Vol. 34, No. 4 (2024) | DOI: 10.37190/ord240410

OPEN ACCESS

Operations Research and Decisions

www.ord.pwr.edu.pl

OPERATIONS RESEARCH AND DECISIONS QUARTERLY

Classification with machine learning algorithms after hybrid feature selection in imbalanced data sets

Meryem Pulat[1*] İpek Deveci Kocakoç[2]

[1] *Department of Business, Fırat University, Elazığ, Turkey*

[2] *Department of Econometrics, Faculty of Economics and Business Administration, Dokuz Eylul University, İzmir, Turkey*

[*] *Corresponding author, email address: mpulat@firat.edu.tr*

Abstract

The efficacy of machine learning algorithms significantly depends on the adequacy and relevance of features in the data set. Hence, feature selection precedes the classification process. In this study, a hybrid feature selection approach, integrating filter and wrapper methods was employed. This approach not only enhances classification accuracy, surpassing the results achievable with filter methods alone, but also reduces processing time compared to exclusive reliance on wrapper methods. Results indicate a general improvement in algorithm performance with the application of the hybrid feature selection approach. The study utilized the Taiwanese Bankruptcy and Statlog (German Credit Data) datasets from the UCI Machine Learning Repository. These datasets exhibit an unbalanced distribution, necessitating data preprocessing that considers this unbalance. After acknowledging the datasets' unbalanced nature, feature selection and subsequent classification processes were executed.

Keywords: *machine learning, ensemble learning, classification, feature selection, unbalanced dataset*

1. Introduction

Blaise Pascal and Gottfried Wilhelm Leibniz, in the seventeenth century, developed machines that could mimic human addition and subtraction, marking the historical beginnings of machine learning. Arthur Samuel of IBM introduced the term machine learning in modern history, demonstrating the ability to program computers to learn how to play checkers. Rosenblatt followed this by developing the perceptron as one of the neural network architectures in 1958. However, Minsky's observation that the perceptron's classification ability was only applicable to linearly separable problems led to a decline in initial intense interest in the perceptron. Werbos achieved a breakthrough in 1975 with the development of the multilayer perceptron (MLP) [39]. Quinlan developed decision trees in 1986, while Cortes and Vapnik developed support vector machines [35]. Subsequently, researchers proposed ensemble machine learning algorithms that combine multiple algorithms, such as Adaboost and random forests [3, 31]. More recently, distributed multilayer learning algorithms have emerged under the concept of deep learning.

Received 27 February 2024, accepted 19 October 2024, published online 19 December 2024
ISSN 2391-6060 (Online)/

The costs of publishing this issue have been co-financed by the Department of Operations Research and Business Intelligence at the Faculty of Management, Wrocław University of Science and Technology, Wrocław, Poland

Machine learning can be used to solve classification, regression, and clustering problems. In this study, the classification problem is discussed. In classification problems, a model is created using data whose classes are known (training data), and then this model is used to classify samples whose classes are unknown and have not been encountered by the system before (test data). As examples of classification problems, these might include determining whether an email is spam, determining whether a tumor is malignant, or identifying the image of an animal as a cat, dog, or another animal.

Most data sets contain relevant, irrelevant, or redundant features (attributes). Feature selection is a process aimed at extracting a smaller, yet significant subset of M features from the original N features, thereby effectively minimizing the feature space based on specific evaluation criteria. This process is crucial for enhancing the efficiency of classification algorithms. By selecting relevant attributes, not only does the speed of these algorithms increase, but data quality and the algorithms' performance are also enhanced, leading to more comprehensible results.

Feature selection techniques are broadly categorized into three groups: filter, wrapper, and embedded methods [24]. Filter methods focus on identifying a relevant subset of features from the original set, independent of the learning algorithms. Wrapper methods, in contrast, select features based on their predicted performance in specific learning algorithms [41]. In this research, the initial feature selection was conducted using filter methods due to their computational speed. This was followed by the application of wrapper methods to refine the feature selection further. The classification was then performed using the optimized feature set obtained through wrapper methods.

Since the purpose of a standard classifier is to maximize overall accuracy, the classifier will learn the class with a large number of data points better, and the result of this training will be a "low error rate for the majority class" but a "higher error rate for the minority class". In summary, the classifier will tend to classify all examples as the majority class, providing high accuracy, but meanwhile, it will miss minority examples. In unbalanced data sets, accuracy is high, and we think that we made a very good prediction. This situation is called the accuracy paradox. This metric, which shows prediction accuracy, has been shown to be meaningless and insufficient in unbalanced data sets. Recall (sensitivity), specificity, and precision values become important metrics to measure the performance of the model.

When handling unbalanced datasets, one initial strategy is to modify class distributions through data resampling. This can be achieved through techniques such as undersampling, oversampling, and various sophisticated sampling methods. Undersampling involves reducing instances of the more prevalent class to balance the class distributions, but a notable drawback is the potential loss of valuable data if the dataset already has limited observations. On the other hand, oversampling increases the instances of the minority class to achieve balance. However, this often involves replicating minority class samples, which can lead to overfitting, especially in large datasets with significant unbalanced. Moreover, oversampling can be computationally demanding. Beyond these methods, there are also advanced sampling techniques employing heuristic strategies to achieve a more balanced distribution. An alternative approach in dealing with unbalanced datasets involves selecting appropriate performance metrics.

In the study, the Taiwanese Bankruptcy and Statlog (German Credit Data) datasets from the UCI Machine Learning Repository database were used. Data sets have an unbalanced distribution. Data preprocessing was carried out, taking into account the unbalanced distribution of the dataset, and then feature selection and classification were carried out. While chi-square, information gain (IG) [11], gain

ratio (GR), symmetric uncertainty coefficient (SU) [24], Correlation Based Feature Selection (CFS) [11] and RELIEF [15] methods are used in the filtering step for feature selection; recursive feature elimination (RFE) [10], Genetic Algorithm (GA) [38], Simulated Annealing (SA) [19] and BORUTA [16] were used as wrapper methods. The classification process was performed using k-nearest neighbors (KNN), naive Bayes (NB) [28], CART (rpart) [4], bagged CART (TREEBAG) [2], J48 [27], C5.0, eXtreme Gradient Boosting (XGBTREE) [6], linear discriminant analysis (LDA), Multi-Layer Perceptron (MLP) [39], Multivariate Adaptive Regression Spline (EARTH) [8], random forest (RF) [3], rotation forest [29], gradient boosting machine (GBM) [31], support vector machines with linear kernel (SVMLINEAR) [35], support vector machines with polynomial kernel (SVMPOLY), support vector machines with radial basis function kernel (SVMRADIAL), tree models from genetic algorithms (EVTREE) [9], and generalized linear model (GLM) [23] algorithms, and their performances were compared using model performance criteria.

As for the contributions of this article, firstly, it presents a comprehensive study by combining various filter and wrapper feature selection methods. In general, this hybrid technique has received little attention in the literature, due in part to the lack of a defined standard for selecting specific methods, necessitating extensive experimentation that takes a long time. These methods may serve as foundational feature selection approaches for future related research. Secondly, the study examines the impact of feature selection on different classification algorithms. There is no such comprehensive study in the literature. Thirdly, it observes the effects of different strategies employed to address unbalances in datasets on classification algorithms. Different approaches have been used in the literature to balance the data set. However, in the study, the performance of feature selection on the unbalanced data set can be seen.

In the study, by combining the methods in the literature in the feature selection step, both a fewer number of features, that is, a less complex model, and better performance were obtained with the sequential feature selection approach. There is no such comprehensive model attempt in the literature. In the classification step, a very comprehensive model trial was conducted. Additionally, the performance of classification models after sequential feature selection was examined. It has been observed that the performance of classification models increases after sequential feature selection.

2. Literature review of datasets

The literature review for the datasets used in the study is presented in Table 1. This table features only studies that have achieved high performance. It includes the bibliographic references of the articles and the most commonly used performance criteria: Accuracy (Acc), Sensitivity (also known as Recall or True Positive Rate – Sen), and Specificity (True Negative Rate – Spe).

In the study of [7] focuses on explaining a bankruptcy prediction model using a counterfactual example. Counterfactual-based explanation provides consumers with an alternate instance in which they can obtain the desired output from the model. This work presents a genetic algorithm (GA)-based counterfactual generation technique that considers feature importance alongside other essential parameters. In this study, the prediction model was trained using a balanced bankruptcy dataset. Experiments were carried out on several bankruptcy datasets, employing machine learning techniques such as ANN and SVM.

Empirical experiments show that the suggested method outperforms a basic counterfactual generating algorithm. For the Taiwanese Bankruptcy Prediction dataset, the SVM algorithm gave the best results.

