, J.; Lian, L.; Huang, C.; Xu, C.; Zhang, S. Automated offset detection approaches: Case study in IGS Repro2 and Repro3. *GPS Solut.* **2024**, *28*, 123. [CrossRef]
26. Brockwell, P.J.; Davis, R.A. *Time Series: Theory and Methods*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 1991.
27. Percival, D.B.; Walden, A.T. *Spectral Analysis for Physical Applications: Multitaper and Conventional Univariate Techniques*; Cambridge University Press: Cambridge, UK, 1993. [CrossRef]
28. Tao, Y.; Liu, C.; Liu, C.; Zhao, X.; Hu, H. Empirical Wavelet Transform Method for GNSS Coordinate Series Denoising. *J. Geovisualization Spat. Anal.* **2021**, *5*, 9. [CrossRef]

 https://doi.org/10.3390/math14050855

29. Santamaría-Gómez, A.; Mémin, A. Geodetic secular velocity estimation using multiyear GNSS data and the MIDAS robust trend estimator. *J. Geophys. Res. Solid Earth* **2015**, *120*, 7343–7361. [CrossRef]
30. Blewitt, G.; Lavallée, D. Effect of Annual Signals on Geodetic Velocity. *J. Geophys. Res. Solid Earth* **2002**, *107*, ETG 9-1–ETG 9-11. [CrossRef]
31. Santamaría-Gómez, A.; Bouin, M.N.; Collilieux, X.; Wöppelmann, G. Correlated errors in GPS position time series: Implications for velocity estimates. *J. Geophys. Res. Solid Earth* **2011**, *116*, B01405. [CrossRef]
32. Dawidowicz, K.; Krzan, G.; Wielgosz, P. Offsets in the EPN station position time series resulting from antenna/radome changes: PCC type-dependent model analyses. *GPS Solut.* **2023**, *27*, 9. [CrossRef]
33. Khazraei, S.M.; Amiri-Simkooei, A.R. Improving offset detection algorithm of GNSS position time-series using spline function theory. *Geophys. J. Int.* **2021**, *224*, 257–270. [CrossRef]
34. Sun, X.; Lu, T.; Hu, S.; Huang, J.; He, X.; Montillet, J.P.; Ma, X.; Huang, Z. The relationship of time span and missing data on the noise model estimation of GNSS time series. *Remote Sens.* **2023**, *15*, 3572. [CrossRef]
35. Qiu, Y.; Fang, C.; Song, S.; Huang, X.; Wang, C.; Wang, J. TsQuality: Measuring Time Series Data Quality in Apache IoTDB. *Proc. VLDB Endow.* **2023**, *16*, 3982–3985. [CrossRef]
36. Santamaría-Gómez, A. SARI: interactive GNSS position time series analysis software. *GPS Solut.* **2019**, *23*, 52. [CrossRef]
37. Bezdek, J.C. *Pattern Recognition with Fuzzy Objective Function Algorithms*; Advanced Applications in Pattern Recognition; Springer: New York, NY, USA, 1981. [CrossRef]
38. Chicco, D.; Campagner, A.; Spagnolo, A.; Ciucci, D.; Jurman, G. The Silhouette coefficient and the Davies-Bouldin index are more informative than Dunn index, Calinski-Harabasz index, Shannon entropy, and Gap statistic for unsupervised clustering internal evaluation of two convex clusters. *PeerJ Comput. Sci.* **2025**, *11*, e3309. [CrossRef]
39. Salman, Z.; Alomary, A. Performance of the K-means and fuzzy C-means algorithms in big data analytics. *Int. J. Inf. Technol.* **2024**, *16*, 465–470. [CrossRef]
40. Pickens, A.; Sengupta, S. Benchmarking studies aimed at clustering and classification tasks using K-means, fuzzy C-means and evolutionary neural networks. *Mach. Learn. Knowl. Extr.* **2021**, *3*, 695–719. [CrossRef]
41. Cebeci, Z.; Yildiz, F. Comparison of K-means and Fuzzy C-means algorithms on different cluster structures. *J. Agric. Inform.* **2015**, *6*, 13–23. [CrossRef]
42. Niu, Y.; Wei, N.; Li, M.; Rebischung, P.; Shi, C.; Chen, G. Quantifying discrepancies in the three-dimensional seasonal variations between IGS station positions and load models. *J. Geod.* **2022**, *96*, 31. [CrossRef]
43. Li, Z.; Yang, K.; Jiang, W.; Deng, L.; Zou, Y. Impacts of period offset and period variation on modeling seasonal signals in GNSS coordinate time series. *Geo-Spat. Inf. Sci.* **2025**, 1–16. [CrossRef]
44. Kaczmarek, A.; Kontny, B. Estimates of seasonal signals in GNSS time series and environmental loading models with iterative least-squares estimation (iLSE). *Acta Geodyn. Geomater.* **2018**, *15*, 131–141. [CrossRef]
45. Nistor, S.; Suba, N.S.; El-Mowafy, A.; Apollo, M.; Malkin, Z.; Nastase, E.I.; Kudrys, J.; Maciuk, K. Implication between geophysical events and the variation of seasonal signal determined in GNSS position time series. *Remote Sens.* **2021**, *13*, 3478. [CrossRef]
46. Ballu, V.; Gravelle, M.; Wöppelmann, G.; de Viron, O.; Rebischung, P.; Becker, M.; Sakic, P. Vertical land motion in the Southwest and Central Pacific from available GNSS solutions and implications for relative sea levels. *Geophys. J. Int.* **2019**, *218*, 1537–1551. [CrossRef]
47. Oelsmann, J.; Marcos, M.; Passaro, M.; Sanchez, L.; Dettmering, D.; Dangendorf, S.; Seitz, F. Regional variations in relative sea-level changes influenced by nonlinear vertical land motion. *Nat. Geosci.* **2024**, *17*, 137–144. [CrossRef] [PubMed]
48. Xu, P.; Jiang, T.; Li, W.; Xu, G.; Zhang, C.; Wang, W.; Tian, K.; Feng, J. Analyzing the seasonal vertical displacement fluctuations using the Global Navigation Satellite System and hydrological load: A case study of the Western Yunnan Region. *Water* **2024**, *16*, 1260. [CrossRef]
49. Argus, D.F.; Gordon, R.G. Present tectonic motion across the Coast Ranges and San Andreas fault system in central California. *Geol. Soc. Am. Bull.* **2001**, *113*, 1580–1592. [CrossRef]
50. McCaffrey, R. Block kinematics of the Pacific–North America plate boundary in the southwestern United States from inversion of GPS, seismological, and geologic data. *J. Geophys. Res.* **2005**, *110*, B07401. [CrossRef]
51. Moreno, M.; Melnick, D.; Rosenau, M.; Bolte, J.; Klotz, J.; Echtler, H.; Baez, J.; Bataille, K.; Chen, J.; Bevis, M.; et al. Heterogeneous plate locking in the South–Central Chile subduction zone: Building up the next great earthquake. *Earth Planet. Sci. Lett.* **2011**, *305*, 413–424. [CrossRef]
52. Loveless, J.P.; Meade, B.J. Geodetic imaging of plate motions, slip rates, and partitioning of deformation in Japan. *Earth Planet. Sci. Lett.* **2010**, *297*, 34–41. [CrossRef]

