MDPI

*Article*

# Comparison of Profit-Based Multi-Objective Approaches for Feature Selection in Credit Scoring

**Naomi Simumba \*, Suguru Okami, Akira Kodaka and Naohiko Kohtake**

Graduate School of System Design and Management, Keio University, Yokohama 223-8526, Japan;
yiu68252@nifty.com (S.O.); akira.kodaka@sdm.keio.ac.jp (A.K.); kohtake@sdm.keio.ac.jp (N.K.)
\* Correspondence: naomi.simumba@keio.jp

**Abstract:** Feature selection is crucial to the credit-scoring process, allowing for the removal of irrelevant variables with low predictive power. Conventional credit-scoring techniques treat this as a separate process wherein features are selected based on improving a single statistical measure, such as accuracy; however, recent research has focused on meaningful business parameters such as profit. More than one factor may be important to the selection process, making multi-objective optimization methods a necessity. However, the comparative performance of multi-objective methods has been known to vary depending on the test problem and specific implementation. This research employed a recent hybrid non-dominated sorting binary Grasshopper Optimization Algorithm and compared its performance on multi-objective feature selection for credit scoring to that of two popular benchmark algorithms in this space. Further comparison is made to determine the impact of changing the profit-maximizing base classifiers on algorithm performance. Experiments demonstrate that, of the base classifiers used, the neural network classifier improved the profit-based measure and minimized the mean number of features in the population the most. Additionally, the NSBGOA algorithm gave relatively smaller hypervolumes and increased computational time across all base classifiers, while giving the highest mean objective values for the solutions. It is clear that the base classifier has a significant impact on the results of multi-objective optimization. Therefore, careful consideration should be made of the base classifier to use in the scenarios.

**Keywords:** multi-objective optimization; profit scoring; feature selection; credit evaluation

## 1. Introduction

Credit-scoring evaluations are an important part of the lending process, allowing financial institutions to manage risks [1]. Feature selection, a crucial part of the credit-scoring process, typically aims to minimize the number of features, thereby reducing model complexity, data acquisition costs, and computation time [2]. Traditionally, feature selection is conducted as a separate step before model training and is used to improve a single statistical measure, such as the area under the receiver operating curve (AUC) [3]. Beyond this, other factors, such as the profitability of the resulting model [4,5], have been the focus of the feature-selection process. These factors, which depend on data and applications, can be incorporated into the feature-selection process as objectives in multi-objective optimizations (MOOs).

MOO algorithms allow designers to balance several, often conflicting, objectives [6]. These methods have been applied to simultaneously consider the number of features and another training objective, such as profit, in feature selection [7]. Several algorithms have been developed to handle MOO problems, including the Strength Pareto Evolutionary Algorithm (SPEA-II), non-dominated sorting genetic algorithm (NSGA-II) [8,9], and its reference-based adaptation for many-objective problems, NSGA-III. Hybrid algorithms, which integrate aspects of two or more optimization methods, have also been employed.

An example is the adaptation of the continuous Grasshopper Optimization Algorithm (GOA) for filter-based feature selection through the introduction of binary conversion [10,11]. Further examples used non-dominated strategies to convert Cuckoo Optimization Algorithm (COA) [12]. A non-dominated sorting binary GOA, NSBGOA, was proposed for feature selection with optimization of multiple objective [13].

Existing research has shown that even for closely related multi-objective algorithms, performance varies depending on the test problem [14]. However, there is limited research comparing performance of different multi-objective algorithms for feature selection. In particular, existing research tends to use on one base classifier, with the base classifier used depending on the analyst's discretion. Changes in performance due to different base classifiers require further examination. This research aims to fill the gap by comparing the performance of several multi-objective methods on feature selection in credit scoring, namely NSGA-II, NSGA-III, and the newly proposed hybrid meta-heuristic, NSBGOA. Secondly, the effect of different base classifiers on performance is considered to determine the most suitable. Third, these multi-objective methods are compared to conventional feature-selection techniques. Three common objectives for credit-scoring feature selection are employed: maximizing profit, selecting features that are more easily explained to stakeholders, and minimizing number of features [4,15].

Related research is considered in Section 2, while Section 3 introduces the methods used in this research. Detailed in Section 4 is the problem formulation and empirical evaluation. Results of the evaluation are shown in Section 5 and discussed in Section 6. Lastly, the conclusion is given in Section 7.

## 2. Related Work

### 2.1. Profit Scoring

Credit scoring is defined as "…a set of decision models and their underlying techniques that aid lenders in granting consumer credit" [1]. Its core purpose is to assess the risk of lending to a prospective borrower. Historically, credit decisions were based the lender's knowledge of the borrower. In modern times, statistical and machine learning approaches have taken precedence. The goal of these approaches to credit scoring is to distinguish borrowers who are likely to show some negative behavior. Recent research has trended towards evaluation of profit as part of the credit-scoring process because it allows for improved decision-making by lenders. A profit measure comprised of benefits versus losses due to misclassification was proposed, with varying data variable acquisition costs also being considered [5]. The Internal Rate of Return (IRR) was used to measure profitability of peer-to-peer loans [16]. A new measure, expected maximum profit (EMP), which is composed of the benefits of correct classification and costs of misclassification, was suggested [17] and reworked for consumer credit scoring [18].

### 2.2. Feature Selection

The selection of input features is an important part of model building. Typically, the process aims to ensure optimum model performance with minimum features. This reduces noise, data costs, and the risk of overfitting. Feature-selection methods are generally classified into wrapper, filter, and embedded methods [2]. With wrapper methods, models are fit with subsets of the features, and the resulting model performance is evaluated. However, due to the high computational cost, they are difficult to run on datasets with a large number of features. Examples include backward and forward selection. For filter methods, features are selected based on inherent properties, such as variance. Analysis of variance (ANOVA) is an example of this [19]. Finally, embedded methods, such as LASSO [20] and ridge regression, perform feature selection and model fitting simultaneously.

Feature selection has been conducted by using support vector machines based on EMP in an embedded method [4]. A profit-based measure was applied with a Holdout Support Vector Machine (HOSVM) to extract the features with highest profitability [5].

Feature selection was conducted by using mixed-integer linear programming models with varying acquisition costs as constraints [15]. The orthogonal transform was used for dimensionality reduction, thus reducing the number of features for model training, leading to faster convergence and better performance [21]. Feature selection is carried out by integrating a multicriteria optimization classifier (MCOC) with a one-norm regularization term inspired by the LASSO regression method to create a sparse feature vector [22].

