



Article

Multi-Stage Evaluation Framework for Identifying Deployment-Ready Prediabetes Prediction Models

Michael Sher  and Milan Toma * 

Algorithmic Medicine Laboratory, Department of Osteopathic Manipulative Medicine, College of Osteopathic Medicine, New York Institute of Technology, Old Westbury, NY 11568, USA; msher02@nyit.edu

* Correspondence: mtoma@nyit.edu

Abstract

Selecting machine learning algorithms for clinical deployment demands comprehensive evaluation beyond conventional performance metrics. While automated frameworks simplify model generation, identifying algorithms suitable for real-world medical applications requires systematic assessment of learning dynamics, generalization stability, and cross-subset reliability. This study addresses prediabetes prediction through a multi-stage evaluation comparing automated machine learning frameworks, neural networks, gradient boosting implementations (XGBoost, CatBoost, LightGBM), and specialized imbalance-handling techniques. A questionnaire-based dataset with a substantial class imbalance was analyzed through progressive evaluation stages: aggregate performance metrics, learning curve analysis, minority class detection capability, and cross-subset generalization stability. Linear Discriminant Analysis achieved maximum validation metrics in automated screening but exhibited flat learning curves, indicating an exhausted learning capacity. XGBoost demonstrated optimal convergence dynamics with the highest validation performance (0.749 AUC), yet suffered substantial validation-to-test degradation (5.9 percentage points). CatBoost, despite inferior validation performance (0.696 accuracy), exhibited exceptional cross-subset stability with minimal performance decline (0.2 percentage points) while achieving a comparable test accuracy (0.694). CatBoost was selected for deployment based on its superior generalization stability, demonstrating that a multi-dimensional evaluation spanning aggregate metrics, learning dynamics, and cross-subset stability is essential for identifying clinically deployable models, as validation performance alone provides insufficient evidence for real-world applicability.

Keywords: machine learning model selection; prediabetes prediction; learning curve analysis; generalization stability; gradient-boosting decision trees; clinical deployment; class imbalance; healthcare machine learning



Academic Editor: Andreas Holzinger

Received: 5 June 2026

Revised: 30 June 2026

Accepted: 1 July 2026

Published: 3 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The deployment of machine learning algorithms in clinical settings demands evaluation far beyond conventional performance metrics. While automated machine learning frameworks have simplified model generation, selecting algorithms for real-world medical applications requires comprehensive assessment of learning dynamics, generalization stability, and cross-population reliability. Models selected solely on cross-validated accuracy often fail when confronting distribution shifts across diverse clinical populations and settings.

Systematic analysis of training and validation performance trajectories reveals critical model behaviors invisible in aggregate metrics. Learning curves expose memorization patterns, convergence failures, and training-validation gaps that compromise real-world applicability. Models exhibiting flat trajectories demonstrate exhausted capacity; those with continuously diverging curves signal overfitting. Only algorithms with appropriate convergence patterns (where training and validation performance stabilize with minimal gaps) provide confidence for reliable clinical deployment. Healthcare's stringent requirements for consistent performance across heterogeneous populations distinguish medical prediction from domains where gradual degradation may be acceptable.

The problem of prediabetes prediction exemplifies the challenges inherent in developing robust clinical decision support systems. Prediabetes represents an intermediate metabolic state characterized by impaired glucose regulation that precedes the development of type 2 diabetes mellitus [1,2]. While various therapeutic interventions have been developed over the past decades, including metformin [3] and glucagon-like peptide-1 receptor agonists [4], early identification remains critical for prevention strategies. Early identification of individuals at elevated risk enables targeted lifestyle interventions and closer monitoring that can prevent or delay progression to overt diabetes with its associated complications [5,6]. Current estimates indicate 12.0% of the US population has diabetes, with 115.2 million adults (≥ 18 years) classified as prediabetic, positioning diabetes as the third most prevalent chronic disease nationally behind hypertension and obesity [7–9]. However, the development of predictive models for prediabetes screening confronts several characteristic challenges of medical prediction tasks, including substantial class imbalance between healthy and at-risk populations, reliance on questionnaire-based behavioral data rather than objective biomarkers, and the imperative to maintain performance across diverse demographic groups and healthcare settings [10–15]. Machine learning approaches have shown promise in diabetes prediction and diagnosis [16–19], complication forecasting [20], and management through smart devices [21]. Recent applications include retinal image analysis for undiagnosed diabetes detection [22], tissue characterization in Type 1 diabetes [23], and medical imaging analysis [24], though limitations remain in clinical deployment [25].

The fundamental question of how to systematically identify machine learning algorithms suitable for clinical deployment is addressed through comprehensive multi-stage evaluation that extends beyond conventional performance metrics. The study compares multiple algorithmic approaches including automated machine learning frameworks, neural network architectures, gradient-boosting decision trees, and specialized imbalance-handling techniques, subjecting each candidate to rigorous assessment spanning aggregate performance metrics, learning curve analysis, minority class detection capability, and cross-subset generalization stability. This methodological progression deliberately mirrors the decision process required for responsible clinical deployment, where initial screening based on validation performance must be supplemented by deeper investigation of learning dynamics and generalization patterns before committing to real-world implementation [26–28].

1.1. Novelty and Contribution

The novelty of this study lies in its evaluation framework rather than in the use of any single algorithm. For structured tabular clinical data, newer or more complex models are not inherently superior; algorithm selection should be based on data characteristics and methodological rigor rather than algorithmic recency [29,30].

This work shows that strong aggregate metrics alone are insufficient for clinical model selection. Learning curves and performance across training, validation, test, and external

subsets help identify memorization, overfitting, instability, and possible leakage that may remain hidden behind high cross-validated accuracy. Recent diabetes prediction studies exemplify this concern, reporting strong classification performance without demonstrating convergence patterns, training-validation gaps, or cross-subset stability [10,31–34]; when training health is not established, such aggregate metrics should be interpreted cautiously as potentially inflated [35,36]. Unlike standard model-selection pipelines that terminate after cross-validation or hold-out testing, this framework explicitly requires demonstration of healthy learning dynamics through training curve analysis and quantification of validation-to-test stability before recommending deployment. While individual components such as learning curve inspection and cross-validation are established techniques, their systematic integration into a sequential elimination process specifically designed for clinical deployment decisions represents the methodological contribution of this work. Accordingly, this study provides a replicable multi-stage framework combining conventional metrics, learning dynamics, minority-class performance, and cross-subset generalization stability to distinguish apparently high-performing models from those that are genuinely suitable for deployment.

1.2. Study Objectives

The optimal machine learning algorithm for prediabetes prediction is identified through multi-stage evaluation encompassing: (1) aggregate performance metrics, (2) learning dynamics analysis, (3) minority class detection, and (4) cross-subset generalization stability. Automated machine learning frameworks, neural networks, gradient boosting implementations, and imbalance-handling techniques are systematically compared to determine which combination of predictive capability and generalization stability best supports clinical deployment. This methodology demonstrates why learning curve analysis and cross-subset validation are vital for detecting model limitations (memorization, convergence failures, unstable generalization) that remain invisible in conventional cross-validation metrics, providing a replicable framework for evaluating medical prediction models where deployment reliability supersedes peak validation performance.

2. Methods

The study utilized a questionnaire-based dataset comprising responses from non-diabetic and prediabetic individuals regarding lifestyle patterns, including physical activity, occupational factors, and health behaviors [37]. The dataset exhibits class imbalance, with non-diabetic individuals substantially exceeding prediabetic cases. The dataset comprises 16,137 respondents, with 10,078 (62.5%) classified as non-diabetic and 6059 (37.5%) as prediabetic. The sample is balanced by sex (49.9% male, 50.1% female) with a mean age of 42.1 years (SD = 11.0, range: 19–60). All respondents were employed at the time of survey completion. Respondents provided information on demographic characteristics (sex, age), anthropometric measurements (weight: mean 66.1 kg, SD 13.0; height: mean 166.2 cm, SD 8.8), occupational status, and physical activity levels including walking frequency (mean 3.85 days/week), strength training (mean 0.80 days/week), and work- and leisure-related physical activity. No missing values were present in the analytic sample. Potential sources of bias include self-reported behavioral data, which may be subject to recall bias and social desirability effects, and the restriction to employed individuals, which may limit generalisability to unemployed or economically inactive populations. To address this imbalance, multiple training strategies incorporating algorithmic modifications and data resampling techniques were employed. The methodological approach was structured into multiple phases, each designed to evaluate different aspects of machine learning

performance on the prediabetes prediction task. The experimental framework can be visualized as shown in Figure 1.

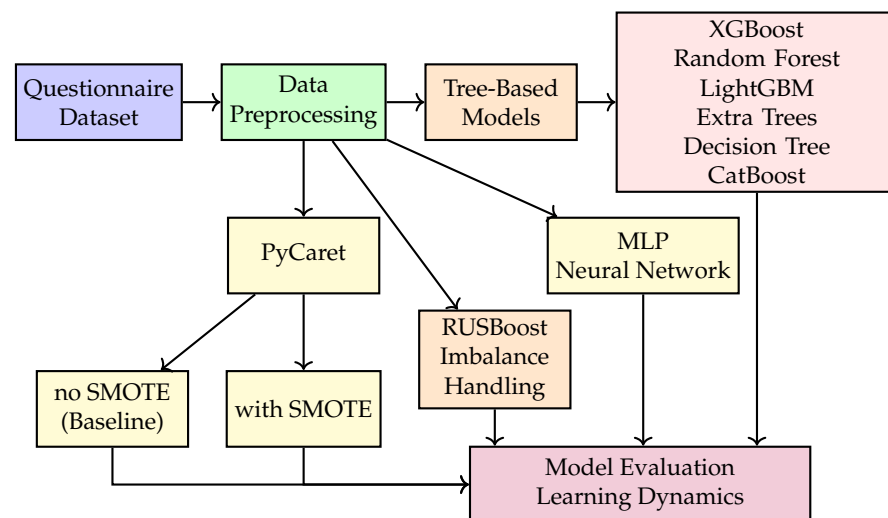


Figure 1. Machine learning workflow diagram illustrating the multiple training approaches employed in this study. The dataset undergoes preprocessing before being applied to five distinct methodological approaches, all of which are ultimately evaluated through learning curve analysis.

2.1. Elementary Principles of Model Selection

The selection of machine learning algorithms must be guided by fundamental principles matching algorithmic architectures to data characteristics, rather than assumptions that newer or more complex methods are universally superior. This principle is particularly critical in medical informatics, where inappropriate algorithm selection can compromise both predictive performance and clinical interpretability.

2.1.1. Data Modality Determines Algorithmic Suitability

Machine learning data broadly categorizes into two distinct modalities: ‘unstructured data’ (images, text, audio, video) and ‘structured tabular data’ (spreadsheets, databases, questionnaire responses). These modalities possess fundamentally different characteristics that determine which algorithmic families are most appropriate [29,30].

Deep learning architectures (including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformers) achieve remarkable success on unstructured data because they excel at automatically learning hierarchical feature representations from raw, high-dimensional inputs where spatial or sequential relationships encode meaningful patterns [38]. Images contain pixel-level correlations across spatial neighborhoods; text contains semantic dependencies across token sequences. Deep learning’s inductive biases align naturally with these data structures.

Structured tabular data, conversely, presents fundamentally different characteristics: heterogeneous feature types (continuous, categorical, ordinal), no inherent spatial or sequential ordering among features, complex non-linear interactions that do not follow hierarchical patterns, and relatively modest dimensionality compared to image or text data. The questionnaire-based prediabetes prediction task in this study exemplifies classical tabular data, comprising categorical responses (occupation, lifestyle behaviors) and numeric values (physical activity metrics) without any spatial, temporal, or sequential structure that deep learning architectures are designed to exploit.

2.1.2. Empirical Evidence for Tree-Based Superiority on Tabular Data

Extensive empirical research demonstrates that gradient-boosting decision trees (GBDTs) consistently match or outperform deep learning approaches on tabular datasets. Grinsztajn et al. [29] conducted comprehensive benchmarks across 45 tabular datasets and concluded that tree-based ensembles represent the most reliable choice for medium-sized tabular datasets, particularly those with fewer than 10,000 observations. Shwartz-Ziv and Armon [30] similarly demonstrated that XGBoost and other tree-based models outperform deep neural networks on most tabular benchmarks. Domain-specific evidence reinforces this recommendation; Arslan et al. [39] demonstrated that tree-based boosting approaches achieve strong predictive performance for type 2 diabetes mellitus prediction using metabolomics data, with KTBoost attaining an AUC of 0.965. Similarly, Jose et al. [40] employed PyCaret-based automated machine learning for diabetes-related cardiovascular risk prediction, finding that the light gradient boosting machine classifier exhibited particular effectiveness as measured by the Kappa statistic and Matthews correlation coefficient.