Table 1. Literature review of data sets

Data sets	Studies	Acc [%]	Sen [%]	Spe [%]
Taiwanese Bankruptcy Prediction	Cho and Shin, 2023	87	85	
	Liang et al., 2015	83		
	Brenes et al., 2022	87	87	89
	Khemka et al., 2023	88		
	Almeida, 2023	94	94	
	Youness et al., 2023	94		
Statlog, German Credit Data	Lin et al., 2009	82		
	Xiao et al., 2016	75		
	Tsai, 2014	89		
	Tsai et al., 2014	76		
	Quan and Sun, 2024	77	95	79
	Seera et al., 2024	77		
	Herrera-Malambo et al., 2023	83		
	Gicić and Đonko, 2023	87	87	
	Emmanuel et al., 2024	83		

Liang et al. [17] examined the effect of applying filter and wrapper-based feature selection methods on financial distress prediction. Two bankruptcy and two credit data sets were used in the experiments. Linear discriminant analysis (LDA), t-test, logistic regression (LR) as filter-based feature selection methods. Two methods were used for wrapper-based feature selection: genetic algorithm (GA) and particle swarm optimization (PSO). Six classification techniques were used as classifiers: linear SVM, RBF SVM, k-NN, Naïve Bayes, CART and MLP. After GA feature selection, the linear SVM classifier gives the highest accuracy value.

In the study of Brenes et al. [5] different setups of optimization algorithms, activation functions, number of neurons, and number of layers were considered for the Multilayer Perceptron (MLP) algorithm. Various evaluation metrics such as average accuracy, specificity, sensitivity, and precision were used to find the parameter setup that achieved the best results. The MLP algorithm with the best performance has two hidden layers, Adam optimization algorithm, ReLU activation function, two hidden layers and the number of cells in the neurons in layer 1 is 3; It is the algorithm with the number of cells in the neurons in layer 2 is 4.

Khemka's et al. [14] study was conducted on Taiwanese Bankruptcy Prediction data set. The application of various machine learning techniques, including SVM, naive Bayes, Decision Trees, and Logistic Regression is examined in this article. The main goal of this article is to create a reliable and accurate bankruptcy prediction model that financial institutions may use to identify businesses that are most likely to fall behind on their payments. In this study, SVM classifier gives the highest accuracy value.

Almeida [1] to examine the variations in performance for the prediction of bankruptcy, the author investigates and applies the functioning of several neural network approaches, including the fundamental design and the use of regularization strategies, such as L1, L2, Dropout, and Early Stopping. To compare their effectiveness with neural networks, other machine learning algorithms including SVM, random forest, and XGBoost have also been put into practice. The accuracy values obtained for the L1, L2,

Dropout, and Early Stopping approaches, as well as the regular neural network model, were 82%, 49%, 89%, 90%, and 94%, respectively. The accuracy of the remaining models, SVM, RF, and XGBoost, was 87%, 86%, and 85%, respectively.

Lin et al. [18] studied the Statlog data set. In order to choose a subset of useful features and find appropriate parameter values for decision trees (DT) and support vector machines (SVM) without lowering the classification accuracy rate, this work uses particle swarm optimization (PSO). The outcomes of the experiments shown that the suggested methods may achieve a better parameter setting, eliminate unnecessary features, and greatly increase classification accuracy. SVM with FS gives the highest accuracy value.

Xiao et al. [40] propose an ensemble classification method for credit scoring based on supervised clustering. The suggested method divides the data samples of each class into a predetermined number of clusters using supervised clustering. A set of training subsets was then created by pairwise combining clusters from various classes. For every training subset, a unique base classifier is constructed. The outputs of these base classifiers are coupled with weighted voting for an example whose class label needs to be predicted. In the tests, two popular techniques for ensemble classification—bagging and RSM—were utilized as comparisons. In terms of classification performance, Bagging and RSM are contrasted with the suggested ECSC. The binary ensemble classification algorithms Bagging-RS, RS-Bagging, and DCE-CC were contrasted with the suggested ECSC. the highest accuracy value. The algorithm with the best performance is RS-Bagging ECSC (Logit as classifier).

The goal of Tsai's [33] research is to create a novel hybrid financial distress model by merging classifier ensembles with the clustering technique. Specifically, these four different types of bankruptcy prediction models are developed using three classification techniques (logistic regression, multilayer-perceptron (MLP) neural network, and decision trees) and two clustering techniques (self-organizing maps, or SOMs, and k-means). Consequently, the type I and II errors and average prediction accuracy of 21 individual models are compared. The greatest results were obtained by merging MLP classifier ensembles with Self-Organizing Maps (SOMs) across five related datasets.

Tsai et al. [34] examined classification ensembles based on two popular combination approaches, bagging and boosting, with three widely used classification algorithms, including multilayer perceptron (MLP) neural networks, support vector machines (SVM), and decision trees (DT). compared as. Three general datasets have been used to conduct the experiments. Additionally, the Wilcoxon signed-rank test demonstrates that DT ensembles outperform other classifier ensembles in a significant way through reinforcement. Furthermore, an additional investigation was carried out using the Taiwan bankruptcy dataset on a real-world scenario; this further proved the superiority of DT communities over others. While DT ensembles using both boosting and bagging performed second best for the German dataset, their accuracy rate was somewhat lower than that of MLP ensembles using bagging.

Quan and Sun [26] conducted studies on Statlog data set. The factorization machine model is used in the field of credit risk assessment in this article. Numerical experiments are carried out on four real-world credit risk evaluation datasets to demonstrate the efficacy of the factorization machine credit risk assessment model and compare its performance with other classification techniques like logical regression, Support Vector Machine, k-nearest neighbors, and artificial neural network. The experimental findings demonstrated that, in comparison to previous machine-learning models, the suggested factorization ma-

chine credit risk assessment model achieves higher accuracy and is computationally more efficient on real-world datasets. The algorithm with the best performance for Statlog dataset is factorization machine (FM) model.

In the study presented, by combining the methods in the literature in the feature selection step, both a fewer number of features, that is, a less complex model, and better performance were obtained with the sequential feature selection approach. There is no such comprehensive model attempt in the literature. In the classification step, a very comprehensive model trial was conducted. Additionally, the performance of classification models after sequential feature selection was examined. It has been observed that the performance of classification models increases after successive feature selections.

3. Method

In this study, the Taiwanese Bankruptcy and Statlog (German Credit Data) datasets from the UCI Machine Learning Repository database were used. Considering the unbalanced distribution of these datasets, both undersampling and oversampling techniques were employed. Following hybrid feature selection, classification was performed with various machine learning algorithms, and their performances were compared.

3.1. Feature selection

Most data sets contain relevant, irrelevant, or redundant attributes (variables). Feature selection can be defined as a process that selects a minimum subset of M features from the original set of N features, thus optimally reducing the feature space according to a given evaluation criterion. Feature selection plays an important role in the performance of classification algorithms. Attribute selection; It speeds up the algorithms, improves the data quality and therefore the performance of the algorithms, and increases the understandability of the results of the algorithms. Feature selection algorithms; It is divided into three: filtering, wrapper, and embedded approach [24]. Filter methods essentially identify a subset of features from the original feature set, with evaluation criteria given independently of the learning algorithms. Wrapper methods, on the other hand, select features with high prediction performance predicted by the specified learning algorithms [41]. Since embedded methods perform feature selection as part of the model training process; They perform classification and feature selection simultaneously.

The main goal of feature selection is to reduce the number of features in order to achieve a high accuracy value without using all of the data we have. The goals of feature selection are diverse, the most important of which are (a) to avoid overfitting and improve model performance; (b) to be able to obtain a simple model that is faster to calculate with little or no degradation in prediction accuracy; and (c) to select the most informative features by the class label.

A typical feature selection process consists of four basic steps: subset creation, subset evaluation, stopping criteria, and result validation. Subset creation is a search procedure that produces subsets of candidate features for evaluation based on a specific search strategy. We evaluate each subset of candidates against a specific evaluation criterion and compare them to the previous best candidate. We repeat the process of creating a subset and evaluating it until we meet a specific stopping criterion. Figure 1 shows the four basic steps of feature selection.

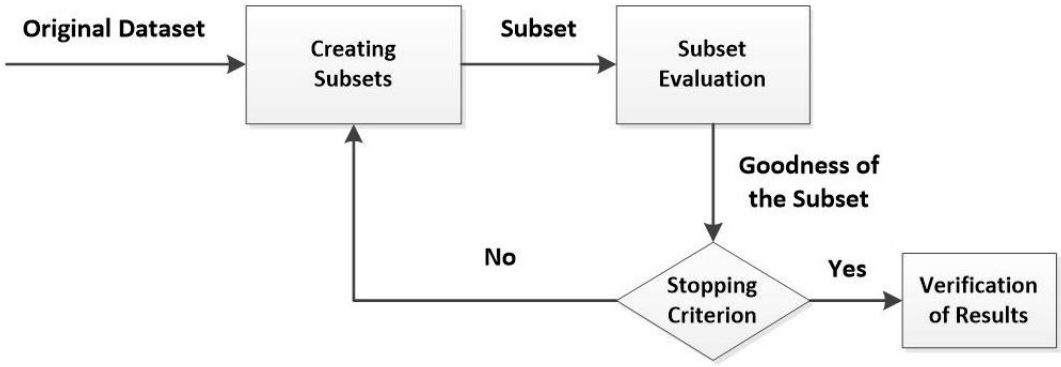

Figure 1. Steps of the feature selection process

Filter methods. These methods are known as the oldest feature selection method based on analyzing the effects of the features one by one on the explanation or verification power of the model. These methods choose features based on measurements of statistical factors such as distance, information, dependency, and consistency; they don't use any classification algorithms [30]. Separate and independent processes carry out feature selection and classification. These methods, working with similar logic, calculate a value for each attribute using statistical functions, and then select the attribute with the highest value among these calculated values. We present the selected features as input data for classification algorithms and evaluate the classification process performance using these features [32]. Examples of filtering methods include Fisher score, chi-square, information gain, gain ratio, F-score, symmetric uncertainty coefficient, correlation-based feature selection, and RELEFF. Figure 2 displays the diagram of the filtering feature selection method.