Multiple objectives have also been optimized in feature selection through multi-objective feature analysis. Existing literature has several examples of optimizing the feature-selection process with two objectives. A non-dominated sorting genetic algorithm-II (NSGA-II) fitted to maximize the expected maximum profit (EMP) and minimize the number of features was demonstrated [7]. Mutual information and entropy were optimized for filter-based feature selection with a non-dominated sorting binary Particle Swarm Optimization (NSBPSO) [23]. Binary Grasshopper Optimization Algorithms were applied for filter-based feature selection based on error rate and number of features in References [10,11]. A wrapper based multi-objective evolutionary algorithm optimized feature selection with three objectives: default prediction, exposure at default, and number of features [24]. Two objectives, number of features and root mean square error (RMSE), were optimized in feature selection with multi-objective genetic algorithm and neurofuzzy models [25].

*2.3. Multi-Objective Optimization*

Multi-objective evolutionary algorithms (MOEAs), including Strength Pareto Evolutionary Algorithm (SEPA-II) and binary non-dominated sorting genetic algorithm (NSGA-II), have been applied for problems with two objectives [8,9]. Non-dominated sorting has also been incorporated with meta-heuristic optimizers for multi-objective problems. For instance, the Particle Swarm Optimizer [26] and the Ant Colony optimizer [27] have both been adapted to multi-objective problems. A binary version of the Grasshopper Optimization Algorithm has been developed for feature selection [11].

For many-objective optimizations (MaOP), which typically involve three or more objectives, their performance degrades with the increase in objectives due in part to the large number of mutually non dominated solutions [28]. As such, indicator, aggregation, and reference-based methods have been proposed to tackle MaOps. Examples include NSGA-III, a reference-based extension of NSGA-II for many-objective problems [29]. When compared on several test problems with different numbers of objectives, it was determined that the NSGA-III does not always outperform NSGA-II. In fact, the performance is affected by the number of objectives and the specific test problem evaluated [14].

**3. Methods**

*3.1. Multi-Objective Optimization*

The Pareto-optimal set is a non-dominated set of solutions which allows decision-makers to find a trade-off where more than one objective is involved [30]. Multi-objective optimization (MOO) methods guide the search for solutions towards the Pareto-optimal set. MOOs are especially important for feature selection where more than one objective must be considered. For instance, maximizing profit while minimizing number of features. Mathematically, multi-objective optimization problems may be expressed by Equations (1)–(3) below:

$$minimize\ F(x) = \left( f_1(x), f_2(x), \ldots, f_m(x) \right)^T \tag{1}$$

$$where\ vector\ X = ( x_1, x_2, \ldots, x_n) \tag{2}$$

$$and\ vector\ X \in \Omega \tag{3}$$

The objective functions vector $F(x)$ maps $F: \Omega \to \Lambda$. Here the decision space and vector are $\Omega$ and $X$, respectively. For many-objective optimization problems (MaOP), m ≥ 3.

### 3.2. Non-Dominated Genetic Algorithm (NSGA-II)

NSGA-II is a popular multi-objective optimization algorithm that can applied to feature-selection problems [8]. The algorithm begins by evaluating the fitness of the initial population of potential solutions. From these, "parent solutions" are selected and crossed to generate "child solutions". Mutation may occur where some components of the solutions are randomly altered. If the stopping criteria are not met, a non-dominated sorting scheme is used to select the best solutions, with diversity maintained by calculating and maximizing the crowding distance between solutions. The result of this sorting process becomes the population for the next round of evaluation. In this manner, the population converges towards the set of overall best, non-dominated solutions known as the Pareto frontier. NSGA-II can be adapted to feature selection by denoting the individuals as different feature combinations and setting the number of features as an objective in addition to the main objective such as profit. The process is given in Algorithm 1.

**Algorithm 1.** Pseudo code for NSGA-II.

| Algorithm for NSGA-II |
| --- |
| Initialize population, $P_t$, of size N |
| $Q_0 = \varnothing$ |
| $F_0$ = Fitness evaluation of $P_0$ |
| ($F_1$, $F_2$, …) = Non dominated sorting of $P_0$ to establish rank |
| Determine crowding distance of ($F_1$, $F_2$, …) |
| **while** stop criterion not satisfied **do** |
| $\quad$ $Q_t$ = selection, crossover, mutation, recombination of $P_t$ |
| $\quad$ $R_t = Q_t \cup P_t$ |
| $\quad$ $F_t$ = Fitness_Evaluation($R_t$) |
| $\quad$ ($F_1$, $F_2$, …) = Non dominated sorting of $R_t$ to establish rank |
| $\quad$ Determine crowding distance |
| $\quad$ $P_{t+1}$ = select new population of size N based on rank and highest crowding distance |
| **end while** |

### 3.3. Non-Dominated Genetic Algorithm (NSGA-III)

Deb and Jain [29] proposed the NSGA-III algorithm as an extension of NSGA-II to deal with MaOPs. NSGA-III generates a reference set from virtual points in the objective space to measure the quality of solutions. A population, composed of potential solutions, is initialized. As with genetic algorithms, the fitness of the solutions is assessed by computing the fitness or objective functions. So-called "parent" solutions are selected and crossed over to obtain "child" solutions.

Mutations may also occur where components of the solutions may be altered. The population is normalized and associated to the reference set by the orthogonal distance to reference lines. During selection, rather than preserving diversity by using the crowding distance, as is the case with NSGA-II, NSGA-III uses niche-preservation based on the reference set [31]. This process is shown in Algorithm 2.

**Algorithm 2.** Pseudo code for NSGA-III.

| Algorithm for NSGA-III |
| --- |
| Initialize reference points |
| Initialize population, $P_0$, of size N |
| **while** stop criterion not satisfied **do** |
| $\quad$ $S_t = \emptyset$, i =1 |
| $\quad$ $Q_t$ = selection, crossover, mutation, recombination of $P_t$ |
| $\quad$ $R_t = Q_t \cup P_t$ |
| $\quad$ $(F_1, F_2, \ldots)$ = Non dominated sorting $(R_t)$ |
| $\quad$ **while** $|S_t| < N$ **do** |
| $\quad\quad$ $S_t = S_t \cup F_i$ |
| $\quad\quad$ i = i + 1 |
| $\quad$ **end while** |
| $\quad$ $F_l = F_i$ ($F_l$ last front included) |
| $\quad$ **if** $|St| = N$ **then** |
| $\quad\quad$ $P_{t+1} = S_t$ |
| $\quad$ **else** |
| $\quad\quad\quad\quad$ $P_{t+1} = \cup_{j=1}^{l-1} F_j$ |
| $\quad\quad$ Individuals to be chosen K = N − $|$ $P_{t+1}$ $|$: |
| $\quad\quad$ Normalize objectives |
| $\quad\quad$ Associate each member s of $S_t$ with a reference point |
| $\quad\quad$ Compute niche count of reference point |
| $\quad\quad$ Choose K members chosen one at a time from $F_l$ to obtain $P_{t+1}$ |
| $\quad$ **end if** |
| **end while** |