It should be noted that recent advances in deep learning architectures specifically designed for tabular data, such as TabNet and TabTransformer [41–43], have shown competitive performance with tree-based methods in certain settings, particularly on very large datasets or when transfer learning is beneficial [44]. The present recommendation for tree-based methods applies specifically to medium-sized clinical datasets like the one examined here, and practitioners should evaluate deep learning alternatives when dataset size or transfer learning requirements differ substantially [45–47].

Deep neural networks, by contrast, assume smooth, differentiable relationships and benefit from homogeneous, high-dimensional inputs where learned representations provide advantages over hand-crafted features [48]. Table 1 summarizes the alignment between data modalities and algorithmic families based on current empirical evidence.

Table 1. Algorithm suitability by data modality. Recommendations based on empirical benchmarks from Grinsztajn et al. [29] and Shwartz-Ziv and Armon [30].

Data Type	Recommended Algorithms	Deep Learning Advantage
Images	CNNs, Vision Transformers	Strong
Text	Transformers, RNNs	Strong
Audio/Speech	CNNs, RNNs, Transformers	Strong
Structured Tabular	GBDTs (XGBoost, CatBoost, LightGBM)	Weak/None
Time Series (short)	GBDTs, Classical ML	Moderate

2.1.3. The Fallacy of Algorithmic Recency

A common misconception equates algorithmic novelty with universal superiority. This “newer is better” assumption fundamentally misunderstands that machine learning algorithms embody specific inductive biases optimized for particular data characteristics. Gradient boosting methods (XGBoost, 2016; LightGBM, 2017; CatBoost, 2018) are not outdated predecessors to deep learning but rather represent a distinct algorithmic paradigm purpose-built for tabular prediction tasks [49–51].

The appropriate question is not “Which algorithm is most advanced?” but rather “Which algorithm’s inductive biases best match the data characteristics?” For the structured questionnaire data in prediabetes prediction (comprising categorical lifestyle variables and numeric health indicators without spatial, sequential, or hierarchical structure) gradient-boosting decision trees represent the methodologically correct choice based on established machine learning principles, not a limitation to be overcome through adoption of architecturally inappropriate alternatives.

2.2. Automated Machine Learning with PyCaret

Initial model development employed the PyCaret library for automated comparison of multiple machine learning architectures [40]. Two experiments were conducted: (1) baseline analysis on the original imbalanced dataset, and (2) analysis with Synthetic Minority Over-sampling Technique (SMOTE) applied to the training data. SMOTE generates synthetic minority class samples along line segments connecting existing instances to their nearest neighbors, addressing class imbalance without simple duplication [52–54]. The complete PyCaret workflow is detailed in Algorithm A1.

Table 2 presents the PyCaret configuration parameters. Normalization was enabled for feature scaling, session ID set to 42 for reproducibility, and an 80-20 train-test split was employed. Sixteen machine learning algorithms were evaluated using stratified cross-validation, including linear models, ensemble methods, probabilistic classifiers, support vector machines, and neural networks. Performance metrics included accuracy, AUC-ROC, recall, precision, F1-score, Cohen’s kappa, and Matthews correlation coefficient.

Table 2. PyCaret setup configuration parameters.

Parameter	Value/Setting
Target Variable	diabetes
Session ID	42
Normalization	True
Train-Test Split Ratio	0.80/0.20
Verbose Output	False
Data Type	Excel spreadsheet (.xlsx)
Preprocessing Strategy	Automatic
Cross-Validation Folds	Default (10-fold)

2.3. Multi-Layer Perceptron Architecture

A Multi-Layer Perceptron (MLP) was implemented using scikit-learn’s MLPClassifier to evaluate deep learning performance on the tabular data. The architecture employs multiple hidden layers with ReLU activation functions and L2 regularization to learn non-linear feature interactions while preventing overfitting. Table 3 presents the MLP configuration, including preprocessing, data splitting, SMOTE application, and architecture hyperparameters. The architecture used 128 and 64 neurons in two hidden layers, ReLU activation, Adam optimizer (learning rate = 0.001), L2 regularization (alpha = 0.0001), and early stopping to prevent overfitting. The complete training procedure is formalized in Algorithm A2.

Table 3. Multi-layer perceptron hyperparameters and configuration.

Component	Parameter	Value
Data Splitting	Train-Validation-Test Ratio	60-20-20
	Stratification	Yes
	Random State	42
	Test Size (Initial Split)	0.4
Preprocessing	Numeric Features	StandardScaler
	Categorical Features	OneHotEncoder
	Handle Unknown Categories	Ignore
SMOTE	Application	Training Set Only
	Number of Neighbors	5
	Random State	42

Table 3. *Cont.*

Component	Parameter	Value
MLP Architecture	Hidden Layer Sizes	(128, 64)
	Activation Function	ReLU
	Solver	Adam
	L2 Regularization (alpha)	0.0001
	Initial Learning Rate	0.001
	Maximum Iterations	500
	Early Stopping	True
	Validation Fraction	0.1

2.4. Gradient Boosting Decision Trees

Gradient-boosting decision trees (GBDTs) were evaluated due to their strong performance on tabular data and ability to handle mixed data types without extensive preprocessing [39,40,55,56]. GBDTs sequentially build ensembles of weak learners where each tree corrects errors from previous trees.

Multiple GBDT implementations were assessed: XGBoost (three configurations: class weighting with threshold optimization, class weighting only, and default), LightGBM, CatBoost, Random Forest, Extra Trees, and single Decision Tree. XGBoost variants tested different imbalance-handling strategies to evaluate their impact on minority class performance.

Extra Trees extends the Random Forest concept by introducing additional randomness in both sample selection and feature split selection, potentially improving generalization at the cost of slightly increased bias [57,58]. A single Decision Tree was also evaluated to assess the performance of non-ensemble tree-based methods and provide context for understanding the performance gains achieved through ensemble techniques. Each of these tree-based approaches offers distinct advantages and trade-offs in terms of training speed, memory efficiency, interpretability, and predictive performance, making comprehensive comparison essential for identifying the most suitable approach for the prediabetes prediction task.

Table 4 presents common configuration parameters for tree-based models. Preprocessing included median imputation for missing values and numeric feature selection. Stratified train-test splitting maintained class distribution ratios. Learning curves were analyzed to identify underfitting, overfitting, and convergence patterns. Three XGBoost variants compared imbalance-handling strategies: class weighting with threshold optimization, class weighting only, and default configuration. The complete training and evaluation workflow for all tree-based models is presented in Algorithm A3.

Table 4. Common configuration parameters for tree-based models.

Component	Parameter	Value
Data Preprocessing	Missing Value Strategy	Median Imputation
	Feature Selection	Numeric Features Only
	Target Variable	diabetes (binary)
	Missing Target Handling	Drop rows
Data Splitting	Train-Test Ratio	80-20
	Stratification	Yes
	Random State	42
Cross-Validation	Strategy	StratifiedKFold
	Number of Folds	5
	Shuffle	True
Learning Curves	Scoring Metric	Accuracy
	Training Sizes	8 points (0.1 to 1.0)
	Parallel Jobs	−1 (all cores)

Table 4. *Cont.*

Component	Parameter	Value
Models Evaluated	XGBoost	Class weighting + threshold optimization
	XGBoost	Class weighting only
	XGBoost	Default configuration
	Random Forest	Default configuration
	LightGBM	Default configuration
	Extra Trees	Default configuration
	Decision Tree	Default configuration
	CatBoost	Default configuration

2.5. Imbalance-Specific Learning with RUSBoost

RUSBoost combines Random Under-Sampling with AdaBoost to address class imbalance. In each boosting iteration, RUSBoost randomly removes majority class instances before training the weak learner, ensuring each base classifier trains on more balanced data while maintaining adaptive weighting [59,60]. RUSBoost was implemented using the imbalanced-learn library with Decision Tree base estimators. Hyperparameter optimization employed stratified 5-fold cross-validation with F1-score as the evaluation metric. Table 5 presents the search space and optimal configuration.

Table 5. RUSBoost hyperparameter configuration and search space for prediabetes prediction. The optimal configuration was selected through stratified 5-fold cross-validation with F1-score as the evaluation metric. All experiments utilized a fixed random state of 42 to ensure reproducibility.

Parameter	Search Space	Optimal Value
Number of Estimators	[100, 200]	100
Learning Rate	[0.05, 0.1]	0.05
Base Estimator	Decision Tree	Decision Tree
Maximum Tree Depth	[5, 10]	5
Minimum Samples per Leaf	[5, 10]	10
Cross-Validation Strategy	Stratified K-Fold	K = 5
Optimization Metric	F1-Score	—*
Random State	42	42

Note: *No optimal value applies; F1-Score is the criterion used to evaluate and rank hyperparameter configurations, not a parameter being optimized.

Preprocessing included StandardScaler for numerical features and one-hot encoding for categorical features. Data were split 60-20-20 for training, validation, and test sets using stratified sampling. The final RUSBoost model was trained on the original imbalanced dataset without SMOTE, leveraging RUSBoost's integrated under-sampling mechanism. The algorithmic workflow for RUSBoost training and evaluation is presented in Algorithm A4. This pseudocode outlines the complete process from data preparation through model evaluation, highlighting the key steps involved in implementing RUSBoost for imbalanced classification tasks. Model evaluation included standard classification metrics (accuracy, precision, recall, F1-score) and threshold optimization on the validation set. Feature importance was computed using weighted scores across the ensemble, and confusion matrices quantified classification errors.

2.6. Model Training and Evaluation Framework

All models were trained using stratified train-test splits with appropriate regularization and cross-validation. Learning curves were analyzed to identify underfitting and overfitting patterns by plotting performance metrics as a function of training set size. Models with unstable learning curves or large training-validation gaps were considered unsuitable for deployment regardless of aggregate accuracy [35,36]. This selection criterion

reflects a principled distinction between development-phase performance and deployment-phase reliability. The multi-stage evaluation framework employed in this study, which integrates learning curve convergence, training-validation gap assessment, minority class performance, and validation-to-test stability, aligns with the U.S. Food and Drug Administration's guidance on trustworthy artificial intelligence in biological products, which explicitly requires demonstration of healthy learning dynamics as a prerequisite for deployment consideration [61]. While formal consensus standards for machine learning model selection in clinical applications are still evolving, regulatory frameworks increasingly recognize that aggregate performance metrics alone are insufficient and that evidence of stable, convergent learning behavior is necessary for deployment readiness. The transition from model development to clinical implementation requires systematic assessment of factors beyond aggregate performance, including interpretability and deployment stability [62]. This methodological approach aligns with prior implementations of regulatory-concordant evaluation pipelines for clinical prediction models [63]. Standard metrics such as AUC or accuracy measured on validation sets answer the question "Which model performs best under idealised evaluation conditions?" However, for clinical deployment, the relevant question is "Which model will perform reliably when encountering distribution variations in practice?" Within this framework, no single indicator is sufficient in isolation; rather, the convergent evidence from multiple complementary analyses informs model selection. A model achieving marginally higher validation AUC but exhibiting problematic learning dynamics (such as flat or diverging curves), large training-validation gaps, or a large validation-to-test performance drop demonstrates characteristics that undermine confidence in deployment stability. Conversely, a model exhibiting healthy convergence patterns, acceptable training-validation gaps, adequate minority class detection, and smaller validation-to-test discrepancy provides convergent evidence of robust behavior across the evaluated conditions. Although such internal stability cannot substitute for external validation, alignment of multiple favorable indicators supports a reasoned model selection decision when external data are not yet available, identifying candidates warranting subsequent external validation rather than establishing deployment readiness. Aggregate performance metrics alone are insufficient for clinical model selection when training health and cross-subset stability have not been established. Performance evaluation included accuracy, precision, recall, F1-score, AUC-ROC, and confusion matrices. Emphasis was placed on metrics reflecting performance on both classes and on models demonstrating stable generalization patterns suitable for clinical deployment.

Validation strategies varied across the multi-stage evaluation framework according to the methodological requirements of each algorithmic family. PyCaret employed its default automated pipeline with 80-20 train-test splitting and 10-fold cross-validation to represent typical automated machine learning usage. Models requiring explicit validation sets for early stopping or hyperparameter tuning (MLP, RUSBoost) employed three-way splitting (60-20-20) to prevent information leakage from the test set into model selection decisions. Tree-based models employed 80-20 splitting with 5-fold stratified cross-validation. These differences reflect the distinct computational and optimization requirements of each algorithmic approach rather than arbitrary methodological inconsistency. The initial screening stages served to identify promising candidates using each method's standard configuration; models failing to demonstrate healthy learning dynamics or acceptable performance were eliminated regardless of validation strategy. Crucially, the final comparative evaluation between the two surviving candidate models employed identical data partitions, cross-validation procedures, and evaluation metrics, ensuring that the deployment recommendation is based on directly comparable results under matched conditions.