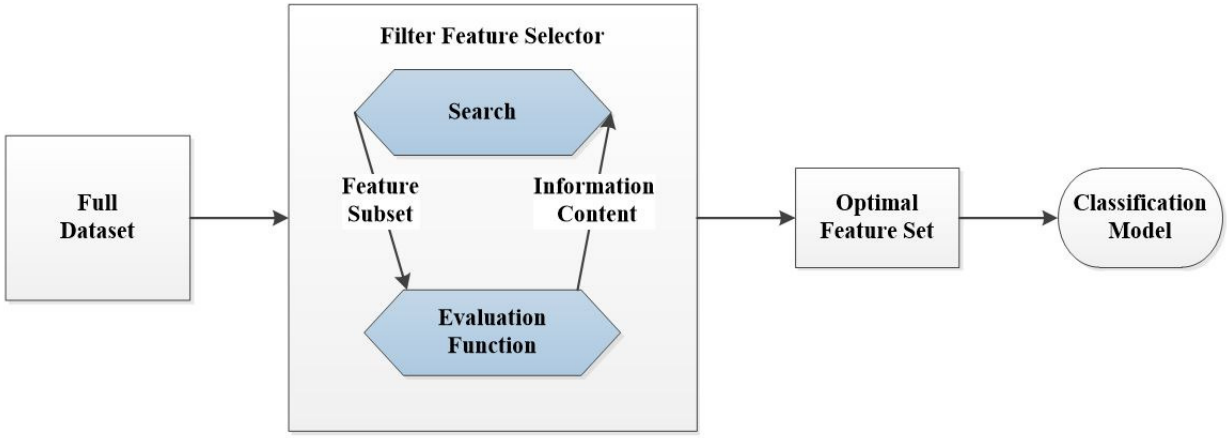

Figure 2. Steps of the filter feature selection process

Wrapper methods. Features that show the best prediction performance are selected using various learning algorithms, where efficiency is measured based on the correct classification rate for feature selection. In other words, the wrapper method uses machine learning algorithms, and the classifier's accuracy rate is the measure of feature selection. In each iteration, a classification result is obtained for a specific feature subset [30]. Examples of wrapper methods are sequential forward selection, sequential backward selection, sequential forward floating selection, sequential backward floating selection, recursive feature

elimination, Genetic Algorithm, Simulated Annealing, and BORUTA. Figure 3 shows a diagram of the wrapper feature selection method.

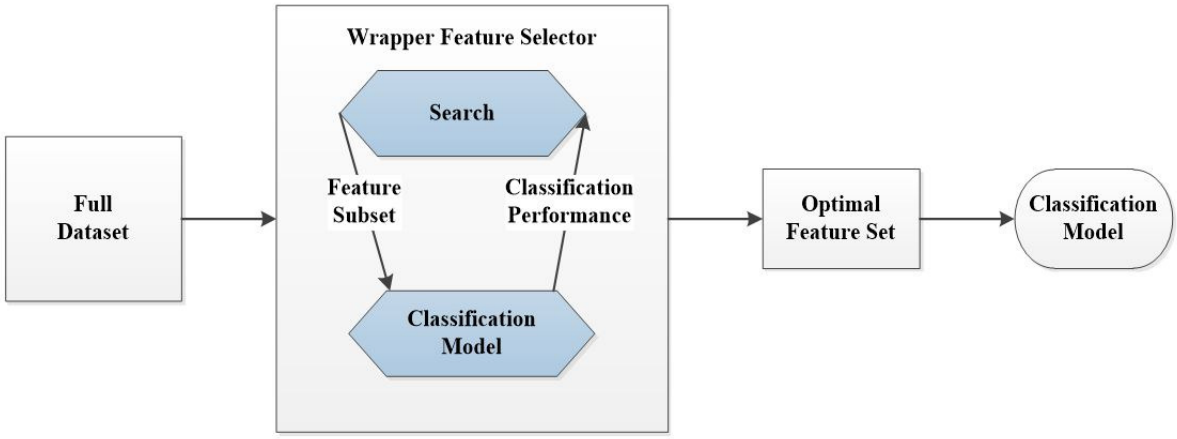

Figure 3. Steps of the Wrapperfeature selection process

Embedded method. Since the model performs feature selection as a part of the training process, it performs classification and feature selection simultaneously. In other words, the machine learning algorithm makes feature selection within itself. Decision trees are one example of this method. The decision tree algorithm inherently performs feature selection because, at each training step, branching is performed by selecting the best feature according to various criteria to split the tree. The LASSO is another example of an embedded method. If we talk about the positive aspects of embedded methods, embedded methods select model-specific attributes. In this sense, embedded methods strike a balance between computational cost and quality of results, resulting in a high success rate. The disadvantages of embedded methods include their direct dependence on the learning algorithm.

3.2. Machine learning classifiers

k-Nearest Neighbor (kNN). This non-parametric method calculates the distances between each observation in the sample set and the desired class value, then selects the k number of observations with the smallest distance.

naive Bayes (NB), This algorithm is a simple probabilistic classifier that calculates a set of probabilities by counting the frequency and combinations of values in a given data set. The algorithm uses Bayes' theorem and assumes that all variables are independent given the value of the class variable [25]. Let $X = \{x_1, x_2, \ldots, x_n\}$ be a data sample with unknown class membership, and assume that there are m classes. We calculate the probabilities related to $C_1, C_2, \ldots, C_n$ class values using equation:

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \tag{1}$$

Since it is assumed that the x_i values of the example are independent of each other, equation (2) can be used:

$$P(X|C_i) = \prod_{k=1}^{n} P(X_k|C_i) \tag{2}$$

To classify unknown example X, select the largest of the probability values calculated above, and the class is defined as the class to which the unknown example belongs, as shown in equation (3).

$$\arg\max_{C}\{P(C_i|X)\} \tag{3}$$

Support Vector Machine (SVM). The first purpose of the Support Vector Machine classifier is to determine the line (hyperplane) that will separate two classes. In other words, it is to create an optimal separating hyperplane between two classes to minimize generalization error and thus maximize margin. An infinite number of hyper-planes can separate any two classes (as shown in Figure 4), and SVM attempts to identify the hyperplane that minimizes the generalization error (i.e., the error for unseen test patterns).

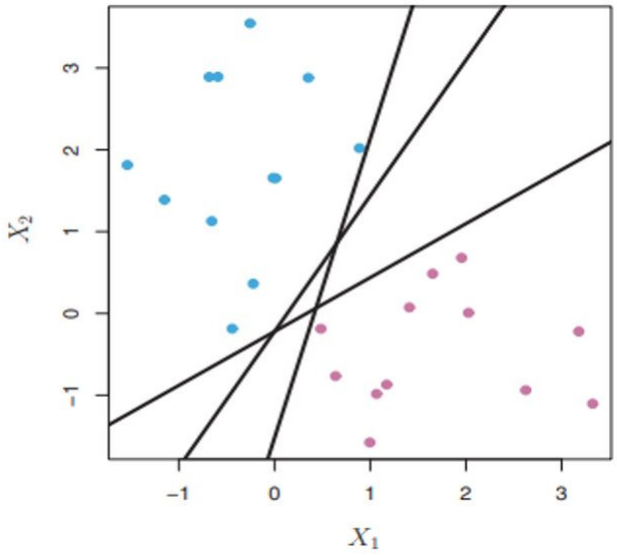

Figure 4. An infinite number of hyperplanes can separate any two classes. Source: [13], p. 340.