*3.4. Non-Dominated Binary Grasshopper Optimization Algorithm (NSBGOA)*

NSBGOA is a hybrid meta-heuristic proposed to handle multi-objective feature selection [13]. It adapts the Grasshopper Optimization Algorithm (GOA) [32], a swarm intelligence based optimizer that models the behavior of grasshoppers in a swarm. Each individual grasshopper's position is a possible solution, and the velocity of the individual grasshoppers as they attempt to swarm into the so-called "comfort zone" is updated with each iteration. This velocity is a function of their social interaction and movement towards the target.

For feature selection, the continuous GOA algorithm is converted to a binary GOA by introducing the sigmoidal transfer function of Equation (4) [10]. The velocity, $\Delta X$, is adapted to Equation (5), where $d_{ij}$ is the distance between two grasshoppers, the function $s$ is the strength of the social forces, parameter $c$ decreases the comfort zone with each iteration, and $ub_d$ and $ib_d$ are the upper bound and lower bound in the $d$th dimension. Equation (6) gives the $d$th dimension of a grasshopper in the next iteration. Additionally, non-dominated sorting is integrated into the algorithm to allow for comparison of multiple objectives. This results in the algorithm of Algorithm 3.

$$T(\Delta X_t) = \frac{1}{1 + e^{-\Delta X_t}} \tag{4}$$

$$\Delta X = c_1 \left( \sum_{\substack{j=1 \\ j \neq i}}^{N} c_2 \frac{ub_d - lb_d}{2} \, s\left(\left|x_j^d - x_i^d\right|\right) \frac{x_j - x_i}{d_{ij}} \right) \tag{5}$$

$$X_{t+1}^d = \begin{cases} 1 & \text{if } r < T(\Delta X_{t+1}^d) \\ 0 & \text{if } r \geq T(\Delta X_{t+1}^d) \end{cases} \tag{6}$$

**Algorithm 3.** Pseudo code for NSBGOA.

---

Algorithm for NSBGOA

---

**Input**:
- Population size, N
- maxIter

**Steps**

T = 0

$G_t$ = Initialize grasshopper positions

$P_t = G_t$

$F_t$ = Fitness wvaluation of population, $P_t$

bestFront$_t$, fitnessOfBestFront$_t$ = non-dominated sorting of $P_t$ and $F_t$

**while** t < maxIter **do**:

    Update c

    **for** grasshopper **in** $G_t$

        normalize the distances between grasshoppers in References [1,4]

        $G_t$ = compute and update new grasshopper positions

    $P_t = G_t$

    $F_t$ = evaluate fitness of population, $P_t$

    bestFront$_t$, fitnessOfBestFront$_t$ = non-dominated sorting of $P_t$ and $F_t$

    $Z_t$ = Sort $P_t$ by position on fronts

    $G_{t+1} = \varnothing$

    i =1

    **while** i <=N **do:**

        $G_{t+1} = G_{t+1} \cup$ ith member of $P_t$

        i = i + 1

    **end while**

    bestFront$_{t+1}$, fitnessOfBestFront$_{t+1}$ = non-dominated sorting of $G_{t+1}$ and $F_t$

    t = t +1

**end while**

S = bestFront

**Output**
- set of non-dominated solutions, S

---

### 3.5. Expected Maximum Profit (EMP)

Expected maximum profit (EMP), the maximum profit obtainable, is a profit-based metric [18] that is applicable to credit scoring. Four potential classification outcomes exist, as per Table 1. When a good borrower is rejected, the lender loses the return on investment (ROI). Additionally, accepting a bad borrower results in loss of the benefit, $b \in [0,1]$, expressed by the Equation (7), with exposure at default (EAD), loss given default (LGD), and principal amount (A) of the loan [33].

$$b = \frac{LGD * EAD}{A} \tag{7}$$

**Table 1.** Confusion matrix for Expected Maximum Profit.

| | | Predicted Class | |
|---|---|---|---|
| | | **Non-Default** | **Default** |
| **Actual** | **Non-default** | Benefit: ROI | Cost: -ROI |
| | | Probability: $\pi_1 (1 - F_1)$ | Probability: $\pi_1 F_1$ |
| | **Default** | Cost: -LGD × EAD/A | Benefit: LGD × EAD/A |
| | | Probability: $\pi_0 (1 - F_0)$ | Probability: $\pi_0 F_0$ |

The benefit, $b$, depends on how much of the loan is repaid in full.

- $b = 0$ with probability $p_0$ that the loan is repaid in full,
- $b = 1$ with probability $p_1$ that no portion of the loan is repaid,
- $b$ is uniformly distributed within (0,1) with $g(b) = 1 - p_0 - p_1$.

Finally, EMP is given by equation 8 below with prior probabilities of default ($\pi_0$), prior probabilities non-default ($\pi_1$), predicted cumulative density functions ($F_0$ and $F_1$), and constant ROI.

$$EMP = \int_0^1 (b * \pi_0 F_0 - ROI * \pi_1 F_1) * g(b) \, \mathrm{d}b \tag{8}$$

### 3.6. Performance Metrics

To evaluate the output of many- and multi-objective optimizations, the hypervolume indicator (HV) may be used. HV, which can be used to evaluate convergence and distribution, is denoted by Equation (9), where $\lambda_m$ is the m-dimensional Lebesgue measure and $m$ is the number of objectives [34]. It calculates the volume of objective space dominated by the Pareto Front approximation, $P$, and delimited from above by the reference point $r$ such that $z \in P$, $z$ dominates $r$.

$$HV(P, r) = \lambda_m \left( \bigcup_{z \in P} [z; r] \right) \tag{9}$$

## 4. Empirical Evaluation

### 4.1. Problem Formulation

With credit-scoring data identified, objectives may be listed. To achieve this, the following definitions are given:

1. *Available features X,* (Equation (10)) a set of *j* variables that could be used to predict loan repayment.
2. *Cardinality (number of features), N,* (Equation (11)) is the number of selected features per solution.
3. *Expected Maximum Profit, EMP,* (Equation (8)) a profit-based measure for credit scoring.
4. *Ease of explanation, C,* (Equation (12)) a vector representing the ease of explaining each variable to stakeholders.