3. Results

The results are presented in four progressive stages that reflect the multi-dimensional evaluation framework employed in this study. First, automated model screening using PyCaret establishes baseline performance across sixteen machine learning algorithms with and without SMOTE-based class balancing. Second, learning curve analysis of direct training approaches reveals critical model behaviors (including memorization, convergence failures, and training-validation gaps) that remain invisible in aggregate performance metrics. Third, the RUSBoost imbalance-handling approach is evaluated through systematic learning curve analysis across multiple training configurations. Finally, comprehensive test set evaluation compares the two most promising candidates (XGBoost and CatBoost) through confusion matrices and cross-subset stability assessment, ultimately identifying the model most suitable for clinical deployment based on generalization reliability rather than peak validation performance.

3.1. Automated Model Screening with PyCaret

Table 6 presents performance metrics for sixteen machine learning algorithms evaluated with and without SMOTE. Highlighted cells indicate maximum values for each metric. Linear Discriminant Analysis (LDA) with SMOTE achieved maximum values for four metrics: AUC (0.7490), Recall (0.7312), F1-score (0.6284), and MCC (0.3614). Without SMOTE, LDA achieved maximum accuracy (0.6990), while other maximum values were distributed across different models.

Table 6. Performance comparison of PyCaret models with and without SMOTE. Highlighted cells represent maximum values achieved for each respective performance metric.

Model	SMOTE	Acc.	AUC	Recall	Prec.	F1	Kappa	MCC
GBC	Yes	0.6914	0.7436	0.6346	0.5815	0.6068	0.3536	0.3546
	No	0.6946	0.7442	0.5411	0.6042	0.5709	0.3350	0.3362
CatBoost	Yes	0.6900	0.7384	0.5484	0.5944	0.5703	0.3285	0.3292
	No	0.6906	0.7399	0.5129	0.6034	0.5544	0.3199	0.3224
LightGBM	Yes	0.6892	0.7353	0.5581	0.5911	0.5740	0.3297	0.3301
	No	0.6904	0.7381	0.5218	0.6010	0.5584	0.3220	0.3240
AdaBoost	Yes	0.6851	0.7389	0.6185	0.5749	0.5957	0.3384	0.3392
	No	0.6945	0.7440	0.4968	0.6152	0.5496	0.3225	0.3267
RF	Yes	0.6820	0.7258	0.5451	0.5817	0.5627	0.3133	0.3138
	No	0.6825	0.7284	0.4871	0.5938	0.5350	0.2976	0.3010
XGBoost	Yes	0.6808	0.7156	0.5308	0.5824	0.5553	0.3073	0.3081
	No	0.6754	0.7150	0.5063	0.5771	0.5392	0.2904	0.2920
LR	Yes	0.6769	0.7488	0.7188	0.5537	0.6255	0.3497	0.3590
	No	0.6976	0.7491	0.4941	0.6226	0.5508	0.3274	0.3325
LDA	Yes	0.6754	0.7490	0.7312	0.5511	0.6284	0.3502	0.3614
	No	0.6990	0.7490	0.4999	0.6238	0.5549	0.3318	0.3364
Ridge	Yes	0.6753	0.7490	0.7312	0.5509	0.6283	0.3499	0.3611
	No	0.6977	0.7490	0.4873	0.6250	0.5475	0.3258	0.3315
ET	Yes	0.6718	0.7168	0.5174	0.5691	0.5419	0.2871	0.2880
	No	0.6763	0.7163	0.4663	0.5868	0.5195	0.2803	0.2846
SVM	Yes	0.6663	0.7326	0.6885	0.5452	0.6076	0.3245	0.3320
	No	0.6858	0.7323	0.4809	0.6029	0.5337	0.3018	0.3068
NB	Yes	0.6460	0.7033	0.6973	0.5213	0.5966	0.2927	0.3027
	No	0.6799	0.7206	0.5645	0.5751	0.5697	0.3149	0.3150
KNN	Yes	0.6176	0.6508	0.5768	0.4922	0.5311	0.2117	0.2137
	No	0.6468	0.6597	0.4702	0.5339	0.4999	0.2287	0.2298
DT	Yes	0.6083	0.5874	0.5032	0.4795	0.4910	0.1730	0.1732
	No	0.6049	0.5813	0.4863	0.4748	0.4804	0.1618	0.1619
QDA	Yes	0.5574	0.5579	0.4867	0.4425	0.4128	0.0845	0.0966
	No	0.5429	0.5499	0.5339	0.4259	0.4404	0.0798	0.0867

Note: GBC = Gradient Boosting Classifier; CatBoost = CatBoost Classifier; LightGBM = Light Gradient Boosting Machine; AdaBoost = Ada Boost Classifier; RF = Random Forest Classifier; XGBoost = Extreme Gradient Boosting; LR = Logistic Regression; LDA = Linear Discriminant Analysis; Ridge = Ridge Classifier; ET = Extra Trees Classifier; SVM = Support Vector Machine—Linear Kernel; NB = Naive Bayes; KNN = K Neighbors Classifier; DT = Decision Tree Classifier; QDA = Quadratic Discriminant Analysis.

3.2. Learning Curve Analysis: Direct Training Approaches

Figure 2 shows learning dynamics for all evaluated models. LDA exhibits flat training and validation curves at approximately 0.699–0.703 with no learning progression. Flat learning curves, where both training and validation performance remain constant regardless of training set size, indicate that additional training examples did not improve predictive capability within the evaluated sample size range. This observation does not, in isolation, demonstrate complete absence of learning capacity; the model may have acquired useful representations from minimal data that additional examples could not improve upon. However, when interpreted within the multi-stage evaluation framework employed in this study, flat curves represent one indicator among several that collectively inform model assessment. LDA's flat curves indicate that the model's performance is largely determined by its initial parameterisation and is unresponsive to additional training data within the range examined. Combined with LDA's modest absolute performance (approximately 70% accuracy) and its failure to demonstrate the convergence dynamics observed in better-performing models, this pattern suggests limited potential for improvement through data augmentation on this task. For clinical deployment, this represents a notable limitation because it suggests the model would be unlikely to benefit from larger training datasets, though determining whether this reflects inherent algorithmic constraints or task-specific characteristics would require evaluation across multiple datasets.

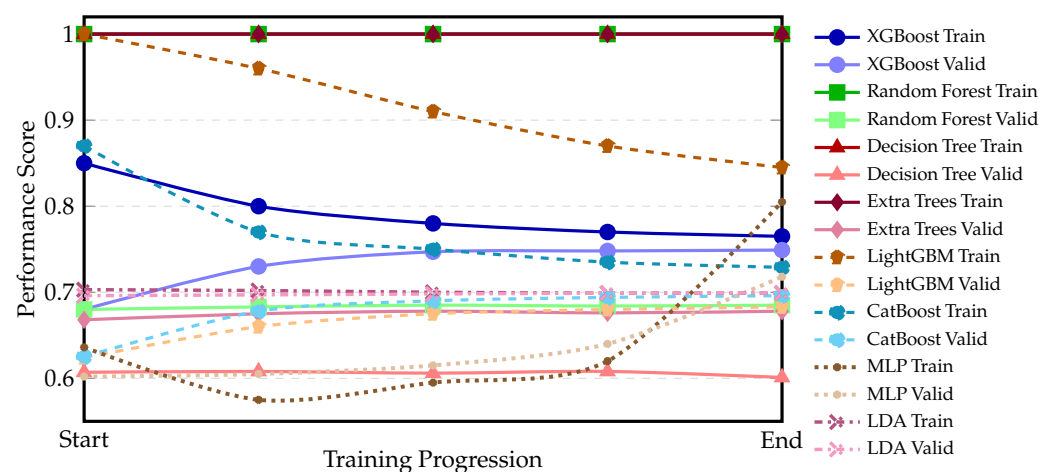


Figure 2. Comparative learning curves showing training and validation performance progression. Tree-based models (Random Forest, Decision Tree, Extra Trees) exhibit flat curves at 1.0 indicating memorization; MLP shows non-convergent behavior with continuously rising curves (training: 0.636 $\rightarrow$ 0.805; validation: 0.602 $\rightarrow$ 0.718). LightGBM demonstrates incomplete training with large training-validation gap. XGBoost and CatBoost display convergence and plateau dynamics consistent with healthy learning, with XGBoost showing the most favorable validation performance (0.749) and smallest training-validation gap (0.016) among evaluated models.

Random Forest, Decision Tree, and Extra Trees show flat training curves at 1.0, indicating memorization. MLP displays non-convergent behavior with continuously rising curves (training: 0.636 $\rightarrow$ 0.805; validation: 0.602 $\rightarrow$ 0.718). LightGBM shows incomplete learning with a 0.16 training-validation gap. CatBoost demonstrates convergence with training decreasing (0.87 $\rightarrow$ 0.729) and validation increasing (0.675 $\rightarrow$ 0.696), yielding a 0.033 gap. XGBoost exhibits favorable learning dynamics: training descends (0.85 $\rightarrow$ 0.765), validation ascends (0.73 $\rightarrow$ 0.749), with a minimal 0.016 gap and highest validation performance among evaluated models (0.749). XGBoost performance differed between PyCaret and direct training implementations. PyCaret yielded accuracy of 0.6808 (with SMOTE) and AUC of 0.7156. Direct training achieved validation AUC of 0.749, representing a 4.7% improvement.

3.3. Learning Curve Analysis: RUSBoost Approach

Learning curves were generated for all models to assess performance as a function of training data availability. The learning curve analysis systematically varied the amount of training data used for model fitting while maintaining fixed validation and test sets. The methodology progressively increased the proportion of the training partition utilized, starting from small fractions and incrementing to the complete training set. This progressive expansion occurred exclusively within the designated training portion of the data. The validation and test sets remained completely separate and untouched throughout the entire learning curve generation process, ensuring unbiased performance estimation at each training size increment.

Figure 3 illustrates learning curves obtained from the RUSBoost model using three different training data proportions: 70%, 80%, and 90% of the available training data. The learning curve analysis employed stratified 5-fold cross-validation at each training size to generate robust performance estimates. Each set of curves (corresponding to one training proportion) displays training F1-score decreasing as more training samples are utilized, while cross-validation F1-score increases with additional training data, demonstrating how model performance improves when trained on progressively larger subsets of the training partition. Convergence of the two curves occurs at approximately 6000 to 8000 training samples across all three training proportions, with relatively small gaps between training and validation performance at the largest training set sizes.

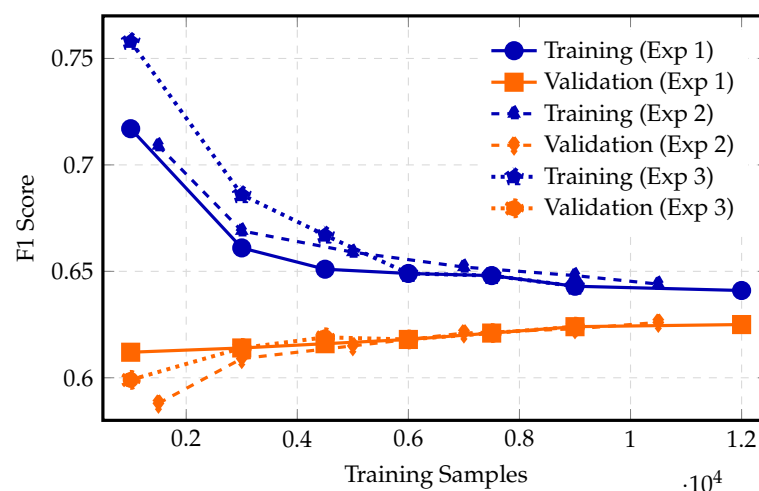


Figure 3. Learning curves for RUSBoost demonstrating inconsistent learning dynamics across three data proportions from the same training session: 70% (Exp 1), 80% (Exp 2), and 90% (Exp 3). Exp 2 appears to show healthy convergence (validation gain of 0.038), while Exp 1 reveals a problematic flat validation curve (gain of only 0.013). This inconsistency within a single model exemplifies how cherry-picking the most favorable experimental configuration (Exp 2) while minimizing discussion of concerning results (Exp 1) can create a misleading impression of model stability and deployment readiness. The variability suggests fundamental training instability rather than robust performance suitable for clinical applications.