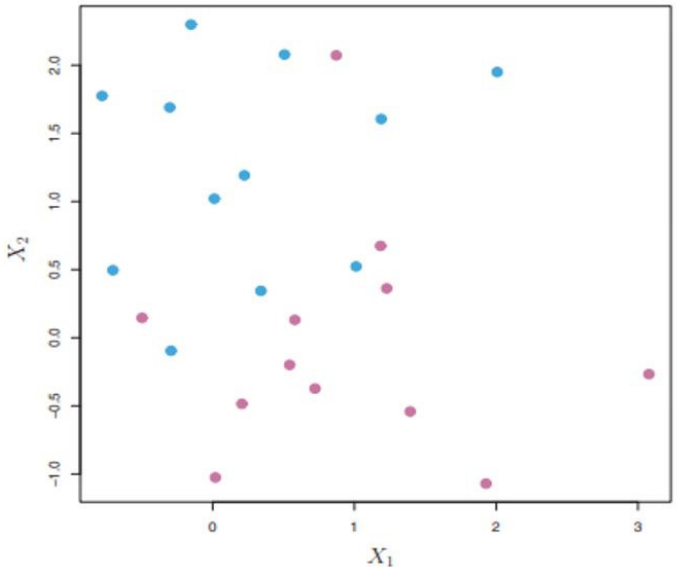

Figure 5. Example of two classes shown in blue and purple.Source: [13], p. 344

Data sets may or may not be linearly separable. In cases where the data cannot be separated linearly (as shown in Figure 5), non-linear classifiers can be used instead of linear classifiers. When SVM cannot distinguish data linearly, it employs kernel functions to analyze the data by relocating it to higher-dimensional spaces. Choosing the right kernel function is crucial, as it can lead to varying performances [22]. Equation (4a)–(4d) [12] provides frequently used kernel functions:

$$K(x_i, x_j) = (x_i^T x_j) \tag{4a}$$

$$K(x_i, x_j) = (\gamma x_i^T x_j + 1)^d \tag{4b}$$

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2) \tag{4c}$$

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r) \tag{4d}$$

where (4a) is linear kernel function, (4b) – d degree polynomial kernel function, (4c) – radial basis kernel function, and (4d) – sigmoid kernel function

Decision tree. In its simplest definition, decision tree analysis is a split and managed approach to distribution and regression. Mitchell [20] defined decision tree development as a method used to estimate discrete value target functions in which the learned function is represented by a decision tree. Decision tree learning is one of the most widely used and practical methods for inductive inference [20]. Decision sets can be used to isolate features and extract patterns that are important in large databases for predictive programming. A decision tree is formed by iteratively splitting the feature space of the training setup. The goal is to provide a set of decision rules that partition the feature space to provide an informative and robust evolving model. Once a decision rule is selected, the feature space is divided into two separate subspaces. The partitioning process is then recursively applied to each of the resulting subspaces until all resulting subspaces contain instances of a single class, preserving a broken decision tree [21]. Create a tree for decision tree leaves, and then the attribute values of the entered data whose output value is unknown are tested in the decision tree. A path is followed from the root to the leaf node, and as it progresses, a prediction is made for the class [20].

Ensemble learning methods. We introduce the concept of ensemble learning to enhance the stability and prediction accuracy of a single learning algorithm in classification, clustering, and regression problems. Ensemble learning methods aim to create models that can predict better performance by reducing the generalization error of basic learning algorithms and increasing the correct classification rate. Ensemble learning modeling is a current field of machine learning research that produces a single final prediction by combining a set of individually trained base models using a specific additive rule. The fact that the base, or basic, models are accurate and diverse ensures that the collection results give more accurate results than individual models. The ensemble model can provide more reliable and accurate predictions than the traditional individual prediction model [36]. The ensemble learning method can be divided into two categories: heterogeneous ensemble and homogeneous ensemble, according to the basic model-building strategy. In the heterogeneous community model, basic models are created by applying

the same training data to different learning algorithms or the same algorithms with different parameter settings. This is an example of stacking. The homogeneous ensemble model creates basic models by applying different training data, resampled from the original data to the same learning algorithm with the same parameter settings. Examples of this are bagging, boosting, random forest, and random subspace [37]. So, the heterogeneous ensemble model lets different learning algorithms work well with each other, and the homogeneous ensemble model improves their ability to make accurate predictions by teaching a chosen learning algorithm with a variety of training data sets.

3.3. Software

The algorithms in this study were applied using packages compatible with RStudio version 1.4, based on R 3.6.1. Different algorithms can be explored at Machine Learning CRAN Task View. In the R program, feature selection was conducted using the `FSelector`, `FSelectorRcpp`, `FSinR`, `caret`, and `Boruta` feature selection packages. Machine learning algorithms were implemented using `caret`, `naivebayes`, `kernlab`, `MASS`, `RSNNS`, `evtree`, `rpart`, `C50`, `plyr`, `Rweka`, `Earth`, `randomForest`, `gbm`, `xgboost`, `ipred`, `e1071`, `fastAdaboost`, `mboost`, `caTools`, `rotationForest`, and `caretEnsemble` packages. It is also possible to implement algorithms using different packages.

3.4. Datasets

The Taiwanese Bankruptcy and Statlog (German Credit Data) datasets from the UCI Machine Learning Repository database were utilized in the study. The first dataset comprises 6819 samples with 96 features in total, including 95 descriptive features and output variables (bankrupt and non-bankrupt firms). Six of the attributes are integer variables; the dependent variable is also an integer; and the remaining eighty-nine attributes are continuous. The class distribution includes two sample clusters: 220 bankrupt companies and 6599 non-bankrupt companies. The second dataset contains 1000 samples with 21 attributes in total, including 20 descriptive features and output variables (good and bad credit). There are twelve categorical features, six integers, two binary variables, and a binary dependent variable. This dataset's class distribution includes two clusters: 700 good and 300 bad credit cases. Both datasets are unbalanced. Detailed information about the datasets is provided in Table 2 and Figure 6.

Table 2. Information on datasets

Characteristic	Taiwanese Bankruptcy Prediction[1]	Statlog (German Credit)[2]
Data set characteristics	multivariate	multivariate
Feature type	integer	categorical, integer
Associated tasks	classification	classification
Subject area	business	social science
Number of instances	6819	1000
Number of features	96	21

[1] https://archive.ics.uci.edu/dataset/572/taiwanese+bankruptcy+prediction
[2] https://archive.ics.uci.edu/dataset/144/statlog+german+credit+data

Both datasets are unbalanced. Initially, the data was analyzed without considering this unbalanced distribution. Subsequently, the performance of the algorithms was evaluated in both datasets through undersampling and oversampling techniques.

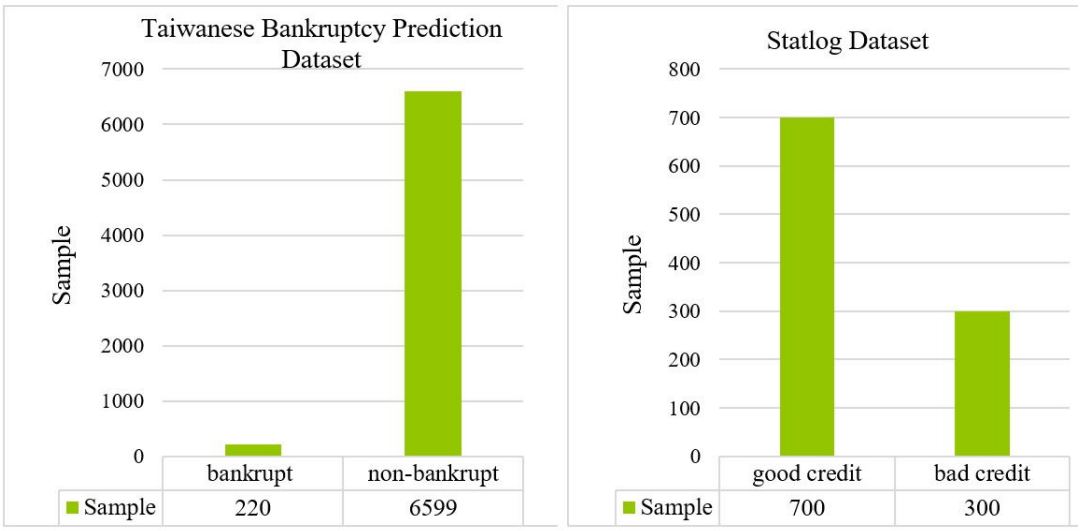

Figure 6. Dataset target class distribution

3.5. Hybrid feature selection approach

In this study, a hybrid feature selection method was employed, combining the advantages of both filter and wrapper methods. Initially, candidate features were selected from the original feature set using computationally efficient filter methods. This candidate feature set was then further refined using more accurate wrapper methods. After sequential feature selection, classification was performed using the final features selected. While chi-square, information gain, gain ratio, symmetric uncertainty coefficient, Correlation Based Feature Selection and RELIEF methods are used in the filtering step for feature selection; recursive feature elimination, Genetic Algorithm, Simulated Annealing, and BORUTA were used as wrapper methods. The classification process was performed using k-nearest neighbors, naive Bayes, CART, bagged CART, J48, C5.0, eXtreme Gradient Boosting, linear discriminant analysis, Multi-Layer Perceptron, Multivariate Adaptive Regression Spline, random forest, rotation forest, gradient boosting machine, support vector machines with linear kernel, support vector machines with polynomial kernel, support vector machines with radial basis function kernel, tree models from genetic algorithms, and generalized linear model algorithms, and their performances were compared using model performance criteria. By integrating filter and wrapper methods, we can improve classification accuracy beyond what is achievable with filter methods alone, while also reducing processing time compared to using only wrapper methods. The flow diagram of the proposed feature selection approach is presented in Figure 7. Generally, this hybrid approach has not been widely explored in the literature, partly because there is no established standard for choosing specific methods.