https://doi.org/10.3390/math14050855

53. Reilinger, R.; McClusky, S.; Vernant, P.; Lawrence, S.; Ergintav, S.; Cakmak, R.; Ozener, H.; Kadirov, F.; Guliev, I.; Stepanyan, R.; et al. GPS constraints on continental deformation in the Africa–Arabia–Eurasia collision zone. *J. Geophys. Res.* **2006**, *111*, B05411. [CrossRef]
54. Milne, G.A.; Davis, J.L.; Mitrovica, J.X.; Scherneck, H.G.; Johansson, J.M.; Vermeer, M.; Koivula, H. Space–geodetic constraints on glacial isostatic adjustment in Fennoscandia. *Science* **2001**, *291*, 2381–2385. [CrossRef] [PubMed]
55. Sella, G.F.; Stein, S.; Dixon, T.H.; Craymer, M.; James, T.S.; Mazzotti, S.; Dokka, R.K. Observations of glacial isostatic adjustment in North America from GPS. *Geophys. Res. Lett.* **2007**, *34*, L02306. [CrossRef]
56. Khan, S.A.; Sasgen, I.; Bevis, M.; van Dam, T.; Bamber, J.L.; Wahr, J.; Willis, M.; Kjær, K.H.; Wouters, B.; Helm, V.; et al. Geodetic measurements reveal differences between Greenland ice sheet flow and GIA uplift. *Geophys. Res. Lett.* **2016**, *43*, 7815–7823. [CrossRef]
57. Ching, K.-E.; Rau, R.-J.; Johnson, K.M.; Lee, J.-C.; Hu, J.-C. Present-day kinematics of active mountain building in Taiwan from GPS observations during 1995–2005. *J. Geophys. Res.* **2011**, *116*, B09405. [CrossRef]
58. Kreemer, C.; Blewitt, G.; Klein, E.C. A geodetic plate motion and global strain rate model. *Geochem. Geophys. Geosyst.* **2014**, *15*, 3849–3889. [CrossRef]
59. Lau, L.; Tai, K.W. A data quality assessment approach for high-precision GNSS Continuously Operating Reference Stations (CORS) with case studies in Hong Kong and Canada/USA. *Remote Sens.* **2023**, *15*, 1925. [CrossRef]
60. Wang, L.; Wu, Q.; Wu, F.; He, X. Noise content assessment in GNSS coordinate time-series with autoregressive and heteroscedastic random errors. *Geophys. J. Int.* **2022**, *231*, 856–876. [CrossRef]
61. Wu, Z.; Lu, C.; Tan, Y.; Zheng, Y.; Liu, Y.; Jin, K. Real-time GNSS tropospheric delay estimation with a novel global random walk processing noise model (GRM). *J. Geod.* **2023**, *97*, 112. [CrossRef]
62. Guo, K.; Veettil, S.V.; Weaver, B.J.; Aquino, M. Mitigating high latitude ionospheric scintillation effects on GNSS Precise Point Positioning exploiting 1-s scintillation indices. *J. Geod.* **2021**, *95*, 30. [CrossRef]
63. Carbonari, R.; Riccardi, U.; De Martino, P.; Cecere, G.; Di Maio, R. Wavelet-like denoising of GNSS data through machine learning: Application to the time series of the Campi Flegrei volcanic area (Southern Italy). *Geomat. Nat. Hazards Risk* **2023**, *14*, 2187271. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/math14050855

MDPI AG
Grosspeteranlage 5
4052 Basel
Switzerland
Tel.: +41 61 683 77 34

Mathematics Editorial Office
E-mail: mathematics@mdpi.com
www.mdpi.com/journal/mathematics



Disclaimer/Publisher's Note: The title and front matter of this reprint are at the discretion of the Guest Editor. The publisher is not responsible for their content or any associated concerns. The statements, opinions and data contained in all individual articles are solely those of the individual Editor and contributors and not of MDPI. MDPI disclaims responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.





Academic Open
Access Publishing

mdpi.com

ISBN 978-3-7258-9396-6