5.  *Default status, D,* (Equation (13)) is a vector with loan repayment information for each borrower.
6.  *Borrower information, B,* (Equation (14)) a matrix with feature values for each borrower.

$$Available\ Features, X = \{x_1, x_2, x_3, \dots x_j\} \tag{10}$$

$$Cardinality, \quad N = \{n_1, n_2, n_3, \dots\} \tag{11}$$

$$Ease\ of\ explanation, = \{c_1, c_2, c_3, \dots\} \tag{12}$$

$$Default\ status, D = \{d_1, d_2, d_3,, \dots d_i\} \tag{13}$$

$$Borrower\ Information, B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1j} \\ b_{21} & b_{22} & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & \cdots & \cdots & \ddots \end{bmatrix} \tag{14}$$

$$possible\ solution, P = \{x_1, x_2, x_3, \dots x_n\} \tag{15}$$

Each solution, $P$ (Equation (15)) is comprised of $n$ features such that $P \subset X$. The goal is to select a set, $S = \{s_1, s_2, s_3,, \dots s_p\}$ so that each element is a distinct subset of $X$ resulting in non-dominated objective values.

### 4.2. Contribution

NSGA-II was used for feature selection in credit scoring with two objectives, profit and number of features [7], and different base classifiers were applied. However, its performance on different base classifiers was not compared to that of other multi-objective methods. Feature selection was conducted by introducing data-acquisition costs as constraints to a support vector machine (SVM) classifier for credit scoring [15]. Although the performance of NSGA-II and NSGA-III on different problems was compared, these comparisons were limited to test problems, not feature selection. Interestingly, it was found that their performance varied depending on the test problem and number of objectives. This leaves open the question of which multi-objective method and base classifier is most suitable for credit-scoring problems. Furthermore, the performance of the NSBGOA algorithm with different base classifiers is still in question. This research aims to answer these open research questions.

### 4.3. Data and Objectives

To test the various algorithms, the German credit dataset [35], which contains 1000 entries with 700 being non-default and 300 being default. There are 20 features in the initial dataset: 13 qualitative and 7 numerical. For the purposes of this evaluation, the variables that were judged to have ambiguous definitions were given ease of explanation values of 0, namely V2, V3, V12, V14, V17, and V18. Meanwhile, the remaining variables were assigned values of 1. The targets of the optimization are given in Table 2.

**Table 2.** Objectives.

| Objective | Optimization | Function |
|---|---|---|
| Expected maximum profit (EMP) | maximize | $EMP = \int_0^1 (b * \pi0\ F0 - ROI * \pi1\ F1) * \mathrm{g}(b)\ \mathrm{d}b$ |
| Number of features (cardinality) | maximize | $-n/j$ |
| Ease of explanation per feature set | maximize | $c = \sum_{p=1}^{j} individual\ variable\ value_p / j$ |

*4.4. Analysis*

Numeric variables went through max–min rescaling, and class imbalance was achieved with the random over-sampling examples technique (ROSE). An assortment of packages in CRAN R was used to conduct analysis. The NSGA-III and NSGA-II algorithms were implemented with the "*rmoo*" package. The analysis of NSBGOA was implemented with an appropriately modified version of the GOA function from the "*metaheuristicOpt*" package. To calculate the objective values for each subset, three classifiers from the "*caret*" package were trained by using tenfold cross-validation, and the best model was selected based on the EMP ("*EMP*" package). The parameters for EMP evaluation were set to $p_0 = 0.55$, $p_1 = 0.1$, and ROI = 0.2644, as proposed by the authors in Reference [18]. The three base classifiers were regression (LR), support vector machine with a linear kernel (SVM LIN), and artificial neural networks (NN). Following this, the values of the remaining objectives were calculated according to Table 2. For comparison, a LASSO [20] regression model with alpha = 1 and lambda = 0 was fit on the data, using tenfold cross-validation to maximize EMP. Additionally, a single-objective genetic algorithm (GA) [36] was used for feature selection with the fitness function being a tenfold cross-validation classifier trained to maximize EMP. Lastly, the three base classifiers were also trained on all the original features to maximize EMP. Population size of 10 and maximum iteration number of 50 was used for NSGA-III, NSBGOA, NSGA-II, and GA methods.

**5. Results**

The mean hypervolumes of the final populations from five runs of the NSGA-III, NSGA-II, and NSBGOA algorithms are given in Figure 1. Further, Figure 2 shows the mean computational time from five runs of each algorithm on a computer with 8GB RAM and an Intel(R) Core(TM) i5-7200U CPU 2.50GHz, while the objective values of the feature set with max EMP are shown in Figure 3. For a single run of each algorithm, the distribution of points in the outputs of the multi-objective algorithms is shown with scatter plots in Figure 4. For the same run, a statistical summary of the final populations is given in Table 3, and the three algorithms trained to optimize a single objective have their results given in Table 4. The nadir point, which describes the solution set, is also given.