4. Results

Considering the unbalanced distribution of the datasets, both undersampling and oversampling techniques were employed. Subsequently, hybrid feature selection was conducted (Figure 7), followed by classification using various machine learning algorithms, with their performances subsequently compared. In general, all algorithms have improved performance after sequential feature selection. Only the algorithms that give the best results are given in the study. The following repository provide all the algorithm results: https://github.com/ipekdk/unbalanced_article.

In the study, for the filter step of feature selection, methods such as chi-square, information gain (IG), gain ratio (GR), symmetric uncertainty coefficient (SU), correlation-based feature selection (CFS), and RELIEF were utilized. In contrast, recursive feature elimination (RFE), genetic algorithm (GA), simulated annealing (SA), and BORUTA served as the wrapper methods.

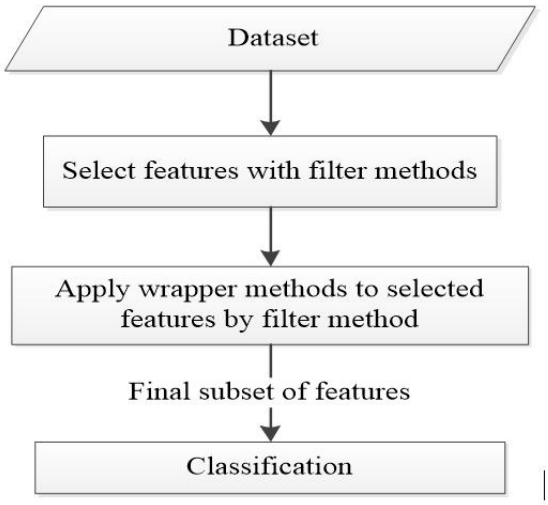

Figure 7. Hybrid feature selection approach

For the classification process, algorithms including k-nearest neighbors (KNN), naive Bayes (NB), CART (rpart), bagged CART (TREEBAG), J48, C5.0, eXtreme Gradient Boosting (XGBTREE), linear discriminant analysis (LDA), multi-layer perceptron (MLP), multivariate adaptive regression spline (EARTH), random forest (RF), rotation forest, gradient boosting machine (GBM), support vector machines with linear kernel (SVMLINEAR), support vector machines with polynomial kernel (SVMPoly), support vector machines with radial basis function kernel (SVMRADIAL), tree models from genetic algorithms (EVTREE), and generalized linear model (GLM) were employed.

4.1. Unbalanced dataset results

In this section, we present the results obtained without performing any balancing operations on the datasets. For the Taiwan Bankruptcy Prediction dataset, we report the outcomes of algorithms that achieved a certain threshold in both accuracy and specificity (the correct prediction rate of the lower class). Similarly, in the Statlog (German Credit Data) dataset, we detail the results of algorithms that surpassed a specified level of accuracy and sensitivity (the correct prediction rate of the lower class).

Upon examining Table 3, it is observed that while the algorithms for the Taiwanese dataset generally show high accuracy, their specificity values are notably low. The Treebag algorithm emerges as the one with the highest accuracy and specificity. In our analysis, specificity represents the algorithm's capability to accurately identify bankrupt businesses. This suggests a significant weakness in the model's prediction of the negative class label, highlighting that relying solely on accuracy is insufficient for unbalanced datasets.

Similarly, in the German dataset, focusing only on accuracy proves inadequate. The sensitivity values, indicating the model's prediction accuracy for the positive class label (bad examples), are low. The algorithm with the highest accuracy in this dataset is random forest (rf).

Table 3. Unbalanced dataset results

Data Set	Results	Algorithm	Acc	Sen	Spe
Taiwanese Bankruptcy Prediction	Algorithms with accuracy values of 0.97 and greater	naive_bayes	0.97	1	0
		svmRadial	0.97	1	0
		mlp	0.97	1	0
		evtree	0.97	1	0
		rpart	0.97	0.99	0.09
		C5.0	0.97	0.99	0.23
		earth	0.97	0.99	0.18
		adaboost	0.97	1	0.23
		rf	0.97	0.99	0.20
		treebag	0.97	0.99	0.25
		glmboost	0.97	0.99	0
		rotationForest	0.97	0.99	0.14
	Algorithms with specificity values of 0.50 and greater	glm	0.93	0.95	0.55
		lda	0.86	0.86	0.93
Statlog (German Credit Data)	Algorithms with accuracy values of 0.70 and greater	glm	0.74	0.43	0.87
		svmLinear	0.73	0.42	0.86
		svmPoly	0.72	0.35	0.88
		svmRadial	0.73	0.33	0.89
		lda	0.73	0.43	0.86
		evtree	0.72	0.30	0.90
		rpart	0.70	0.42	0.81
		C5.0	0.72	0.47	0.83
		j48	0.73	0.42	0.86
		earth	0.73	0.37	0.88
		rf	0.75	0.30	0.94
		gbm	0.72	0.42	0.84
		xgbtree	0.72	0.40	0.86
		treebag	0.72	0.40	0.85
		adaboost	0.74	0.32	0.92
		glmboost	0.73	0.32	0.91
		LogitBoost	0.70	0.40	0.83
		rotationForest	0.73	0.42	0.86
	Algorithms withsensitivity values of 0.40 and greater	glm	0.74	0.43	0.87
		nb	0.69	0.62	0.72
		svmLinear	0.73	0.42	0.86
		lda	0.73	0.43	0.86
		rpart	0.70	0.42	0.81
		C5.0	0.72	0.47	0.83
		j48	0.73	0.42	0.86
		gbm	0.72	0.42	0.84
		xgbtree	0.72	0.40	0.86
		treebag	0.72	0.40	0.85
		LogitBoost	0.70	0.40	0.83
		rotationForest	0.73	0.42	0.86

Further analysis of the Taiwanese dataset shows that among machine learning algorithms with specificity values of 0.50 and above, linear discriminant analysis (lda) holds the highest specificity. In the German dataset, when considering algorithms with specificity values of 0.40 and above, the naive Bayes (nb) algorithm leads in specificity.

Regarding the number of variables chosen through sequential feature selection in the Taiwanese dataset, the CFS + SA algorithm results in the least complexity by selecting the fewest variables. The same trend is observed in the German dataset with the CFS + SA algorithm.

Upon examining Table 4, it is noted that in the Taiwanese dataset, the algorithm demonstrating the highest accuracy and specificity values is the LogitBoost algorithm, following the RELIEF + RFE sequential feature selection. When focusing on algorithms with high specificity values within this unbalanced dataset, naive Bayes stands out as the top performer post RELIEF + RFE sequential feature selection.

Table 4 presents the results of machine learning algorithms after sequential feature selection, considering only those algorithms that surpassed a certain performance threshold in both datasets.

Table 4. Results of machine learning algorithms after sequential feature selection for unbalanced data set

Data set	Results	Feature selection method	Algorithm	Acc	Sen	Spe
Taiwanese Bankruptcy Prediction	algorithm with accuracy ≥ 0.97	RELIEF + RFE	adaboost	0.97	1	0.09
		RELIEF + RFE	LogitBoost	0.97	0.99	0.36
		RELIEF + GA	adaboost	0.97	1	0.16
		SU + GA	rf	0.97	1	0.16
		SU + BORUTA	adaboost	0.97	1	0.09
	algorithm with high specifity value	RELIEF + RFE	naive_bayes	0.90	0.91	0.64
		RELIEF + GA	naive_bayes	0.94	0.95	0.57
		RELIEF + BORUTA	naive_bayes	0.95	0.96	0.45
		RELIEF + RFE	LogitBoost	0.97	0.99	0.36
Statlog (German Credit Data)	algorithm with accuracy ≥ 0.75	SU + RFE	earth	76	49	91
		SU + RFE	rf	77	54	89
		SU + RFE	xgbtree	76	51	89
		SU + GA	xgbtree	76	50	90
	algorithm with sensivity ≥ 0.55	SU + RFE	C5.0	69	56	76
		SU + RFE	treebag	74	56	83
		GR + SA	treebag	69	60	73

In the context of the German dataset, the algorithm achieving the highest accuracy is the random forest (RF) algorithm, subsequent to SU + RFE sequential feature selection. In terms of sensitivity, the best-performing algorithm is Treebag following the GR + SA sequential feature selection process.

4.2. Balanced (undersampling) Data Set Results

This section presents the results obtained after balancing the datasets through undersampling. The Taiwanese dataset was balanced to include 440 samples, comprising an equal number of 220 bankrupt and 220 non-bankrupt cases. Similarly, the German dataset was balanced to consist of 600 samples, with an equal split of 300 good and 300 bad credit cases.

When examining the number of variables selected as a result of sequential feature selection in the Taiwanese undersampled dataset, the CFS + SA algorithm stands out for selecting the fewest variables,

thereby indicating the least complexity. In the German dataset, the CFS + SA, SU + SA, and chi-square + SA sequential feature selection algorithms are noted for having the fewest variables.

Figures 8 and 9 illustrate the variables most frequently selected by the feature selection methods for both datasets, highlighting the most important variables identified in the analysis.