The inconsistent learning dynamics observed across multiple training configurations within the same RUSBoost model revealed fundamental instability unsuitable for clinical deployment. While Experiment 2 appeared to demonstrate healthy convergence with a validation gain of 0.038, Experiment 1 exhibited a problematic flat validation curve with minimal learning (gain of only 0.013). This substantial variability across different data proportions from identical training sessions indicates that RUSBoost's performance is highly sensitive to specific data configurations rather than reflecting robust, generalizable learning. Most concerning, the validation F1 scores across all three experiments remained below 0.626,

substantially lower than the validation performance achieved by both XGBoost (0.749 AUC) and CatBoost (0.696 accuracy) in direct training approaches. The combination of training instability, configuration sensitivity, and inferior validation performance led to RUSBoost's elimination from final consideration. Moreover, RUSBoost's validation set performance is lower than both XGBoost and CatBoost, further confirming the unsuitability of this approach for clinical deployment. Subsequent evaluation therefore focused exclusively on the two algorithms demonstrating both acceptable learning dynamics and superior validation performance: XGBoost and CatBoost.

3.4. Statistical Comparison of CatBoost and XGBoost

To determine whether the observed performance differences between CatBoost and XGBoost exceed sampling variability, three complementary statistical analyses were conducted: McNemar's test for paired classification disagreements, bootstrap confidence intervals on accuracy and AUC differences, and paired t -tests on cross-validation fold accuracies.

McNemar's test examines whether two classifiers make errors on different subsets of instances. Among the 3,228 test samples, 167 instances were correctly classified by XGBoost but incorrectly by CatBoost, while 213 instances showed the reverse pattern. The resulting chi-squared statistic was 5.33 ($p = 0.021$), indicating that the classifiers exhibit significantly different error patterns at the $\alpha = 0.05$ level.

Bootstrap resampling (10,000 iterations) was employed to construct 95% confidence intervals on performance metric differences. The observed accuracy difference (CatBoost – XGBoost) was 0.0143, with a 95% confidence interval of [0.0025, 0.0257]. Because this interval excludes zero, the accuracy advantage of CatBoost is statistically significant. Similarly, the observed AUC difference was 0.0188 (CatBoost: 0.7394; XGBoost: 0.7206), with a 95% confidence interval of [0.0107, 0.0270], again excluding zero and confirming a significant discriminative performance advantage for CatBoost.

Paired t -tests on 5-fold cross-validation accuracies provided additional confirmation. CatBoost achieved a mean CV accuracy of 0.6897 (SD = 0.0067) compared to XGBoost's 0.6768 (SD = 0.0090). The paired t -test yielded $t(4) = 7.62$ ($p = 0.0016$), with a 95% confidence interval on the mean difference of [0.0082, 0.0175]. Figure 4 and Table 7 summarize these findings.

Table 7. Statistical comparison of CatBoost versus XGBoost performance. All tests indicate statistically significant superiority of CatBoost at $\alpha = 0.05$.

Metric	XGBoost	CatBoost	Difference [95% CI]	p -Value
Test Accuracy	0.6778	0.6921	0.014 [0.003, 0.026]	0.021 ^a
Test AUC	0.7206	0.7394	0.019 [0.011, 0.027]	<0.001 ^b
CV Accuracy (mean $\pm$ SD)	0.677 $\pm$ 0.009	0.690 $\pm$ 0.007	0.013 [0.008, 0.018]	0.002 ^c

^a McNemar's test; ^b Bootstrap (10,000 iterations); ^c Paired t -test (5 folds)

Collectively, these analyses suggest that the observed performance differences between CatBoost and XGBoost are unlikely to be attributable solely to sampling variability. The preference for CatBoost as the candidate for further evaluation is supported by statistically significant differences across multiple complementary tests and metrics, providing preliminary evidence that this advantage may replicate in independent samples drawn from similar populations, though external validation remains necessary to confirm generalizability.

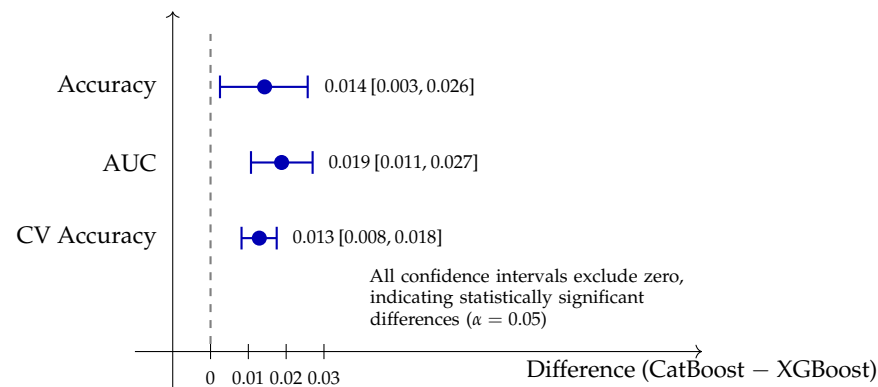


Figure 4. Forest plot of performance differences between CatBoost and XGBoost with 95% confidence intervals. Accuracy and AUC differences were estimated via bootstrap resampling (10,000 iterations); CV accuracy difference was estimated via paired *t*-test across five folds. The dashed vertical line represents zero (no difference). All intervals exclude zero, indicating CatBoost significantly outperforms XGBoost across all metrics.

3.5. Model Calibration Assessment

Beyond discriminative performance, the clinical utility of a predictive model depends on whether predicted probabilities accurately reflect true outcome frequencies. Calibration analysis was conducted to assess this property for both CatBoost and XGBoost (Figure 5). Calibration curves were constructed by binning test set predictions into deciles based on predicted probability, then plotting the mean predicted probability against the observed proportion of prediabetic cases within each bin. Perfect calibration corresponds to the diagonal reference line, where predicted probabilities exactly match observed outcome rates.

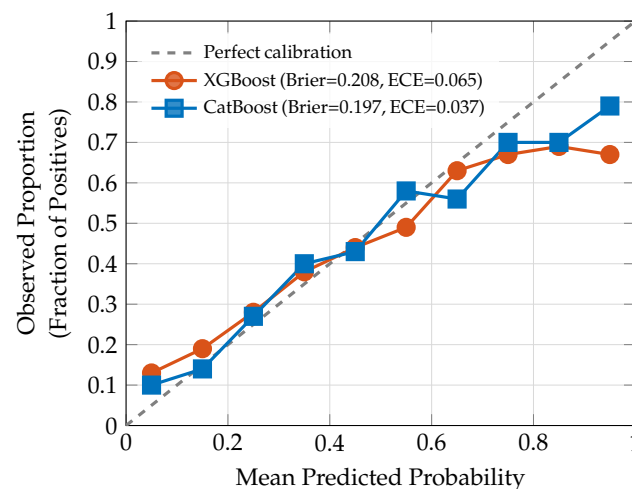


Figure 5. Calibration curves comparing CatBoost and XGBoost predicted probabilities to observed prediabetes proportions. The dashed diagonal represents perfect calibration where predicted probabilities equal observed outcome frequencies. Points represent decile bins of predicted probabilities. CatBoost (blue squares) demonstrates superior calibration with lower Brier score (0.197 vs. 0.208) and substantially lower expected calibration error (ECE = 0.037 vs. 0.065, a 43% reduction). XGBoost (orange circles) shows systematic underestimation of observed risk at higher predicted probabilities (>0.7), with predictions plateauing around 0.67–0.69 observed proportion regardless of predicted probability, whereas CatBoost maintains appropriate calibration with observed proportions reaching 0.79 at the highest prediction decile.

Visual inspection of the calibration curves reveals that both models demonstrate reasonable calibration in the lower probability range (0.0–0.5), with curves closely tracking the diagonal. However, notable differences emerge at higher predicted probabilities. CatBoost

maintains closer adherence to the diagonal throughout the probability range, whereas XGBoost exhibits systematic underestimation of risk at higher predicted probabilities (0.7–1.0 range). Specifically, when XGBoost predicts probabilities exceeding 0.8, observed prediabetes rates plateau around 0.67–0.69, indicating that XGBoost’s high-confidence predictions overestimate certainty without corresponding increases in true positive rates. In contrast, CatBoost predictions in the 0.9–1.0 range correspond to observed proportions approaching 0.80, demonstrating superior calibration at high-risk thresholds most relevant for clinical decision-making.

Quantitative calibration metrics confirm these visual observations. CatBoost achieved a lower Brier score (0.197) compared to XGBoost (0.208), indicating superior overall calibration and discrimination. The Brier score, calculated as the mean squared difference between predicted probabilities and binary outcomes, penalizes both poor calibration and poor discrimination, with lower values indicating better performance. More notably, CatBoost demonstrated substantially lower expected calibration error (ECE = 0.037) compared to XGBoost (ECE = 0.065), representing a 43% reduction in calibration error. ECE quantifies the weighted average absolute difference between predicted probabilities and observed frequencies across bins, providing a direct measure of calibration quality independent of discrimination.

These calibration characteristics have important implications for clinical deployment. In a screening context where predicted probabilities inform risk stratification or patient counseling, well-calibrated predictions enable clinicians to communicate risk estimates that correspond meaningfully to actual prediabetes likelihood. For example, a CatBoost prediction of 0.7 can be interpreted as approximately 70% probability of prediabetes, whereas the same prediction from XGBoost corresponds to only approximately 63% observed risk, i.e., a clinically meaningful discrepancy that could affect patient communication and downstream decision-making. CatBoost’s superior calibration, combined with its higher discriminative performance established in previous section, further supports its selection as the recommended model for potential clinical application.

3.6. Predicted Probability Distributions and Class Separation

To understand the discriminative behavior of each model beyond summary metrics, the distributions of predicted probabilities were examined separately for non-diabetic and prediabetic individuals (Figure 6). These distributions reveal how each model assigns risk scores to the two classes and illuminate the source of classification errors.

For XGBoost (Figure 6a), non-diabetic individuals ($y = 0$, upper portion) receive predicted probabilities concentrated in the 0.0–0.2 range, though with a long tail extending toward higher values. Prediabetic individuals ($y = 1$, lower portion) exhibit a broadly distributed pattern spanning 0.1–0.8, with peak density around 0.5–0.6. Critically, substantial overlap exists between the two distributions in the 0.2–0.5 range, representing individuals for whom XGBoost predictions are ambiguous and classification errors are likely.

CatBoost (Figure 6b) demonstrates improved class separation. Non-diabetic individuals show a sharper concentration at very low predicted probabilities (0.0–0.15), with density declining more rapidly than XGBoost as probability increases. Prediabetic individuals exhibit peak density around 0.4–0.5, with the distribution extending further into the 0.6–0.8 range compared to XGBoost. While overlap between classes persists, explaining the moderate overall accuracy of 69.2%, CatBoost achieves greater separation, consistent with its higher AUC (0.739 vs. 0.721) and superior calibration documented above.

The persistent overlap between class distributions provides important context for interpreting the moderate predictive performance reported in this study. Unlike diagnostic tasks where biomarkers may cleanly separate disease states, prediabetes risk prediction

from lifestyle questionnaire data involves inherently overlapping population characteristics. Both models assign probabilities near 0.5 to a substantial proportion of individuals, reflecting genuine uncertainty rather than model failure.

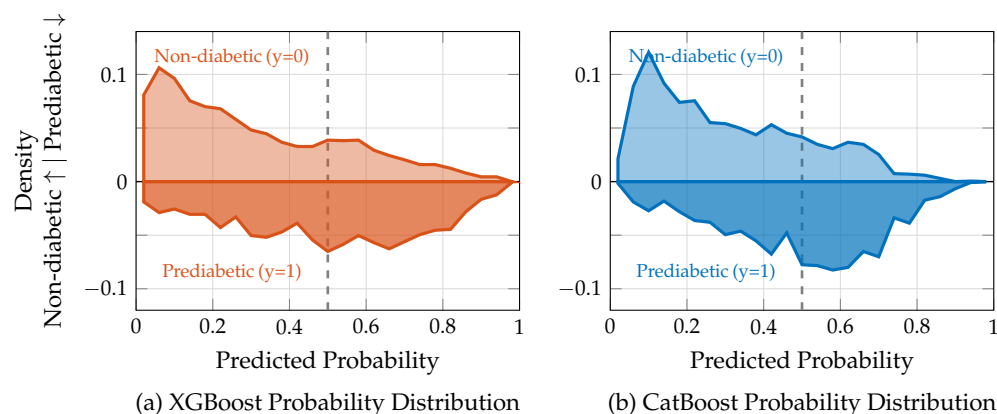


Figure 6. Predicted probability distributions stratified by true class for XGBoost (a) and CatBoost (b). Upper portions (lighter shading) show density of predictions for non-diabetic individuals ($y = 0$); lower portions (darker shading) show prediabetic individuals ($y = 1$). The vertical dashed line indicates the default classification threshold (0.5). CatBoost demonstrates superior class separation, with non-diabetic predictions more sharply concentrated at low probabilities and prediabetic predictions extending further into higher probability ranges. The substantial overlap between distributions in both models (particularly in the 0.2–0.5 range) explains the moderate overall accuracy and supports the recommendation that predictions in this range warrant additional clinical assessment rather than definitive classification.