**Table 3.** Statistical summary of multi- and many-objective results.

| Algorithm | Mean # of Features/Solution | Objective | Min | Max | SD | Mean | Nadir Point |
|---|---|---|---|---|---|---|---|
| NSGA-II with Logistic Regression | 11.8 | emp | 0.083 | 0.093 | 0.087 | 0.087 | 0.093 |
| | | cardinality | −0.8 | −0.4 | −0.59 | −0.59 | −0.4 |
| | | ease of explanation | 0.15 | 0.5 | 0.31 | 0.31 | 0.5 |
| NSGA-II with Neural Network | 9.1 | emp | 0.084 | 0.096 | 0.09 | 0.09 | 0.096 |
| | | cardinality | −0.65 | −0.2 | −0.455 | −0.455 | −0.2 |
| | | ease of explanation | 0.15 | 0.4 | 0.245 | 0.245 | 0.4 |
| NSGA-II with Linear Support Vector Machine | 12 | emp | 0.082 | 0.095 | 0.088 | 0.088 | 0.095 |
| | | cardinality | −0.8 | −0.3 | −0.6 | −0.6 | −0.3 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | ease of explanation | 0.15 | 0.5 | 0.34 | 0.34 | 0.5 |
| NSGA-III with Logistic Regression | 12.4 | emp | 0.084 | 0.09 | 0.087 | 0.087 | 0.09 |
| | | cardinality | −0.8 | −0.5 | −0.62 | −0.62 | −0.5 |
| | | ease of explanation | 0.2 | 0.5 | 0.335 | 0.335 | 0.5 |
| NSGA-III with Neural Network | 11.5 | emp | 0.09 | 0.104 | 0.093 | 0.093 | 0.104 |
| | | cardinality | −0.8 | −0.35 | −0.575 | −0.575 | −0.35 |
| | | ease of explanation | 0.1 | 0.5 | 0.28 | 0.28 | 0.5 |
| NSGA-III with Linear Support Vector Machine | 11.9 | emp | 0.084 | 0.091 | 0.087 | 0.087 | 0.091 |
| | | cardinality | −0.75 | −0.45 | −0.595 | −0.595 | −0.45 |
| | | ease of explanation | 0.15 | 0.45 | 0.31 | 0.31 | 0.45 |
| NSGBOA with Logistic Regression | 10 | emp | 0.091 | 0.096 | 0.093 | 0.093 | 0.096 |
| | | cardinality | −0.65 | −0.25 | −0.5 | −0.5 | −0.25 |
| | | ease of explanation | 0.25 | 0.45 | 0.35 | 0.35 | 0.45 |
| NSGBOA with Neural Network | 9.5 | emp | 0.088 | 0.104 | 0.097 | 0.097 | 0.104 |
| | | cardinality | −0.7 | −0.25 | −0.475 | −0.475 | −0.25 |
| | | ease of explanation | 0.2 | 0.6 | 0.356 | 0.356 | 0.6 |
| NSGBOA with Linear Support Vector Machine | 11.7 | emp | 0.087 | 0.097 | 0.094 | 0.094 | 0.097 |
| | | cardinality | −0.75 | −0.45 | −0.586 | −0.586 | −0.45 |
| | | ease of explanation | 0.3 | 0.45 | 0.393 | 0.393 | 0.45 |

**Table 4.** Single-objective results.

| Algorithm | Number of Features | Objective | Value |
|---|---|---|---|
| Lasso | 20 | emp | 0.1 |
| | | cardinality | −1 |
| | | affordability | 0.25 |
| All features with Logistic Regression | 20 | emp | 0.1 |
| | | cardinality | −1 |
| | | affordability | 0.7 |
| All features with Neural Network | 20 | emp | 0.103 |
| | | cardinality | −1 |
| | | affordability | 0.7 |
| All features with Linear Support Vector Machine | 20 | emp | 0.098 |
| | | cardinality | −1 |
| | | affordability | 0.7 |
| GA with Logistic Regression | 17 | emp | 0.099 |
| | | cardinality | −0.85 |
| | | affordability | 0.6 |
| GA with Neural Network | 17 | emp | 0.103 |
| | | cardinality | −0.85 |
| | | affordability | 0.6 |
| GA with Linear Support Vector Machine | 13 | emp | 0.1 |
| | | cardinality | −0.65 |
| | | affordability | 0.5 |
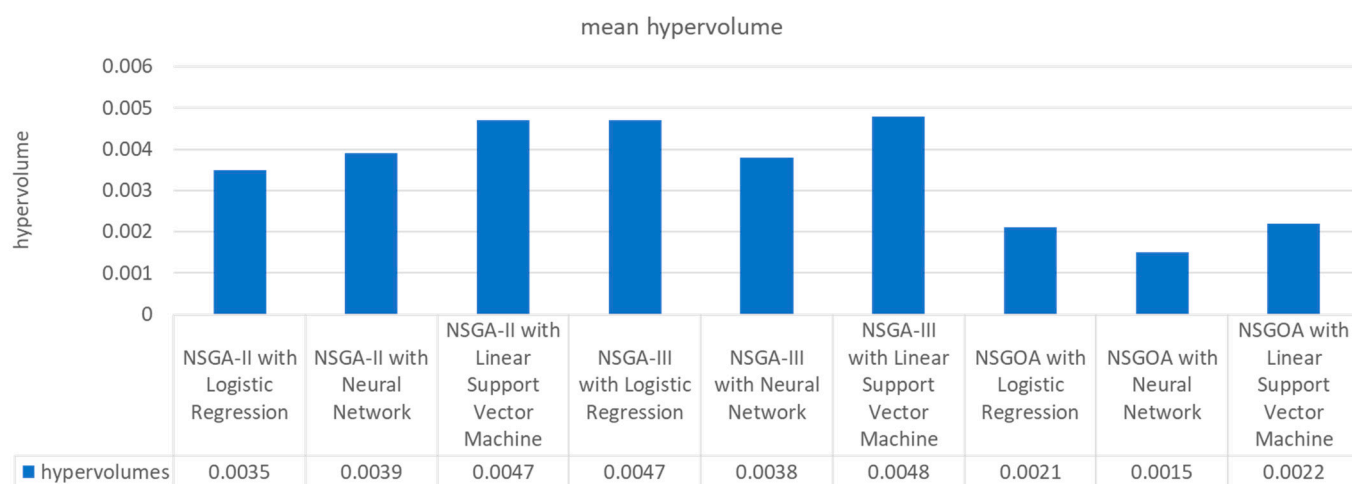
**Figure 1.** Mean hypervolumes.
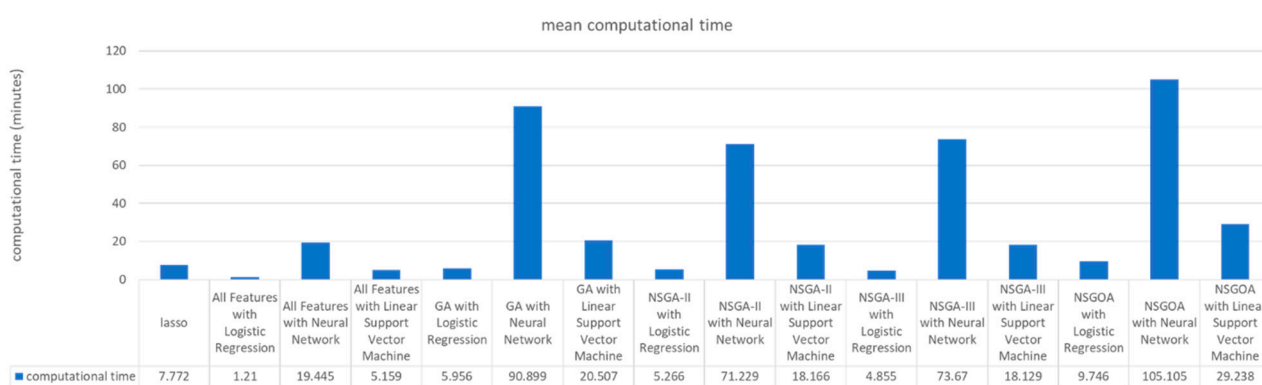