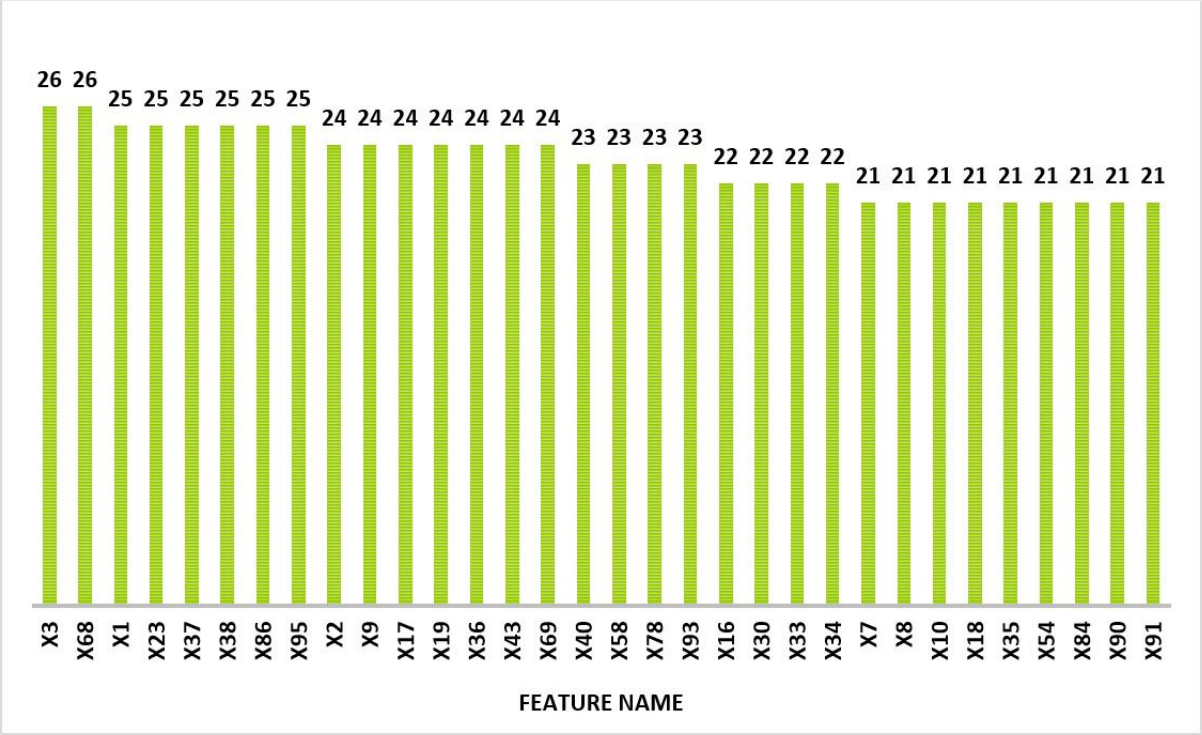

Figure 8. Most frequently selected variables with feature selection methods for the balanced (undersampling) Taiwanese dataset

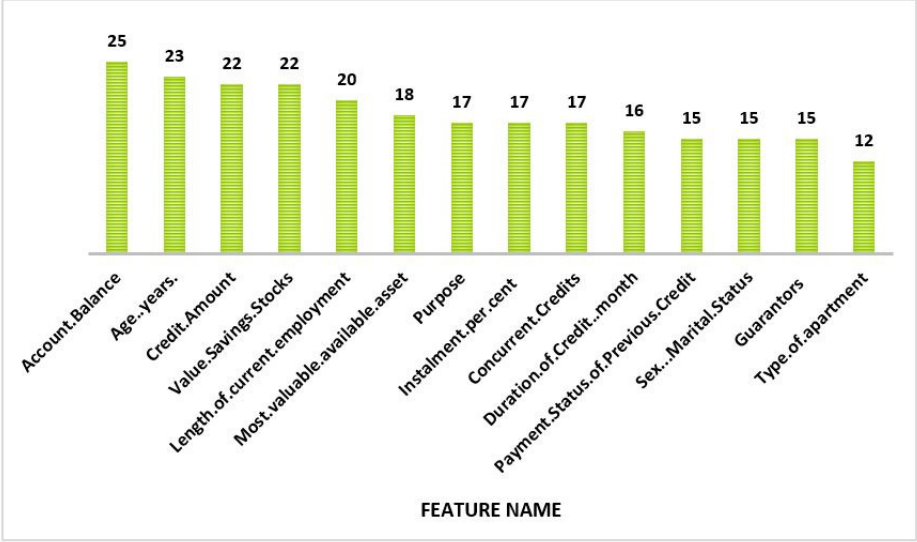

Figure 9. Most frequently selected variables with feature selection methods for the balanced (undersampling) German dataset

The performance of various algorithms was evaluated in the balanced dataset after applying different sequential feature selection methods.

Table 5. Results of machine learning algorithms after sequential feature selection for balanced Taiwanese Bankruptcy Prediction dataset (undersampling)

Algorithm	Feature selection method	Acc [%]	Sen [%]	Spe [%]
glm	no feature selection	82	86	77
	RELIEF + GA	93	93	93
knn	no feature selection	63	66	60
	CFS + SA	90	89	91
nb	no feature selection	83	100	66
	SU + RFE	91	91	91
svmLinear	no feature selection	48	52	43
	RELIEF + GA	92	89	95
svmPoly	no feature selection	61	57	66
	CFS + BORUTA	91	93	89
	RELIEF + GA	91	89	93
svmRadial	no feature selection	61	57	66
	GR + GA	90	86	93
lda	no feature selection	86	86	86
	RELIEF + GA	92	89	95
mlp	no feature selection	81	86	75
	CFS + SA	88	75	100
	chi-square + BORUTA	82	64	100
	SU + SA	88	96	79
evtree	no feature selection	89	86	91
	SU + BORUTA	91	89	93
rpart	no feature selection	85	91	80
	IG + GA	86	93	80
C5.0	no feature selection	89	91	86
	RELIEF + GA	91	89	93
j48	no feature selection	84	86	82
	SU + RFE	91	91	91
earth	no feature selection	85	82	89
	GR + GA	91	89	93
rf	no feature selection	80	77	82
	GR + GA	92	89	95
gbm	no feature selection	78	77	80
	GR + GA	91	86	95
xgbtree	no feature selection	77	75	80
	GR + GA	92	89	95
treebag	no feature selection	82	84	80
	GR + GA	92	89	95
adaboost	no feature selection	80	80	80
	RELIEF + SA	92	91	93
glmboost	no feature selection	83	82	84
	RELIEF + GA	91	89	93
LogitBoost	no feature selection	74	80	68
	GR + GA	89	89	89
	RELIEF + BORUTA	89	89	89
rotationForest	no feature selection	82	82	82
	SU + RFE	91	87	95

Table 6. Results of machine learning algorithms after sequential feature selection for balanced German Credit Dataset (undersampling)

Algorithm	Feature Selection Method	Acc [%]	Sen [%]	Spe [%]
glm	no feature selection	74	77	72
	SU + RFE	77	83	73
	IG + BORUTA	75	67	83
knn	no feature selection	57	63	50
	CFS + SA	69	80	58
	RELIEF + GA	65	68	62
nb	no feature selection	73	82	65
	SU + RFE	80	77	82
	GR + GA	63	42	85
	RELIEF + RFE	58	85	30
svmLinear	no feature selection	74	77	72
	IG + BORUTA	76	83	71
	SU + RFE	76	70	82
svmPoly	no feature selection			
	chi-square + RFE	77	68	85
	SU + RFE	74	85	67
svmRadial	no feature selection	78	78	77
	chi-square + RFE	77	70	83
	IG + SA	68	80	55
lda	no feature selection	73	77	70
	SU + RFE	77	83	73
	chi-square + RFE	74	67	82
mlp	no feature selection	50	100	0
	CFS + SA	69	80	58
	CFS + GA	69	80	58
evtree	no feature selection	71	75	67
	chi-square + RFE	73	73	72
	IG + RFE	70	65	75
	IG + SA	69	90	48
rpart	no feature selection	69	68	70
	IG + BORUTA	73	78	67
	GR + SA	63	43	82
	SU + BORUTA	62	87	45
C5.0	no feature selection	73	80	67
	IG + BORUTA	74	65	83
	IG + SA	71	88	53
j48	no feature selection	75	83	67
	IG + BORUTA	76	78	73
	chi-square + RFE	75	75	75
	IG + SA	69	90	48
earth	no feature selection	73	77	68
	SU + RFE	74	83	68
	IG + BORUTA	74	75	73
	IG + SA	69	90	48

Table 6. Results of machine learning algorithms after sequential feature selection for balanced German Credit Dataset (undersampling) (continued)

Algorithm	Feature Selection Method	Acc (%)	Sen (%)	Spe (%)
rf	no feature selection	78	80	77
	chi-square + RFE	74	70	78
	SU + RFE	72	83	64
gbm	no feature selection	74	82	67
	IG + BORUTA	76	73	78
xgbtree	no feature selection	76	77	75
	IG + RFE	75	70	80
	chi-square + BORUTA	73	82	65
treebag	no feature selection	77	80	73
	IG + RFE	70	62	78
	SU + RFE	72	81	66
adaboost	no feature selection	75	73	77
	chi-square + BORUTA	76	82	70
	chi-square + RFE	72	65	78
glmboost	no feature selection	73	77	68
	IG + BORUTA	76	68	83
	GR + BORUTA	75	77	73
	SU + RFE	78	83	74
LogitBoost	no feature selection	67	58	75
	chi-square + RFE	73	70	77
	RELİEF + GA	57	83	30
rotationForest	no feature selection	73	77	70
	IG + BORUTA	77	80	73
	IG + RFE	70	58	82
	SU + RFE	73	83	66

Tables 5 and 6 show the performance of each algorithm both without feature selection (labeled as no feature selection) and with the sequential feature selection method that yielded the best results. In general, it was observed that compared to scenarios with no feature selection, all sequential feature selection methods enhanced the performance of the algorithms.