The vertical dashed line at 0.5 in Figure 6 represents the default classification threshold. Threshold optimization could shift this boundary to favor sensitivity or specificity depending on clinical priorities; however, the fundamental distributional overlap constrains achievable performance regardless of threshold selection. Future work incorporating additional predictive features (such as laboratory values or genetic risk scores) may reduce this overlap and improve discriminative separation.

3.7. Test Set Evaluation: XGBoost Versus CatBoost

XGBoost and CatBoost were selected for final evaluation based on acceptable learning dynamics. Figure 7 presents confusion matrices for both models on the test set. XGBoost achieved 1593 true negatives, 634 true positives, 423 false positives, and 578 false negatives (test accuracy: 0.690). CatBoost achieved 1609 true negatives, 630 true positives, 407 false positives, and 582 false negatives (test accuracy: 0.694).

1593	423		
578	634		
0	1		
Predicted Classes			
(a) XGBoost			

1609	407	0	True
582	630	1	Classes
0	1		
Predicted Classes			
(b) CatBoost			

Figure 7. Confusion matrices showing the number of true and predicted classes for XGBoost (Tuned) and CatBoost models for prediabetes prediction. The rows represent true labels (0 = non-prediabetic, 1 = prediabetic) and columns represent predicted labels in the same order. The numbers in blue indicate correct predictions (true positives and true negatives) and the numbers in orange indicate incorrect predictions (false positives and false negatives).

Figure 8 presents accuracy across training, validation, and test subsets. XGBoost: training (0.765), validation (0.749), test (0.690). CatBoost: training (0.729), validation (0.696), test (0.694). XGBoost shows a 5.9 percentage point decline from validation to test. CatBoost maintains stability with only a 0.2 percentage point decline. Both models achieve similar test accuracy (XGBoost: 0.690; CatBoost: 0.694). Both models demonstrate similar test performance (XGBoost: 0.690; CatBoost: 0.694). XGBoost shows larger validation-to-test decline (5.9 percentage points) compared to CatBoost (0.2 percentage points).

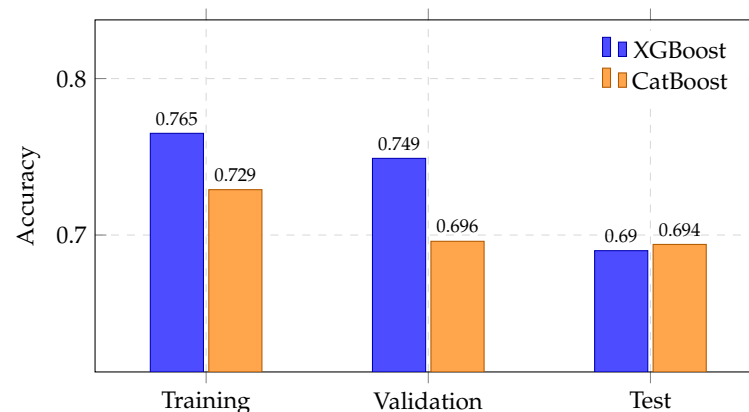


Figure 8. Accuracy comparison for XGBoost and CatBoost across training, validation, and test subsets. CatBoost demonstrates superior validation-test stability with only a 0.2 percentage point decline, indicating robust generalization. XGBoost exhibits concerning validation-to-test degradation of 5.9 percentage points, suggesting potential instability when encountering distribution shifts expected in real-world deployment across diverse demographics.

4. Discussion

This section traces the multi-stage evaluation journey that led to the final model selection, demonstrating how the conclusions systematically evolved as progressively deeper analytical perspectives revealed limitations invisible in previous assessment stages. Each evaluation stage challenged and often reversed conclusions drawn from prior analyses, illustrating why detailed multi-dimensional assessment proves vital for identifying clinically deployable models rather than merely high-performing development artifacts.

4.1. From Aggregate Metrics to Clinical Applicability

The journey to identifying the most clinically applicable algorithm exemplifies the critical importance of multi-faceted model evaluation beyond reliance on single performance metrics. This investigation revealed a progression of insights that fundamentally challenged initial conclusions drawn from automated framework analysis, ultimately demonstrating that real-world deployment readiness demands multi-layered assessment spanning aggregate metrics, learning dynamics, minority class performance, and cross-subset stability.

4.1.1. Stage 1: Initial Promise of LDA

The PyCaret analysis (Section 3.1) initially suggested LDA with SMOTE as the optimal choice, achieving maximum values for four of seven metrics: AUC (0.7490), Recall (0.7312), F1-score (0.6284), and MCC (0.3614). This strong performance across multiple evaluation criteria created a compelling initial case for LDA deployment. However, relying solely on these aggregate cross-validation metrics would have led to a fundamentally flawed model selection, as subsequent analyses revealed critical limitations invisible in percentage-based performance summaries [35,36].

4.1.2. Stage 2: Learning Curves Expose LDA's Fatal Flaw

The learning curve analysis (Section 3.2) exposed LDA's principal limitation: completely flat training and validation curves at approximately 0.699–0.703 with no observable improvement as training set size increased. While flat curves alone do not demonstrate complete absence of learning capacity (the model may have learned effectively from minimal data), when combined with LDA's modest absolute performance and its lack of the convergence dynamics exhibited by better-performing models, this pattern contributes to an unfavorable overall assessment. While LDA's modest 70% performance level is not catastrophic, the flat-curve characteristic suggests limited potential for improvement through increased training data on this task. Practitioners relying exclusively on PyCaret's aggregate metrics would have selected a model showing no evidence of continued learning from additional data, which, in combination with its inferior performance relative to models demonstrating healthy convergence, supports its elimination from consideration for clinical deployment. Whether such models would exhibit reduced adaptability to distribution shifts remains a hypothesis requiring external validation to confirm.

The learning curves further revealed problematic behaviors disqualifying other candidates: Random Forest, Decision Tree, and Extra Trees exhibited flat training curves at 1.0 (complete memorization); MLP showed non-convergent behavior with continuously rising curves; and LightGBM demonstrated incomplete learning with excessive training-validation gaps. This systematic elimination left two viable candidates: XGBoost (training-validation gap: 0.016, validation AUC: 0.749) and CatBoost (gap: 0.033, validation AUC: 0.696).

4.1.3. Stage 3: XGBoost's Apparent Superiority

Based on learning curve analysis alone, XGBoost appeared to be the clear winner, exhibiting optimal convergence dynamics with the minimal training-validation gap (0.016) and highest validation performance (0.749). The 5.3 percentage point validation advantage over CatBoost (0.749 vs. 0.696) seemed decisive, suggesting XGBoost offered superior discriminative capability. Many model selection processes would have terminated at this stage, deploying XGBoost based on its strong validation performance and excellent learning dynamics.

4.1.4. Stage 4: Cross-Subset Analysis Reveals the Truth

However, confusion matrix analysis (Section 3.7) and cross-subset stability evaluation (Figure 8) fundamentally altered the selection landscape. While both models demonstrated acceptable minority class detection (XGBoost: 634 true positives; CatBoost: 630), the critical differentiator emerged in cross-subset stability. XGBoost exhibited a concerning 5.9 percentage point decline from validation (0.749) to test (0.690), whereas CatBoost maintained remarkable stability with only a 0.2 percentage point drop (0.696 to 0.694).

This validation-to-test degradation pattern has implications for model selection, particularly when considered alongside the learning curve and convergence analyses presented in earlier sections. Cross-dataset performance inherently declines when models encounter data from different collection contexts, demographics, or geographic populations. XGBoost's substantial 5.9 percentage point drop between validation and test partitions from the same underlying dataset raises concerns about sensitivity to data partitioning; while this does not directly predict external performance, it represents an additional indicator (beyond the convergence analyses) that warrants caution. In contrast, CatBoost's demonstrated stability (0.2 vs. 5.9 percentage points) provides convergent evidence, when combined with its healthy learning dynamics and acceptable training-validation gap, that this model exhibits more consistent behavior across internal data subsets. Although internal stability

cannot substitute for external validation, the alignment of multiple favorable indicators supports CatBoost as the more promising candidate for subsequent external validation studies in settings where such data are not yet available.

Critically, XGBoost's validation advantage of 5.3 percentage points evaporates at test time where both models achieve nearly identical accuracy (0.690 vs. 0.694). This convergence reveals that XGBoost's superior validation performance does not translate to genuine advantages on unseen data, suggesting the validation set may have idiosyncratic characteristics that favor XGBoost but do not generalize.

4.1.5. Model Selection Summary

This evaluation establishes CatBoost as the recommended choice for potential future clinical deployment, pending external validation. It should be acknowledged that the present analysis relies on internal validation using train-validation-test partitions from a single dataset rather than external validation on an independent cohort. The three-way splitting strategy (train-validation-test) was employed because it provides a principled separation of concerns: the test set remains entirely untouched during model development and hyperparameter selection, preserving its integrity as an unbiased estimate of held-out performance. This approach is more robust than repeated partitions with two-way splitting, where the same data alternates between training and evaluation roles across folds. However, this design choice means that traditional cross-validation confidence intervals are not directly applicable to the final test set evaluation; instead, bootstrap confidence intervals and paired statistical tests were employed to quantify uncertainty in performance comparisons.

Critically, validation-to-test stability within a single dataset provides evidence against overfitting to the validation set but does not constitute direct evidence of generalisability under distribution shifts. True generalisability can only be established through evaluation on data collected from different populations, time periods, or healthcare settings. High validation-to-test stability is therefore a necessary but not sufficient condition for deployment readiness, and external validation remains essential before any clinical implementation.

Furthermore, it must be acknowledged that the observed test accuracy of approximately 0.694 and AUC of approximately 0.75 represent moderate rather than high predictive performance. Such performance levels may be suitable for initial risk stratification or screening applications where the goal is to identify individuals warranting further clinical assessment, but would be insufficient for definitive diagnostic or treatment decisions. Future work should examine model calibration to ensure predicted probabilities correspond to observed event rates, and decision curve analysis should be conducted to quantify net clinical benefit across plausible threshold ranges before deployment in patient care pathways. Real-world clinical applications demand consistent, stable performance across diverse patient populations, healthcare settings, and geographic regions [27,64]. CatBoost's 29.5-fold superior validation-to-test stability (0.2 vs. 5.9 percentage point drop) suggests greater robustness to data partitioning, which may indicate more reliable performance when encountering distribution variations, though this hypothesis requires confirmation through external validation. While XGBoost demonstrates stronger performance during model development (validation phase), CatBoost's superior stability across internal data subsets makes it the more promising candidate for further evaluation in settings where consistent performance is prioritized over peak validation metrics. These findings represent preliminary evidence supporting CatBoost as the preferred model for subsequent external validation studies rather than established deployment readiness.

This multi-stage evaluation process demonstrates the critical inadequacy of relying on any single analytical perspective. Aggregate metrics suggested LDA; learning curves indicated XGBoost; only the combination of confusion matrices and cross-subset stability analysis revealed CatBoost's true superiority for real-world deployment. This progression underscores the essential principle that clinically applicable models must be selected through comprehensive evaluation spanning multiple analytical dimensions, as premature conclusions based on incomplete evidence lead to fundamentally flawed deployment decisions.

Figure 9 visualizes the multi-stage decision process that led to CatBoost selection, illustrating how each analytical stage systematically eliminated candidates based on specific failure modes until identifying the model most suitable for clinical deployment.