**Figure 2.** Mean computational times.
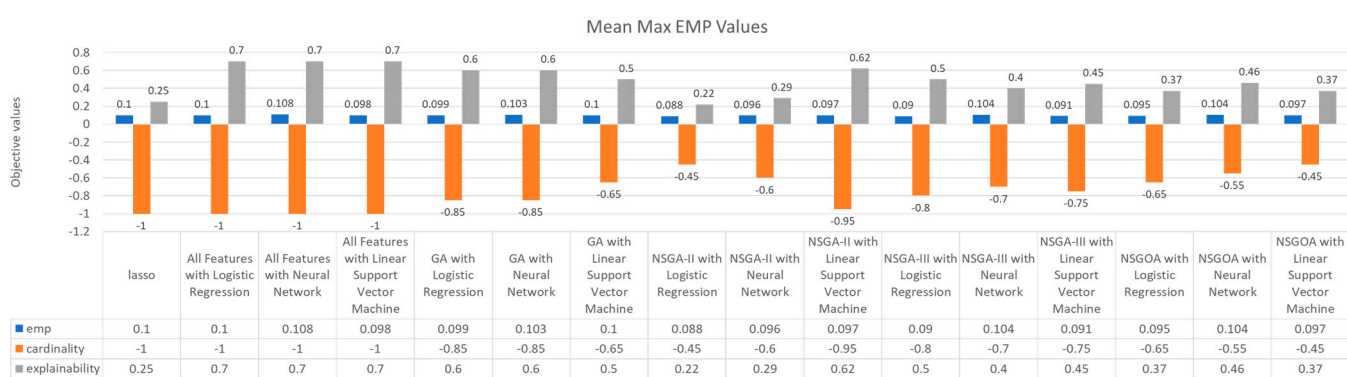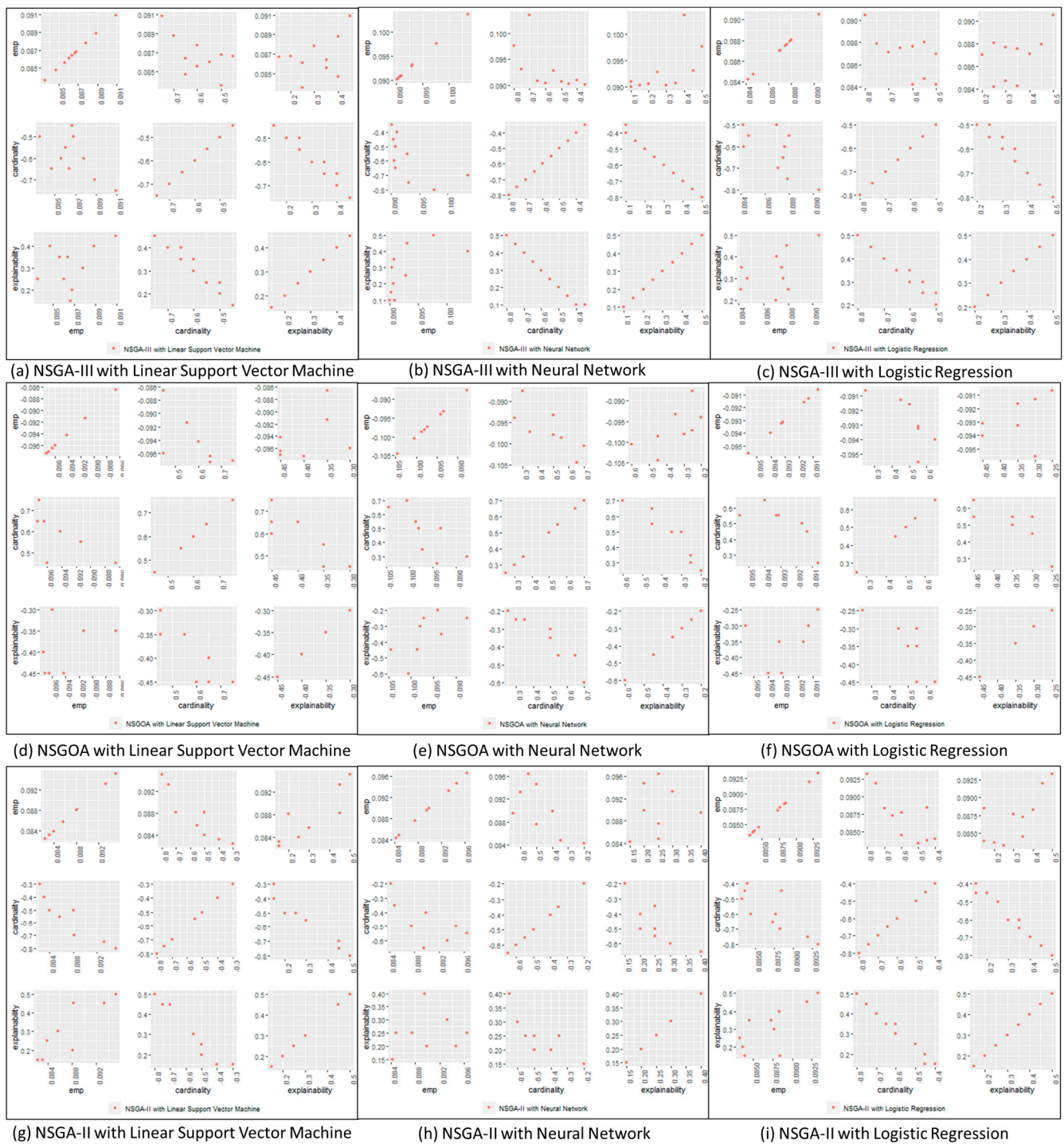


**Figure 3.** Max EMP from each algorithm.

(a) NSGA-III with Linear Support Vector Machine  (b) NSGA-III with Neural Network  (c) NSGA-III with Logistic Regression

(d) NSGOA with Linear Support Vector Machine  (e) NSGOA with Neural Network  (f) NSGOA with Logistic Regression

(g) NSGA-II with Linear Support Vector Machine  (h) NSGA-II with Neural Network  (i) NSGA-II with Logistic Regression

**Figure 4.** Scatter plots of final population.

## 6. Discussion

### 6.1. Base Classifier

The performance of three base classifiers, namely logistic regression, neural network classifier, and linear support vector machine, was compared. To achieve this, each multi-objective optimization (MOO) algorithm was evaluated with the three base classifiers in turn. The neural network classifier gave the smallest mean number of features per solution (Table 3), with the best EMP values across all multi-objective methods. It is possible that the neural network classifier's ability to consider non-linear interactions in the data leads

to better performance with fewer features. Additionally, this classifier required the greatest computational time (Figure 2), regardless of the MOO algorithm applied. This is to be expected as the neural network method has a higher computation complexity than the other two classifiers.