Specifically, for the Taiwanese dataset, the generalized linear model (GLM) achieved the best performance following the RELIEF + GA sequential feature selection. In the case of the German dataset, the sequential feature selection method SU + RFE, combined with the naive Bayes (NB) algorithm, produced the most favorable outcomes.

4.3. Balanced (oversampling) dataset results

This section details the results obtained after balancing the datasets through oversampling. The Taiwanese dataset was balanced to include 13198 samples, with an equal distribution of 6599 bankrupt and 6599 non-bankrupt cases. Similarly, the German dataset was balanced with a total of 1400 samples, comprising 700 good and 700 bad credit cases.

In the analysis of the Taiwanese dataset post-oversampling, the IG + RFE algorithm was noted for selecting the fewest variables, thus indicating the least complexity. In the German dataset, the CFS

+ SA sequential feature selection algorithm was found to select the fewest variables. Figures 10 and 11 illustrate the variables most frequently identified by the feature selection methods in both datasets, thereby highlighting the most significant variables.

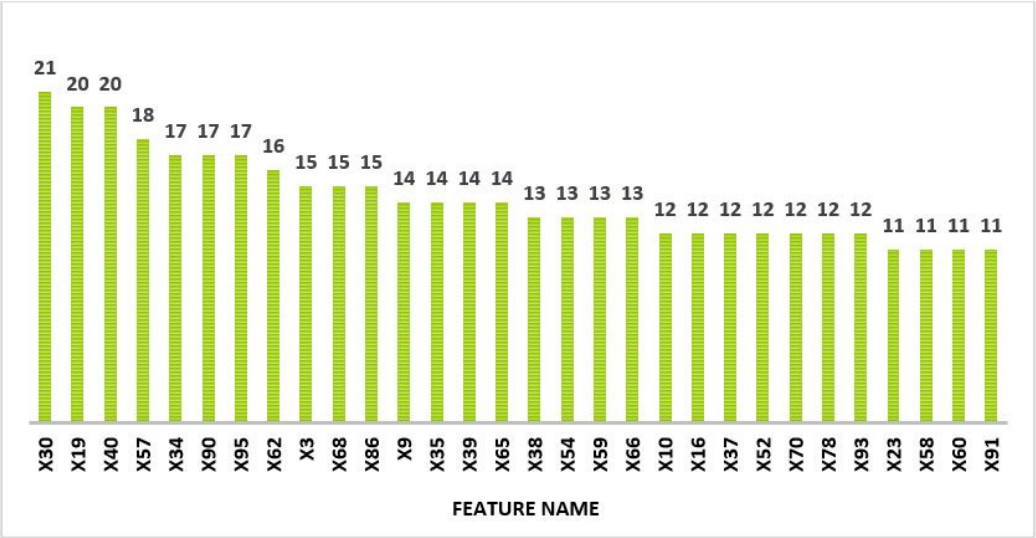

Figure 10. Most frequently selected variables with feature selection methods for the balanced (oversampling) Taiwanese dataset

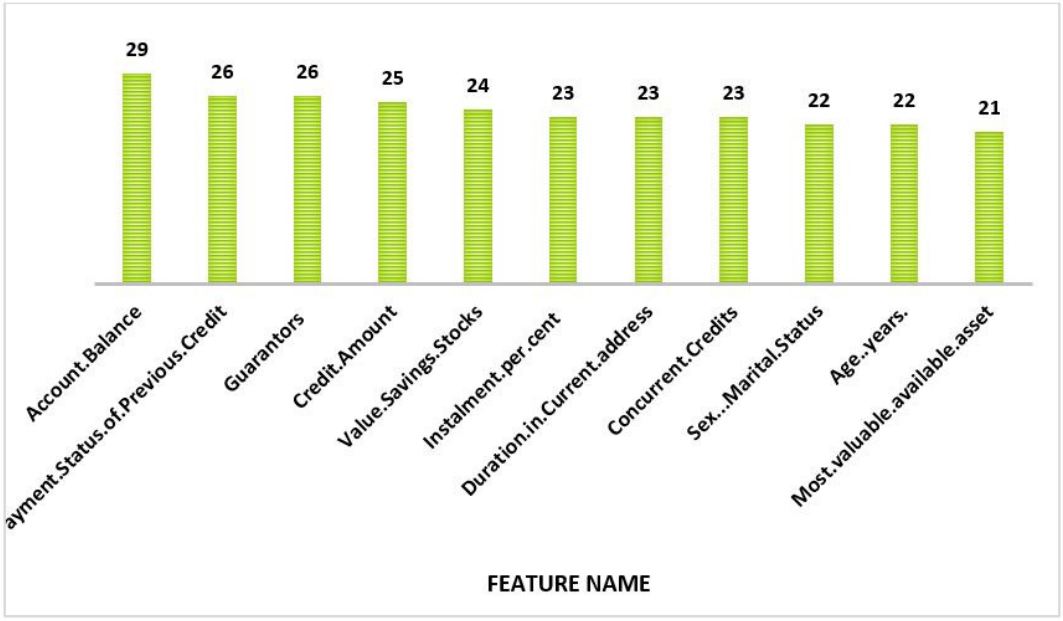

Figure 11. Most frequently selected variables with feature selection methods for the balanced (oversampling) German dataset

The performance of various algorithms in the balanced dataset, post-oversampling, was evaluated after applying different sequential feature selection methods. Tables 7 and 8 provide insights into the performance of each algorithm both without feature selection (labelled as "no feature selection") and with the sequential feature selection method that yielded the best outcomes. Generally, it was found that, in comparison to scenarios with no feature selection, all sequential feature selection methods led to improved algorithm performance.

Table 7. Results of machine learning algorithms after sequential feature selection for balanced Taiwanese bankruptcy prediction dataset (oversampling)

Algorithm	Feature selection method	Acc [%]	Sen [%]	Spe [%]
glm	no feature selection	74	77	72
	SU + RFE	77	83	73
knn	no feature selection	93	87	100
	RELIEF + SA	96	92	100
	SU + RFE	96	92	100
	SU+BORUTA	96	92	100
nb	no feature selection	72	51	94
	IG+BORUTA	96	99	11
	GR+BORUTA	96	98	27
	CFS + GA	61	99	23
svmRadial	no feature selection	81	79	84
	IG+BORUTA	97	100	0
	GR + GA	97	100	0
	GR + SA	97	100	0
	GR+BORUTA	97	100	0
	SU + RFE	94	91	97
mlp	no feature selection	59	78	40
	IG+BORUTA	97	100	0
	GR + GA	97	100	0
	GR + SA	97	100	0
	GR+BORUTA	97	100	0
	GR + RFE	83	74	92
rpart	no feature selection	85	88	83
	IG+BORUTA	97	100	10
	GR + SA	97	100	10
	GR+BORUTA	97	100	10
	chi-square + SA	83	72	95
C5.0	no feature selection	99	99	100
	chi-square + RFE	99	99	100
	GR+BORUTA	97	100	20
j48	no feature selection	99	97	100
	SU + GA	99	97	100
	GR + SA	97	100	10
earth	no feature selection	89	88	90
	chi-square+BORUTA	98	96	100
	GR + SA	96	99	11
rf	no feature selection	100	99	100
	RELİEF + GA	100	99	100
	GR + SA	97	100	18
gbm	no feature selection	96	92	100
	SU + RFE	95	92	99
	GR+BORUTA	97	99	20
xgbtree	no feature selection	99	98	100
	SU + GA	99	98	100

Table 7. Results of machine learning algorithms after sequential feature selection for balanced Taiwanese bankruptcy prediction dataset (oversampling) (continued)

treebag	no feature selection	99	97	100
	SU + SA	99	99	100
adaboost	no feature selection	99	99	100
	SU + SA	100	100	100
glmboost	no feature selection	87	86	89
	IG + RFE	97	99	0
LogitBoost	no feature selection	93	89	97
	GR + GA	96	99	18
	IG+BORUTA	96	99	32
rotationForest	no feature selection	93	89	97
	GR+BORUTA	97	99	20

Table 8. Results of machine learning algorithms after sequential feature selection for balanced German Credit Dataset (oversampling)