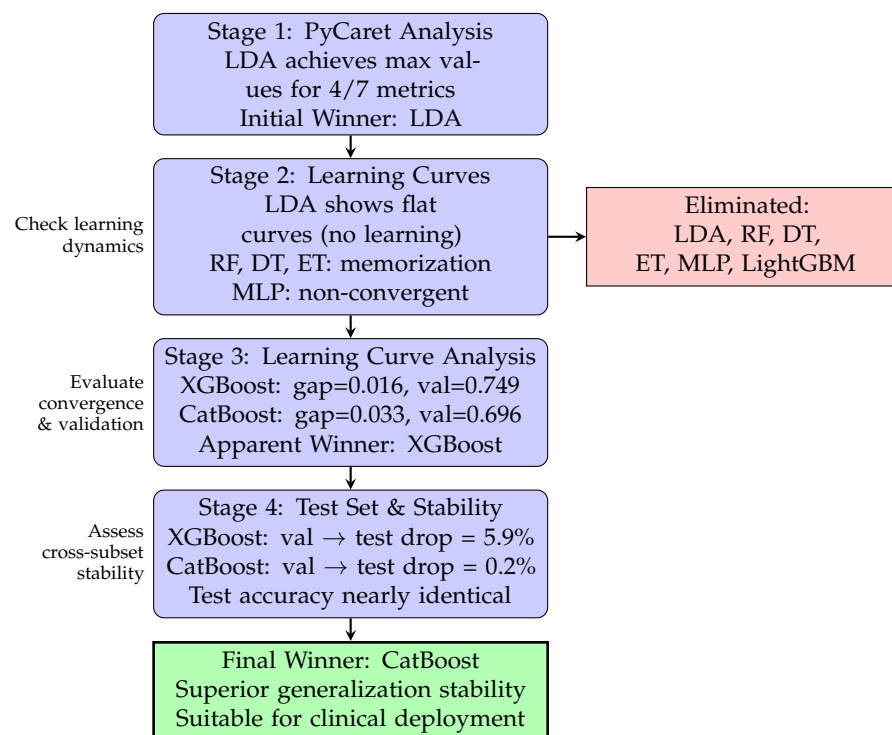


Figure 9. Decision flowchart illustrating the multi-stage model selection process. Each stage revealed critical limitations of apparently superior models, ultimately identifying CatBoost as most suitable for clinical deployment based on validation-to-test stability rather than absolute validation performance.

4.2. Implications of Multi-Stage Evaluation

The methodological journey from PyCaret's aggregate metrics through learning curves to final cross-subset validation carries important implications for machine learning in healthcare. Automated frameworks like PyCaret provide valuable rapid prototyping capabilities but may not fully exploit model potential for specialized medical prediction tasks with domain-specific characteristics such as class imbalance and complex feature interactions [65–67]. The performance improvements achieved through direct training over PyCaret demonstrate that targeted hyperparameter optimization, tailored preprocessing pipelines, and imbalance-specific training strategies yield meaningful returns in predictive capability. Specifically, XGBoost showed a 4.7% performance improvement (direct training validation AUC: 0.749 vs. PyCaret with SMOTE: 0.7156), while CatBoost achieved validation accuracy of 0.696 through direct training compared to 0.6900 through PyCaret with SMOTE.

Direct training implementations differed from PyCaret in several critical aspects. Both XGBoost and CatBoost implementations utilized 5-fold rather than 10-fold cross-validation,

explicit stratification, and a standardized preprocessing pipeline including median imputation for missing values and StandardScaler normalization. For XGBoost specifically, direct training employed expanded hyperparameter search space, explicit class weighting for imbalance handling, threshold optimization for F1 maximization, and AUC-focused evaluation rather than accuracy-focused optimization. In contrast, CatBoost utilized default configurations but benefited from the improved preprocessing pipeline and cross-validation strategy. These targeted optimizations for the imbalanced prediabetes prediction task explain the performance improvements over PyCaret's generalized automated approach, with XGBoost showing more substantial gains (4.7% improvement) due to its intensive hyperparameter tuning while CatBoost demonstrated that even default configurations with proper preprocessing can yield competitive results.

However, even direct training with superior validation performance (XGBoost's 0.749) proves insufficient without cross-subset stability analysis. The dramatic reversal where CatBoost's initially inferior validation performance (0.696) proved superior for deployment due to minimal test degradation (0.2 vs. 5.9 percentage points) underscores that validation performance alone provides an incomplete picture of real-world applicability. Models must demonstrate not only strong performance on development data but also stable generalization patterns that transfer reliably to diverse deployment scenarios.

For prediabetes prediction specifically, CatBoost's combination of acceptable minority class detection (630 true positives), reasonable test accuracy (0.694), and exceptional validation-to-test stability positions it as the most trustworthy choice for clinical screening applications. While neither XGBoost nor CatBoost achieves perfect sensitivity (both miss approximately 48% of prediabetic cases), CatBoost's superior generalization stability provides greater confidence that this performance level will remain consistent across diverse patient populations and healthcare settings.

4.3. Limitations and Future Directions

The present analysis focused on a single dataset and specific demographic population. Future work should validate CatBoost's performance across multiple external datasets representing diverse geographic regions, age groups, and healthcare systems to confirm the generalization stability observed in within-dataset cross-validation. Additionally, the moderate sensitivity ($\approx 52\%$) achieved by both final candidates indicates room for improvement through ensemble methods, advanced feature engineering, or incorporation of additional data modalities beyond questionnaire responses. Threshold optimization techniques could be explored to adjust the sensitivity-specificity trade-off based on specific clinical deployment requirements, potentially sacrificing some specificity to improve identification of at-risk individuals.

5. Conclusions

The evaluation of machine learning algorithms for clinical deployment is not straightforward but requires a detailed, multi-faceted approach that extends far beyond conventional performance metrics. This study demonstrates that relying exclusively on cross-validated accuracy or automated framework rankings leads to fundamentally flawed model selection, as models achieving maximum validation metrics may exhibit flat learning curves indicating exhausted learning capacity, while others demonstrating superior validation performance can suffer substantial generalization degradation unsuitable for real-world deployment. Healthcare applications demand systematic evaluation spanning aggregate performance metrics, learning curve analysis to detect memorization and convergence failures, minority class detection capability to ensure equitable performance across patient populations, and cross-subset stability assessment to identify potential overfitting to vali-

dation data, though such internal stability does not substitute for external validation when assessing reliability under distribution shifts. The findings reveal that deployment-ready models must prioritize consistent, stable performance over peak development metrics, as algorithms with inferior validation performance may ultimately prove more suitable for real-world use when they demonstrate superior generalization stability. In healthcare settings where prediction errors directly impact patient outcomes, this multi-dimensional evaluation framework proves vital for identifying candidate models warranting external validation rather than merely high-performing development artifacts. The internal stability demonstrated through validation-to-test consistency represents a necessary but not sufficient condition for deployment readiness; confirmation of reliability across diverse demographics, geographic regions, and temporal contexts requires prospective external validation studies.

Author Contributions: Conceptualization, M.S. and M.T.; methodology, M.S. and M.T.; software, M.S. and M.T.; validation, M.S. and M.T.; formal analysis, M.S. and M.T.; investigation, M.S. and M.T.; resources, M.S. and M.T.; data curation, M.S. and M.T.; writing—original draft preparation, M.S. and M.T.; writing—review and editing, M.S. and M.T.; visualization, M.S. and M.T.; supervision, M.T.; project administration, M.T.; funding acquisition, M.T. 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: All datasets utilized in this study were obtained from publicly available repositories and are in the public domain. The primary training datasets have been properly cited throughout this manuscript with their corresponding references provided in the bibliography section. Complete methodological details including all preprocessing algorithms, architectural modifications, training configurations, and validation protocols have been documented through detailed pseudocode descriptions provided in Appendix A to ensure full reproducibility of the experimental procedures. Model hyperparameters, optimization settings, and evaluation metrics are explicitly specified within the Methods Section 2. Readers requiring additional information regarding implementation details, data processing pipelines, or access to trained model parameters are encouraged to contact the corresponding author for further enquiries.

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

Abbreviations

The following abbreviations are used in this manuscript:

AdaBoost	Adaptive Boosting
AUC	Area Under the Curve
CatBoost	Categorical Boosting
DT	Decision Tree
ECE	Expected Calibration Error
ET	Extra Trees
F1	F1-Score
GBC	Gradient Boosting Classifier
GBDT	Gradient-Boosting Decision Trees
KNN	K Neighbors Classifier
LDA	Linear Discriminant Analysis
LightGBM	Light Gradient Boosting Machine
LR	Logistic Regression
MCC	Matthews Correlation Coefficient

MLP	Multi-Layer Perceptron
NB	Naive Bayes
QDA	Quadratic Discriminant Analysis
ReLU	Rectified Linear Unit
RF	Random Forest
Ridge	Ridge Classifier
ROC	Receiver Operating Characteristic
RUSBoost	Random Under-Sampling Boost
SMOTE	Synthetic Minority Over-sampling Technique
SVM	Support Vector Machine
XGBoost	Extreme Gradient Boosting

Appendix A

Appendix A.1

Algorithm A1 PyCaret automated machine learning workflow.

Require: Dataset file path, target variable name

Ensure: Trained model with performance metrics

- 1: Input: Load questionnaire dataset from Excel file
 - 2: Initialize: Import PyCaret classification module
 - 3: Setup Phase:
 - 4: Configure experimental environment with:
 - 5: ▷ Specify target variable (diabetes status)
 - 6: ▷ Set random seed for reproducibility
 - 7: ▷ Enable feature normalization
 - 8: ▷ Define train-test split ratio
 - 9: ▷ Suppress verbose output
 - 10: Model Comparison Phase:
 - 11: Execute compare_models() function:
 - 12: ▷ Train all available classifiers
 - 13: ▷ Apply cross-validation
 - 14: ▷ Compute performance metrics for each model
 - 15: ▷ Rank models by selected criterion
 - 16: ▷ Return best performing model
 - 17: Evaluation Phase:
 - 18: Analyze best model using evaluate_model():
 - 19: ▷ Generate confusion matrix
 - 20: ▷ Plot ROC-AUC curve
 - 21: ▷ Display feature importance
 - 22: ▷ Create learning curves
 - 23: ▷ Compute calibration plots
 - 24: Output: Return trained model and comprehensive evaluation metrics
-

Algorithm A2 Multi-layer perceptron training procedure.

```

1: Input: Dataset with features  $X$  and target variable  $y$ 
2: Output: Trained MLP model with performance metrics
3: // Step 1: Data Preprocessing
4: Identify categorical columns:  $\{EC\_stt\_1, EC\_occp\}$ 
5: Identify numeric columns: all remaining feature columns
6: Create ColumnTransformer with StandardScaler for numeric and OneHotEncoder for
   categorical
7:  $X_{processed} \leftarrow$  Apply preprocessing to  $X$ 
8: // Step 2: Data Partitioning
9: Split data stratified by  $y$  with random state 42:
10:  $X_{train}, X_{temp}, y_{train}, y_{temp} \leftarrow \text{train\_test\_split}(X_{processed}, y, \text{test\_size}=0.4)$ 
11: Split temporary set:
12:  $X_{val}, X_{test}, y_{val}, y_{test} \leftarrow \text{train\_test\_split}(X_{temp}, y_{temp}, \text{test\_size}=0.5)$ 
13: // Step 3: Handle Class Imbalance
14: Initialize SMOTE with  $k\_neighbors = 5$  and  $\text{random\_state}=42$ 
15:  $X_{train\_smote}, y_{train\_smote} \leftarrow \text{SMOTE.fit\_resample}(X_{train}, y_{train})$ 
16: Display class distribution before and after SMOTE
17: // Step 4: Model Configuration
18: Initialize MLPClassifier:
19:   hidden_layer_sizes = (128, 64)
20:   activation = 'relu'
21:   solver = 'adam'
22:   alpha = 0.0001
23:   learning_rate_init = 0.001
24:   max_iter = 500
25:   early_stopping = True
26:   validation_fraction = 0.1
27:   random_state = 42
28: // Step 5: Model Training
29: Train model on SMOTE-balanced training data:
30:    $\text{mlp\_model.fit}(X_{train\_smote}, y_{train\_smote})$ 
31: // Step 6: Evaluation
32:  $y_{test\_pred} \leftarrow \text{mlp\_model.predict}(X_{test})$ 
33: Calculate performance metrics:
34:   accuracy  $\leftarrow \text{accuracy\_score}(y_{test}, y_{test\_pred})$ 
35:   precision  $\leftarrow \text{precision\_score}(y_{test}, y_{test\_pred})$ 
36:   recall  $\leftarrow \text{recall\_score}(y_{test}, y_{test\_pred})$ 
37:   f1  $\leftarrow \text{f1\_score}(y_{test}, y_{test\_pred})$ 
38: return trained model and performance metrics