Based on the results given in Figure 1, the base classifier does not appear to have a uniform impact on hypervolume across all multi-objective methods. The hypervolumes obtained by the neural network and linear support vector machine classifiers were similar for NSGA-III and NSGA-II algorithms. On the other hand, for the logistic regression classifier, the hypervolume was higher for NSGA-III. Where number of features and EMP are more of a concern than computational time, the neural network may be the best option, with NSGA-II or NSGOA for reduced hypervolume.

### 6.2. Feature Selection Algorithm

Several feature-selection algorithms were compared, namely LASSO, Genetic Algorithm, NSGA-II, NSGA-III, and NSBGOA. Of the methods that focused on optimizing one objective, LASSO and logistic regression gave the poorest values for cardinality, even when the other two objectives performed well. As observed in Table 4 and Figure 3, LASSO regression gave a high EMP value at the expense of cardinality and explainability. Additionally, each of the base classifiers gave high values for explainability and EMP with poor cardinality when computed with all the available features. This was as expected and highlights the insufficiency of such methods when multiple objectives are to be optimized. Furthermore, it is observed that GA methods achieved improved cardinality compared to the LASSO and all feature evaluations. However, the multi-objective methods gave better cardinality than GA in all but two cases (GA with linear support vector machine outperformed NSGA-II and NSGA-III with linear support vector machine). The GA method would be the best option where a single solution is required, and where a high EMP is the main concern with at least some reduction in the number of features.

Among the multi-objective algorithms (NSGA-II, NSGA-III, and NSGOA), NSBGOA had the longest computation time regardless of base classifier (Figure 2). It was able to produce the smallest hypervolume compared to the other multi-objective methods (Figure 1). The NSBGOA methods also gave the smallest average number of features per feature set (Table 3). In addition, in most cases, the mean objective values were higher than those obtained with NSGA-II and NSGA-III, using comparable base classifiers. Of the multi-objective methods, the NSGOA algorithm would work best where computational time was not a concern, minimizing the number of feature of features was a major consideration, and multiple solutions were required.

### 6.3. Application

Overall, the multi-objective methods were more efficient in reducing the number of features; however, this led to lower EMP and explainability in some cases. This demonstrates the ability of multi-objective methods to balance several factors. Such methods may be advantageous in cases where the decision-maker is willing to make slight sacrifices in EMP and explainability to reduce the number of features, which could result in reduced data-acquisition costs. Furthermore, it should be noted that the multi-objective methods produce several non-dominated solutions or feature sets with different objective values for each. From the business perspective, these methods give the benefit of a posteriori decision-making. To apply these results, the decision-maker would consider the objective values obtained from each subset of features and select the final solution based on these objectives and priorities.

## 7. Conclusions

A comparison was made of the effect of different base classifiers on multi-objective feature-selection methods in credit scoring. Of the base classifiers used (neural network, logistic regression and linear support vector machine), the neural network classifier improved the profit-based measure and reduced the number of features the most. However, it also significantly increased computational time. Further, the base classifier was found to have an uneven impact on the hypervolume of multi-objective optimization output. It was found that all the multi-objective methods gave a better balance of objectives than the single-objective methods. Additionally, the NSBGOA algorithm gave relatively smaller hypervolumes and increased computational time across all base classifiers. It also resulted in better mean objective values in most case. This study showed that the performance of the multi-objective method is affected by the base classifier chosen. As such, the implementation of multi-objective methods should carefully consider the base classifier used for evaluation.

## References

1. Thomas, L.C.; Edelman, B.D.; Crook, N.J. *Credit Scoring and Its Applications*; Society for Applied and Industrial Mathematics: Philadelphia, 2002;.
2. Guyon, I.; Gunn, S.; Nikravesh, M.; Zadeh, L.A.; Ed.; *Feature Extraction, Foundations and Applications*; Springer: Berlin/Heidelberg, Germany, 2006.
3. Djeundje, V.B.; Crook, J.; Calabrese, R.; Hamid, M. Enhancing credit scoring with alternative data. *Expert Syst. Appl.* **2021**, *163*, 113766, doi:10.1016/j.eswa.2020.113766.
4. Maldonado, S.; Flores, Á.; Verbraken, T.; Baesens, B.; Weber, R. Profit-based feature selection using support vector machines—General framework and an application for customer retention. *Appl. Soft Comput. J.* **2015**, *35*, 740–748.
5. Maldonado, S.; Bravo, C.; López, J.; Pérez, J. Integrated framework for profit-based feature selection and SVM classification in credit scoring. *Decis. Support Syst.* **2017**, *104*, 113–121.
6. Odu, G.O.; Charles-Owaba, O.E. Review of Multi-criteria Optimization Methods—Theory and Applications. *IOSR J. Eng.* **2013**, *3*, 1–14, doi:10.9790/3021-031020114.
7. Kozodoi, N.; Lessmann, S.; Papakonstantinou, K.; Gatsoulis, Y.; Baesens, B. A multi-objective approach for profit-driven feature selection in credit scoring. *Decis. Support Syst.* **2019**, *120*, 106–117, doi:10.1016/j.dss.2019.03.011.
8. Emmerich, M.T.; Deutz, A.H. A tutorial on multiobjective optimization: Fundamentals and evolutionary methods. *Nat. Comput.* **2018**, *17*, 585–609.
9. Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., & Murata, T. Evolutionary Multi-Criterion Optimization. In *4th International Conference, EMO 2007 Matsushima, Japan, March 5–8, 2007 Proceedings 13*; Springer: Berlin/Heidelberg, Germany, 2007.
10. Mafarja, M.; Aljarah, I.; Faris, H.; Hammouri, A.I.; Al-zoubi, A.M.; Mirjalili, S. Binary grasshopper optimisation algorithm approaches for feature selection problems. *Expert Syst. Appl.* **2019**, *117*, 267–286, doi:10.1016/j.eswa.2018.09.015.
11. Hichem, H.; Elkamel, M.; Rafik, M.; Mesaaoud, M.T.; Ouahiba, C. A new binary grasshopper optimization algorithm for feature selection problem. *J. King Saud Univ.-Comput. Inf. Sci.* **2019**, doi:10.1016/j.jksuci.2019.11.007.
12. Usman, A.M.; Yusof, U.K.; Naim, S. Filter-Based Multi-Objective Feature Selection Using NSGA III and Cuckoo Optimization Algorithm. *IEEE* **2020**, doi:10.1109/ACCESS.2020.2987057.