```

Algorithm A3 Tree-based models training and evaluation.

```

1: Input: Excel file containing questionnaire data
2: Output: Trained models with performance metrics and learning curves
3: // Step 1: Setup and Data Loading
4: Install required libraries: xgboost, lightgbm, catboost (if missing)
5: Load dataset from Excel file and display dimensions
6: // Step 2: Preprocessing and Splitting
7: Drop rows with missing target variable (diabetes)
8:  $X \leftarrow$  numeric features only,  $y \leftarrow$  target (diabetes as integer)
9: Apply SimpleImputer with median strategy to  $X$ 
10: Split data:  $X_{train}, X_{test}, y_{train}, y_{test} \leftarrow$  train_test_split( $X, y$ , test_size=0.2, stratify= $y$ , random_state=42)
11: // Step 3: Define Learning Curve Function
12: Function SHOW_LEARNING_CURVE(model,  $X$ ,  $y$ , title)
13:    $cv \leftarrow$  StratifiedKFold(n_splits=5, shuffle=True, random_state=42)
14:   sizes, train_scores, val_scores  $\leftarrow$  learning_curve(model,  $X$ ,  $y$ ,  $cv=cv$ , train_sizes=linspace(0.1, 1.0, 8))
15:   Plot training and validation curves with title, labels, and legend
16: end Function
17: // Step 4: Model Training and Evaluation Loop
18: Initialize models: [Decision Tree, Random Forest, Extra Trees, XGBoost variants (3), LightGBM, CatBoost]
19: for each model in models do
20:   model.fit( $X_{train}, y_{train}$ )
21:    $cv\_scores \leftarrow$  cross_val_score(model,  $X_{train}, y_{train}$ ,  $cv=$ StratifiedKFold(5))
22:   Print mean CV accuracy  $\pm$  standard deviation
23:   Generate learning curve: show_learning_curve(model,  $X_{train}, y_{train}$ , model_name)
24:    $y_{pred} \leftarrow$  model.predict( $X_{test}$ )
25:   Compute metrics: accuracy, precision, recall, F1-score, ROC-AUC
26:   Generate and display confusion matrix
27:   if model has feature_importances attribute then
28:     Extract and visualize feature importance scores
29:   end if
30: end for
31: return trained models, performance metrics, learning curves

```

Algorithm A4 RUSBoost training and evaluation for prediabetes prediction.

```

1: Input: Dataset  $D$  with features  $X$  and labels  $y$ 
2: Output: Trained RUSBoost model  $M$  and performance metrics
3: // Data Preprocessing
4: Identify categorical columns: EC_stt_1, EC_occp
5: Identify numerical columns: all remaining features
6: Create preprocessing pipeline:
7:   Apply StandardScaler to numerical features
8:   Apply OneHotEncoder to categorical features
9: Transform features:  $X_{processed} \leftarrow preprocess(X)$ 
10: // Data Splitting with Stratification
11: Split data:  $X_{train}, X_{temp}, y_{train}, y_{temp} \leftarrow stratified\_split(X_{processed}, y, ratio=0.70)$ 
12: Split temporary set:  $X_{val}, X_{test}, y_{val}, y_{test} \leftarrow stratified\_split(X_{temp}, y_{temp}, ratio=0.50)$ 
13: // Hyperparameter Search Space Definition
14:  $n\_estimators \leftarrow [100, 200]$ 
15:  $learning\_rate \leftarrow [0.05, 0.1]$ 
16:  $max\_depth \leftarrow [5, 10]$ 
17:  $min\_samples\_leaf \leftarrow [5, 10]$ 
18: // Grid Search with Cross-Validation
19:  $best\_f1 \leftarrow 0$ 
20:  $best\_params \leftarrow \text{None}$ 
21: Initialize StratifiedKFold with  $k = 5$  splits
22: for each combination  $(n, lr, md, ml)$  in parameter grid do
23:    $f1\_scores \leftarrow []$ 
24:   for each fold  $(train\_idx, val\_idx)$  in cross-validation do
25:      $X_{tr}, y_{tr} \leftarrow X_{train}[train\_idx], y_{train}[train\_idx]$ 
26:      $X_{va}, y_{va} \leftarrow X_{train}[val\_idx], y_{train}[val\_idx]$ 
27:     Initialize base estimator:  $tree \leftarrow \text{DecisionTree}(max\_depth = md, min\_samples\_leaf = ml)$ 
28:     Initialize RUSBoost:  $model \leftarrow \text{RUSBoost}(estimator = tree, n\_estimators = n, learning\_rate = lr)$ 
29:     Train model:  $model.fit(X_{tr}, y_{tr})$ 
30:      $predictions \leftarrow model.predict(X_{va})$ 
31:      $f1 \leftarrow \text{compute\_f1\_score}(y_{va}, predictions)$ 
32:     Append  $f1$  to  $f1\_scores$ 
33:   end for
34:    $mean\_f1 \leftarrow \text{average}(f1\_scores)$ 
35:   if  $mean\_f1 > best\_f1$  then
36:      $best\_f1 \leftarrow mean\_f1$ 
37:      $best\_params \leftarrow (n, lr, md, ml)$ 
38:   end if
39: end for
40: // Train Final Model with Optimal Parameters
41:  $(n^*, lr^*, md^*, ml^*) \leftarrow best\_params$ 
42:  $tree^* \leftarrow \text{DecisionTree}(max\_depth = md^*, min\_samples\_leaf = ml^*)$ 
43:  $M \leftarrow \text{RUSBoost}(estimator = tree^*, n\_estimators = n^*, learning\_rate = lr^*)$ 
44:  $M.fit(X_{train}, y_{train})$ 
45: // Model Evaluation
46:  $y_{pred} \leftarrow M.predict(X_{test})$ 
47: Compute performance metrics: accuracy, precision, recall, F1-score
48: Generate confusion matrix
49: Compute weighted feature importance from ensemble
50: return  $M$ , performance metrics

```

References

1. Polonsky, K.S. The Past 200 Years in Diabetes. *N. Engl. J. Med.* **2012**, *367*, 1332–1340. <https://doi.org/10.1056/nejmra1110560>.
2. Himsworth, H. Diabetes Mellitus. *Lancet* **1936**, *227*, 127–130. [https://doi.org/10.1016/s0140-6736\(01\)36134-2](https://doi.org/10.1016/s0140-6736(01)36134-2).

3. Bailey, C.J. Metformin: Historical overview. *Diabetologia* **2017**, *60*, 1566–1576. <https://doi.org/10.1007/s00125-017-4318-z>.
4. Darwish, R.; Abu-Sharia, G.; Butler, A.E. History of glucagon-like peptide-1 receptor agonists. *Pharmacol. Res.* **2025**, *222*, 108045. <https://doi.org/10.1016/j.phrs.2025.108045>.
5. Huvos, A.; Rocha, H. Frequency of Bacteriuria in Patients with Diabetes Mellitus: A Controlled Study. *N. Engl. J. Med.* **1959**, *261*, 1213–1216. <https://doi.org/10.1056/nejm195912102612403>.
6. Piette, J.D.; Kerr, E.A. The Impact of Comorbid Chronic Conditions on Diabetes Care. *Diabetes Care* **2006**, *29*, 725–731. <https://doi.org/10.2337/diacare.29.03.06.dc05-2078>.
7. Benavidez, G.A.; Zahnd, W.E.; Hung, P.; Eberth, J.M. Chronic Disease Prevalence in the US: Sociodemographic and Geographic Variations by Zip Code Tabulation Area. *Prev. Chronic Dis.* **2024**, *21*, 230267. <https://doi.org/10.5888/pcd21.230267>.
8. Aggarwal, R.; Yeh, R.W.; Joynt Maddox, K.E.; Wadhera, R.K. Cardiovascular Risk Factor Prevalence, Treatment, and Control in US Adults Aged 20 to 44 Years, 2009 to March 2020. *JAMA* **2023**, *329*, 899. <https://doi.org/10.1001/jama.2023.2307>.
9. Lee, H.H.; Crawford, E.B.; Cho, S.M.J.; Krawisz, A.K.; Juraschek, S.P.; Cluett, J.L.; Lee, H.; Kim, H.C.; Natarajan, P.; Wadhera, R.K.; et al. Trends in Prevalence, Treatment, and Control of Cardiometabolic Risk Factors Among Adults With Hypertension in the United States, 1999–2023. *JACC* **2025**, *86*, 2577–2593. <https://doi.org/10.1016/j.jacc.2025.09.1607>.
10. Abousaber, I.; Abdallah, H.F.; El-Ghaish, H. Robust predictive framework for diabetes classification using optimized machine learning on imbalanced datasets. *Front. Artif. Intell.* **2025**, *7*, 1499530. <https://doi.org/10.3389/frai.2024.1499530>.
11. Toleva, B.; Atanasov, I.; Ivanov, I.; Hooper, V. An Effective Methodology for Diabetes Prediction in the Case of Class Imbalance. *Bioengineering* **2025**, *12*, 35. <https://doi.org/10.3390/bioengineering12010035>.
12. Almadhoun, M.B.; Burhanuddin, M. Optimizing Feature Selection and Machine Learning Algorithms for Early Detection of Prediabetes Risk: Comparative Study. *JMIR Bioinform. Biotechnol.* **2025**, *6*, e70621. <https://doi.org/10.2196/70621>.
13. Bashar, A.K.M.R.; Goudarzi, M.; Tsokos, C.P. A Machine Learning Classification Model for Detecting Prediabetes. *J. Data Anal. Inf. Process.* **2024**, *12*, 462–478. <https://doi.org/10.4236/jdaip.2024.123024>.
14. Cveticanin, L.; Arsenovic, M. Prediction Models for Diabetes in Children and Adolescents: A Review. *Appl. Sci.* **2025**, *15*, 2906. <https://doi.org/10.3390/app15062906>.
15. Mohd Hamdan, M.D.H.; Ab Jabal, M.F.; Abdul Rahman, S.; Kapi, A.Y. Fairness And Bias Mitigation in AI Models for Diabetes Diagnosis: A Comparative Evaluation of Algorithmic Approaches. *J. Telecommun. Electron. Comput. Eng. (JTEC)* **2025**, *17*, 27–34. <https://doi.org/10.54554/jtec.2025.17.03.004>.
16. Olisah, C.C.; Smith, L.; Smith, M. Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective. *Comput. Methods Programs Biomed.* **2022**, *220*, 106773. <https://doi.org/10.1016/j.cmpb.2022.106773>.
17. Rafie, Z.; Talab, M.S.; Koor, B.E.Z.; Garavand, A.; Salehnasab, C.; Ghaderzadeh, M. Leveraging XGBoost and explainable AI for accurate prediction of type 2 diabetes. *BMC Public Health* **2025**, *25*, 3688. <https://doi.org/10.1186/s12889-025-24953-w>.
18. Sun, Q.; Cheng, X.; Han, K.; Sun, Y.; Ren, H.; Li, P. Machine learning-based assessment of diabetes risk. *Appl. Intell.* **2024**, *55*, 106. <https://doi.org/10.1007/s10489-024-05912-1>.
19. Selvaraj, J.; Jerith, G.G.; R, K.; K, S. Assessment of CatBoost for Diabetes Prevention in Comparison to XGBoost: AI model capable of predicting the onset of diabetes. *EAI Endorsed Trans. Internet Things* **2025**, *11*, 5880. <https://doi.org/10.4108/eetiot.5880>.
20. Dagliati, A.; Marini, S.; Sacchi, L.; Cogni, G.; Teliti, M.; Tibollo, V.; De Cata, P.; Chiovato, L.; Bellazzi, R. Machine Learning Methods to Predict Diabetes Complications. *J. Diabetes Sci. Technol.* **2017**, *12*, 295–302. <https://doi.org/10.1177/1932296817706375>.
21. Makroum, M.A.; Adda, M.; Bouzouane, A.; Ibrahim, H. Machine Learning and Smart Devices for Diabetes Management: Systematic Review. *Sensors* **2022**, *22*, 1843. <https://doi.org/10.3390/s22051843>.
22. Zee, B.; Lee, J.; Lai, M.; Chee, P.; Rafferty, J.; Thomas, R.; Owens, D. Digital solution for detection of undiagnosed diabetes using machine learning-based retinal image analysis. *BMJ Open Diabetes Res. Care* **2022**, *10*, e002914. <https://doi.org/10.1136/bmjdr-2022-002914>.
23. Tang, X.; Kusmartseva, I.; Kulkarni, S.; Posgai, A.; Speier, S.; Schatz, D.A.; Haller, M.J.; Campbell-Thompson, M.; Wasserfall, C.H.; Roep, B.O.; et al. Image-Based Machine Learning Algorithms for Disease Characterization in the Human Type 1 Diabetes Pancreas. *Am. J. Pathol.* **2021**, *191*, 454–462. <https://doi.org/10.1016/j.ajpath.2020.11.010>.
24. Chun, J.W.; Kim, H.S. The Present and Future of Artificial Intelligence-Based Medical Image in Diabetes Mellitus: Focus on Analytical Methods and Limitations of Clinical Use. *J. Korean Med Sci.* **2023**, *38*, e253. <https://doi.org/10.3346/jkms.2023.38.e253>.
25. Nur, A.; Yumnaniha, D.; Tjandra, S.; Bachtar, A.; Harbuwono, D.S. Potential Use and Limitation of Artificial Intelligence to Screen Diabetes Mellitus in Clinical Practice: A Literature Review. *Acta Med. Indones.* **2024**, *56*, 563–570.
26. Sher, M.; Remyes, D.; Sharma, R.; Toma, M. Learning Dynamics Analysis: Assessing Generalization of Machine Learning Models for Optical Coherence Tomography Multiclass Classification. *Informatics* **2025**, *12*, 128. <https://doi.org/10.3390/informatics12040128>.
27. Harris, S.; Bonnici, T.; Keen, T.; Lilaonitkul, W.; White, M.J.; Swanepoel, N. Clinical deployment environments: Five pillars of translational machine learning for health. *Front. Digit. Health* **2022**, *4*, 939292. <https://doi.org/10.3389/fdgh.2022.939292>.

28. Yang, Y.; Liao, C.Y.; Keyvanshokoo, E.; Shao, H.; Weber, M.B.; Pasquel, F.J.; Garcia, G.G.P. A Responsible Framework for Assessing, Selecting, and Explaining Machine Learning Models in Cardiovascular Disease Outcomes Among People With Type 2 Diabetes: Methodology and Validation Study. *JMIR Med. Inform.* **2025**, *13*, e66200. <https://doi.org/10.2196/66200>.
29. Grinsztajn, L.; Oyallon, E.; Varoquaux, G. Why Do Tree-Based Models Still Outperform Deep Learning on Typical Tabular Data? In *Proceedings of the Advances in Neural Information Processing Systems 35, New Orleans, LA, USA, 28 November–9 December 2022*; NeurIPS 2022; Neural Information Processing Systems Foundation, Inc. (NeurIPS): La Jolla, CA, USA, 2022; pp. 507–520. <https://doi.org/10.52202/068431-0037>.
30. Shwartz-Ziv, R.; Armon, A. Tabular data: Deep learning is not all you need. *Inf. Fusion* **2022**, *81*, 84–90. <https://doi.org/10.1016/j.inffus.2021.11.011>.
31. Kulkarni, S.; Hire, D.; Charles, P.; Suryawanshi, S. A Two-Stage Ensemble Approach for Diabetes Prediction: Early Diagnosis with CatBoost and Advanced Diagnosis with LightGBM. *Biomed. Pharmacol. J.* **2026**, *19*, 2. Available online: <https://bit.ly/4cxeiT2> (accessed on 22 June 2026).
32. Yoo, D.; Maggiore, U.; Jolliet, O. Enhancing prediabetes and diabetes detection through a machine learning-enabled self-assessment approach. *J. Clin. Epidemiol.* **2026**, *195*, 112266. <https://doi.org/10.1016/j.jclinepi.2026.112266>.
33. Netayawijit, P.; Chansanam, W.; Sorn-In, K. Interpretable Machine Learning Framework for Diabetes Prediction: Integrating SMOTE Balancing with SHAP Explainability for Clinical Decision Support. *Healthcare* **2025**, *13*, 2588. <https://doi.org/10.3390/healthcare13202588>.
34. Nguyen, H.V.; Choi, Y.; Byeon, H. An explainable hybrid deep learning model for prediabetes prediction in men aged 30 and above. *J. Men's Health* **2024**, *20*, 52–72. <https://doi.org/10.22514/jomh.2024.166>.
35. Toma, M. *AI-Assisted Medical Diagnostics: A Clinical Guide to Next-Generation Diagnostics*; Dawning Research Press: New York, NY, USA, 2025; ISBN: 979-8-9998324-3-6. Available online: <https://openlibrary.org/works/OL44048041W/> (accessed on 22 June 2026).
36. Toma, M. *Diagnosing AI: Evaluation of AI in Clinical Practice*; Dawning Research Press: New York, NY, USA, 2026; ISBN: 979-8-9998324-8-1. Available online: <https://openlibrary.org/works/OL61375300M/> (accessed on 22 June 2026).
37. Kim, J.h. *Prediabetes and Health*; Yonsei University-Mirae Campus: Wonju, Republic of Korea, 2025. <https://doi.org/10.17632/7XB9DHVBCN.1>.
38. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. <https://doi.org/10.1038/nature14539>.
39. Arslan, A.K.; Yagin, F.H.; Algarni, A.; Karaaslan, E.; Al-Hashem, F.; Ardigo, L.P. Enhancing type 2 diabetes mellitus prediction by integrating metabolomics and tree-based boosting approaches. *Front. Endocrinol.* **2024**, *15*, 1444282. <https://doi.org/10.3389/fendo.2024.1444282>.
40. Jose, R.; Syed, F.; Thomas, A.; Toma, M. Cardiovascular Health Management in Diabetic Patients with Machine-Learning-Driven Predictions and Interventions. *Appl. Sci.* **2024**, *14*, 2132. <https://doi.org/10.3390/app14052132>.
41. Arik, S.O.; Pfister, T. TabNet: Attentive Interpretable Tabular Learning. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 6679–6687. <https://doi.org/10.1609/aaai.v35i8.16826>.
42. Mustafa, F.; Al-Hussainy, A.; Doshi, H.; Yadav, A.; Rekha, M.; Kundlas, M.; Sabarivani, A.; Kubaev, A.; Taher, S.; Alwan, M.; et al. TabNet and TabTransformer: Novel Deep Learning Models for Chemical Toxicity Prediction in Comparison With Machine Learning. *J. Appl. Toxicol.* **2025**, *45*, 1730–1749. <https://doi.org/10.1002/jat.4803>.
43. Jiang, W.; Liu, A.; Zhang, Y.; Han, H.; Mu, J.; Liu, S.; Gu, W.; Huang, S. Coverage Prediction in Mobile Communication Networks: A Deep Learning Approach With a Tabular Foundation Model. *Internet Technol. Lett.* **2025**, *8*, e70034. <https://doi.org/10.1002/itl2.70034>.
44. Ebbelohj, A.; Thunbo, M.O.; Andersen, O.E.; Glindtvad, M.V.; Hulman, A. Transfer learning for non-image data in clinical research: A scoping review. *PLoS Digit. Health* **2022**, *1*, e0000014. <https://doi.org/10.1371/journal.pdig.0000014>.
45. Kalaycioglu, O.; Pavlou, M.; Akhanli, S.E.; de Belder, M.A.; Ambler, G.; Omar, R.Z. Evaluating the sample size requirements of tree-based ensemble machine learning techniques for clinical risk prediction. *Stat. Methods Med. Res.* **2025**, *34*, 1356–1372. <https://doi.org/10.1177/09622802251338983>.
46. Adler, A.I.; Painsky, A. Feature Importance in Gradient Boosting Trees with Cross-Validation Feature Selection. *Entropy* **2022**, *24*, 687. <https://doi.org/10.3390/e24050687>.
47. Shmuel, A.; Glickman, O.; Lazebnik, T. A comprehensive benchmark of machine and deep learning models on structured data for regression and classification. *Neurocomputing* **2025**, *655*, 131337. <https://doi.org/10.1016/j.neucom.2025.131337>.
48. Borisov, V.; Leemann, T.; Seßler, K.; Haug, J.; Pawelczyk, M.; Kasneci, G. Deep Neural Networks and Tabular Data: A Survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 7499–7519. <https://doi.org/10.1109/tnnls.2022.3229161>.
49. 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*; KDD'16; Association for Computing Machinery: New York, NY, USA, 2016; pp. 785–794. <https://doi.org/10.1145/2939672.2939785>.

50. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017*; NIPS'17; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 3149–3157.
51. Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A.V.; Gulin, A. CatBoost: Unbiased boosting with categorical features. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018*; NIPS'18; Curran Associates Inc.: Red Hook, NY, USA, 2018; pp. 6639–6649.
52. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. <https://doi.org/10.1613/jair.953>.
53. Husain, G.; Nasef, D.; Jose, R.; Mayer, J.; Bekbolatova, M.; Devine, T.; Toma, M. SMOTE vs. SMOTEENN: A Study on the Performance of Resampling Algorithms for Addressing Class Imbalance in Regression Models. *Algorithms* **2025**, *18*, 37. <https://doi.org/10.3390/a18010037>.
54. Fernandez, A.; Garcia, S.; Herrera, F.; Chawla, N.V. SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *J. Artif. Intell. Res.* **2018**, *61*, 863–905. <https://doi.org/10.1613/jair.1.11192>.
55. Yildiz, A.Y.; Kalayci, A. Gradient Boosting Decision Trees on Medical Diagnosis Over Tabular Data. In *Proceedings of the 2025 IEEE International Conference on AI and Data Analytics (ICAD), Medford, MA, USA, 24 June 2025*; IEEE: New York, NY, USA, 2025; pp. 1–8. <https://doi.org/10.1109/icad65464.2025.11114069>.
56. Grinsztajn, L.; Holzmüller, D.; Steinwart, I. Better by default: Strong pre-tuned MLPs and boosted trees on tabular data. In *Proceedings of the Advances in Neural Information Processing Systems 37, New Orleans, LA, USA, 10–16 December 2023*; NeurIPS 2024; Neural Information Processing Systems Foundation, Inc. (NeurIPS): La Jolla, CA, USA, 2024; pp. 26577–26658. <https://doi.org/10.52202/079017-0837>.
57. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **2006**, *63*, 3–42. <https://doi.org/10.1007/s10994-006-6226-1>.
58. He, J.; Li, Z.; Yin, L. An Efficient and Accurate Random Forest Node-Splitting Algorithm Based on Dynamic Bayesian Methods. *Mach. Learn. Knowl. Extr.* **2025**, *7*, 70. <https://doi.org/10.3390/make7030070>.
59. Seiffert, C.; Khoshgoftaar, T.M.; Van Hulse, J.; Napolitano, A. RUSBoost: A Hybrid Approach to Alleviating Class Imbalance. *IEEE Trans. Syst. Man Cybern. Part A Syst. Humans* **2010**, *40*, 185–197. <https://doi.org/10.1109/tsmca.2009.2029559>.
60. Seiffert, C.; Khoshgoftaar, T.M.; Van Hulse, J.; Napolitano, A. RUSBoost: Improving classification performance when training data is skewed. In *Proceedings of the 2008 19th International Conference on Pattern Recognition, Tampa, FL, USA, 8–11 December 2008*; IEEE: New York, NY, USA, 2008; pp. 1–4. <https://doi.org/10.1109/icpr.2008.4761297>.
61. U.S. Food and Drug Administration. Considerations for the Use of Artificial Intelligence to Support Regulatory Decision-Making for Drug and Biological Products. Draft Guidance for Industry, U.S. Department of Health and Human Services, Food and Drug Administration. 2025. Available online: <https://www.fda.gov/media/184830/download> (accessed 30 June 2026).
62. Toma, M. From Interpretable Models to Clinical Implementation: Advances in AI-Assisted Medical Diagnostics. *Algorithms* **2026**, *19*, 330. <https://doi.org/10.3390/a19050330>.
63. Nasef, D.; Nasef, D.; Sher, M.; Toma, M. A Standardized Validation Framework for Clinically Actionable Healthcare Machine Learning with Knee Osteoarthritis Grading as a Case Study. *Algorithms* **2025**, *18*, 343. <https://doi.org/10.3390/a18060343>.
64. Balendran, A.; Beji, C.; Bouvier, F.; Khalifa, O.; Evgeniou, T.; Ravaut, P.; Porcher, R. A scoping review of robustness concepts for machine learning in healthcare. *npj Digit. Med.* **2025**, *8*, 38. <https://doi.org/10.1038/s41746-024-01420-1>.
65. Thirunavukarasu, A.J.; Elangovan, K.; Gutierrez, L.; Hassan, R.; Li, Y.; Tan, T.F.; Cheng, H.; Teo, Z.L.; Lim, G.; Ting, D.S.W. Clinical performance of automated machine learning: A systematic review. *Ann. Acad. Med. Singap.* **2024**, *53*, 187–207. <https://doi.org/10.47102/annals-acadmedsg.2023113>.
66. Zafar, M.M.; Khan, Z.A.; Javaid, N.; Aslam, M.; Alrajeh, N. From Data to Diagnosis: A Novel Deep Learning Model for Early and Accurate Diabetes Prediction. *Healthcare* **2025**, *13*, 2138. <https://doi.org/10.3390/healthcare13172138>.
67. Scott, I.A.; De Guzman, K.R.; Falconer, N.; Canaris, S.; Bonilla, O.; McPhail, S.M.; Marxen, S.; Van Garderen, A.; Abdel-Hafez, A.; Barras, M. Evaluating automated machine learning platforms for use in healthcare. *JAMIA Open* **2024**, *7*, ooae031. <https://doi.org/10.1093/jamiaopen/ooae031>.

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.