13. Simumba, N.; Okami, S.; Kodaka, A.; Kohtake, N. Hybrid Many Objective Metaheuristics for Feature Selection Based on Stakeholder Requirements in Credit Scoring with Alternative Data No Title. Unpublished Manuscript. Under Review. **2021**.

14. Ishibuchi, H.; Imada, R.; Setoguchi, Y.; Nojima, Y. Performance Comparison of NSGA-II and NSGA-III on Various Many-Objective Test Problems. In Proceedings of the 2016 IEEE Congress on Evolutionary Computation, Vancouver, BC, Canada, 16–21 July 2016.

15. Maldonado, S.; Pérez, J.; Bravo, C. Cost-based feature selection for Support Vector Machines: An application in credit scoring. *Eur. J. Oper. Res.* **2017**, *261*, 656–665.

16. Serrano-Cinca, C.; Gutiérrez-Nieto, B. The use of profit scoring as an alternative to credit scoring systems in peer-to-peer (P2P) lending. *Decis. Support Syst.* **2016**, *89*, 113–122, doi:10.1016/j.dss.2016.06.014.

17. Verbraken, T.; Member, S.; Verbeke, W.; Baesens, B. A Novel Profit Maximizing Metric for Measuring Classification Performance of Customer Churn Prediction Models. *IEEE Trans. Knowl. Data Eng.* **2013**, *25*, doi:10.1109/TKDE.2012.50.

18. Verbraken, T.; Bravo, C.; Weber, R.; Baesens, B. Development and application of consumer credit scoring models using profit-based classification measures. *Eur. J. Oper. Res.* **2014**, *238*, 505–513, doi:10.1016/j.ejor.2014.04.001.

19. Bonev, B.; Escolano, F.; Cazorla, M. Feature selection, mutual information, and the classification of high-dimensional patterns. *Pattern Anal. Appl.* **2008**, *11*, doi:10.1007/s10044-008-0107-0.

20. Tibshirani, R. Regression shrinkage and selection via the lasso: A retrospective. *J. R. Stat. Soc.* **2011**, *73*, 273–282.

21. Han, L.; Han, L.; Zhao, H. Engineering Applications of Artificial Intelligence. *Eng. Appl. Artif. Intell.* **2013**, *26*, 848–862, doi:10.1016/j.engappai.2012.10.005.

22. Zhang, Z.; He, J.; Gao, G.; Tian, Y. Sparse multi-criteria optimization classifier for credit risk evaluation. *Soft Comput.* **2019**, *23*, 3053–3066, doi:10.1007/s00500-017-2953-4.

23. Xue, B.; Cervante, L.; Shang, L.; Zhang, M. A Particle Swarm Optimisation Based Multi-Objective Filter Approach to Feature Selection for Classification. In *Proceedings of the PRICAI 2012: Trends in Artificial Intelligence. PRICAI 2012. Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2012.

24. Papouskova, M.; Hajek, P. Two-stage consumer credit risk modelling using heterogeneous ensemble learning. *Decis. Support Syst.* **2019**, *118*, 33–45, doi:10.1016/j.dss.2019.01.002.

25. Emmanouilidis, C.; Hunter, A.; Macintyre, J.; Cox, C. Selecting Features in Neurofuzzy Modelling by Multiobjective Genetic Algorithms. In Proceedings of the ICANN'99. 9th International Conference on Artificial Neural Networks, Edinburgh, UK, 7–10 September 1999; pp. 749–754.

26. Xue, B.; Cervante, L.; Shang, L.; Browne, W.N.; Zhang, M. A Multi-Objective Particle Swarm Optimisation for Filter Based Feature Selection in Classification Problems. *Conn. Sci.* **2012**, *24*, 91–116.

27. Doerner, K.; Gutjahr, W.J.; Hartl, R.F.; Strauss, C.; Stummer, C. Pareto Ant Colony Optimization : A Metaheuristic Approach to Multiobjective Portfolio Selection. *Ann. Oper. Res.* **2004**, *131*, 79–99.

28. Wagner, T.; Beume, N.; Naujoks, B. Pareto-, Aggregation-, and Indicator-Based Methods in Many-Objective Optimization. In Proceedings of the 4th International Conference, EMO 2007, Matsushima, Japan, 5–8 March 2007; pp. 742–756.

29. K, D.; Jain, H. Handling many-objective problems using an improved NSGA-II procedure. In Proceedings of the 2012 IEEE Congress on Evolutionary Computation (CEC'12), Kraków, Poland, June 28–July 1 2012; pp. 1–8.

30. Censor, Y. Pareto Optimality in Multiobjective Problems. *Appl. Math. Optim.* **1977**, *4*, 41–59.

31. Li, B.; Li, J.; Tang, K.; Yao, X. Many-objective evolutionary algorithms: A survey. *ACM Comput. Surv.* **2015**, *48*, doi:10.1145/2792984.

32. Saremi, S.; Mirjalili, S.; Lewis, A. Advances in Engineering Software Grasshopper Optimisation Algorithm : Theory and application. *Adv. Eng. Softw.* **2017**, *105*, 30–47, doi:10.1016/j.advengsoft.2017.01.004.

33. Mays, E.; Nuetzel, P. *Credit Scoring for Risk Managers: The Handbook for Lenders*; Ch. Scorecard Monitoring Reports; South-Western Publishing: Mason, OH, USA, 2004;.

34. Audet, C.; Bigeon, J.; Cartier, D.; Digabel, S. Le; Salomon, L. Performance indicators in multiobjective optimization. *Eur. J. Oper. Res.* **2021**, *292*, 397–422.

35. Dua, D.; Graff, C. *German Credit Dataset*; University of California, School of Information and Computer Science: Irvine, CA, USA, 2019.

36. Khan, S.; Asjad, M.; Ahmad, A. Review of Modern Optimization Techniques. *Int. J. Eng. Tech. Res.* **2015**, doi:10.17577/IJERTV4IS041129.