Algorithm	Feature selection method	Acc [%]	Sen [%]	Spe [%]
glm	no feature selection	73	69	76
	IG + RFE	75	72	78
	RELIEF +BORUTA	71	76	67
knn	no feature selection	66	54	77
	IG + SA	71	65	78
	CFS + RFE	68	58	79
nb	no feature selection	75	62	87
	GR + RFE	75	79	70
	SU + SA	62	33	87
svmLinear	no feature selection	73	69	78
	chi-square + RFE	73	71	75
	CFS + GA	68	54	81
	RELIEF + SA	65	75	55
svmPoly	no feature selection	84	77	90
	GR + SA	67	77	56
svmRadial	no feature selection	78	75	80
	RELIEF +BORUTA	78	78	78
	CFS + GA	68	53	84
lda	no feature selection	72	68	76
	IG + RFE	75	71	78
	IG+BORUTA	72	72	72
mlp	no feature selection	50	0	100
	CFS + GA	69	70	68
evtree	no feature selection	76	71	81
	chi-square+BORUTA	75	68	82
	chi-square + RFE	75	74	75
rpart	no feature selection	72	66	78
	GR + SA	61	95	28

Table 8. Results of machine learning algorithms after sequential feature selection for balanced German Credit Dataset (oversampling) (continued)

Algorithm	Feature selection method	Acc [%]	Sen [%]	Spe [%]
C5.0	no feature selection	86	79	93
	SU+BORUTA	90	86	93
	chi-square + GA	89	85	94
	RELIEF + GA	88	86	89
j48	no feature selection	82	76	89
	SU+BORUTA	85	77	93
	chi-square + RFE	83	71	95
earth	no feature selection	74	72	76
	IG + RFE	75	71	79
	CFS + RFE	68	54	81
	RELIEF + GA	74	76	71
rf	no feature selection	88	81	94
	chi-square + SA	90	86	94
	IG+BORUTA	89	81	96
gbm	no feature selection	77	76	78
	RELIEF +BORUTA	88	80	96
	SU+BORUTA	84	81	86
xgbtree	no feature selection	85	78	93
	SU+BORUTA	88	82	93
	IG+BORUTA	87	79	95
treebag	no feature selection	87	80	94
	SU+BORUTA	89	81	95
	IG+BORUTA	87	77	96
	IG + SA	87	77	96
	RELIEF + RFE	87	83	91
adaboost	no feature selection	89	86	93
	GR+BORUTA	90	86	93
	SU+BORUTA	90	88	91
	IG + SA	89	83	96
	chi-square + RFE	90	88	92
glmboost	no feature selection	73	69	76
	IG + RFE	74	70	79
	RELIEF +BORUTA	71	74	69
LogitBoost	no feature selection	70	69	71
	SU + GA	73	70	76
	RELIEF +BORUTA	72	57	87
	IG + RFE	71	77	66
rotationForest	no feature selection	78	74	82
	CFS + RFE	66	48	85
	chi-square + RFE	77	74	79

Specifically, in the Taiwanese dataset, the ADABOOST algorithm, following the SU + SA sequential feature selection, achieved the most favorable results. In the German dataset, the algorithm that demon-

strated both the fewest variables and the best performance was the random forest (RF), subsequent to chi-square + SA sequential feature selection.

5. Discussion

5.1. The impact of feature selection on classifier performance

When selecting a model, preference is given to algorithms that showcase the highest performance values, minimal complexity, and the shortest calculation time. A dataset with fewer variables typically exhibits less complexity. Feature selection improves data quality by removing unnecessary, irrelevant, or noisy data. This process not only mitigates the risk of overfitting but also enhances the overall performance of the models. Post sequential feature selection, a noticeable improvement in the performance of the algorithms was observed in both datasets.

5.2. Optimal combinations of feature selection methods and prediction models

In both datasets, the Symmetric Uncertainty (SU) algorithm, employed as the filter method, demonstrated high performance. As for wrapper methods, the Genetic Algorithm (GA) and Simulated Annealing (SA) algorithms showed the best performance. Specifically, the SU + SA combination yielded the best results in the Taiwanese dataset, whereas the chi-square + RFE and chi-square + SA combinations excelled in the German dataset. Moreover, the correlation-based feature selection (CFS) as a filtering method and SA as a wrapper method generally resulted in the fewest variables.

5.3. Evaluating the impact of feature selection on model performance and complexity

In Table 9, the algorithms that yield the best results are presented. Upon reviewing these results, it becomes evident that heuristic approaches, commonly utilized as wrapper methods in feature selection, not only reduce the number of variables but also enhance performance. Sequential feature selection effectively improves the performance of machine learning algorithms while simultaneously reducing both the model's complexity and calculation time. The algorithm that gives the best results on the Taiwanese dataset is the AdaBoost algorithm after SU + SA sequential feature selection after oversampling. The algorithms that give the best results in the Statlog data set are the AdaBoost algorithm after chi-square + RFE sequential feature selection after oversampling and the random forest (RF) algorithm after chi-square + SA sequential feature selection.

Table 9. Best resulting algorithms for datasets

Data Sets	Algorithm	Feature selection method	Acc [%]	Sen [%]	Spe [%]
Taiwanese Bankruptcy Prediction (oversampling)	ADABOOST	SU + SA	100	100	100
Statlog (German Credit data) (oversampling)	ADABOOST	chi-square + RFE	90	88	92
	RF	chi-square + SA	90	86	94

6. Conclusion

This study demonstrates the efficacy of hybrid feature selection methods, combining both filter and wrapper approaches, in enhancing the performance of machine learning algorithms. Initially, filter methods were employed for their computational efficiency to select features from the original dataset. Subsequently, wrapper methods further refined these features, leading to the final feature set used for classification. Notably, sequential feature selection was instrumental in improving algorithm performance while concurrently reducing model complexity and calculation time.

A key insight from this research is the inadequacy of relying solely on accuracy as a performance metric in unbalanced datasets. This study highlights the importance of considering both dataset balancing techniques and alternative performance metrics, such as specificity and sensitivity, to attain a more comprehensive evaluation of model performance. In datasets like the Taiwanese Bankruptcy and Statlog (German Credit Data), where class distribution was initially unbalanced, applying undersampling and oversampling significantly impacted the algorithms' performance.

Additionally, the study makes a significant contribution by demonstrating that heuristic approaches, particularly when used as wrapper methods in feature selection, effectively reduce the number of variables while enhancing overall performance. This was evident in the superior results achieved by algorithms such as SU + SA, chi-square + RFE, and ADABOOST, following sequential feature selection in both the Taiwanese and German datasets.

The research further establishes that in dealing with complex, real-world datasets, both the choice of feature selection methods and the strategy for handling data unbalances are crucial for optimizing machine learning models. These findings lay a foundation for future studies, which will aim to explore various approaches to balance unbalanced datasets and experiment with different heuristic wrapper methods, offering potential for deeper insights and further advancements in the field of machine learning.

In conclusion, this study not only enhances the understanding of feature selection in machine learning but also provides practical solutions for effectively managing unbalanced datasets, thereby contributing significantly to both the theoretical and practical aspects of machine learning research. It is meaningless and insufficient to evaluate the performance of the model by looking at the accuracy value in unbalanced data sets. When dealing with unbalanced data sets, it is crucial to focus on balancing the data set and selecting the appropriate metric. In future studies, it is aimed to try different approaches to balance the unbalanced data set and also to try different heuristic approaches, such as the wrapper method.

Acknowledgement

The authors are grateful to two anonymous reviewers for their valuable comments and suggestions made on the previous draft of this manuscript. This work has not been supported by any funds.

References

[1] Almeida, S. Exploring the impact of regularization to improve bankruptcy prediction for corporations. In *2023 World Conference on Communication & Computing (WCONF)* (RAIPUR, India, 2023), IEEE, pp. 1–5.

[2] Breiman, L. Bagging predictors. *Machine learning 24*, 2 (1996), 123–140.

[3] Breiman, L. Random forests. *Machine learning 45*, 1 (2001), 5–32.

[4] Breiman, L., Friedman, J., Olshen, R. A., and Stone, C. J. *Classification and Regression Trees*. Wadsworth, 1984.

[5] Brenes, R. F., Johannssen, A., and Chukhrova, N. An intelligent bankruptcy prediction model using a multilayer perceptron. *Intelligent Systems with Applications 16* (2022), 200136.

[6] Chen, T., and 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, 2016), pp. 785–794.

[7] Cho, S. H., and Shin, K.-s. Feature-weighted counterfactual-based explanation for bankruptcy prediction. *Expert Systems with Applications 216* (2023), 119390.

[8] Friedman, J. H. Multivariate adaptive regression splines. *The Annals of Statistics 19*, 1 (1991), 1–67.

[9] Grubinger, T., Zeileis, A., and Pfeiffer, K.-P. evtree: Evolutionary learning of globally optimal classification and regression trees in R. *Journal of Statistical Software 61*, 1 (2014), 1–29.

[10] Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. Gene selection for cancer classification using support vector machines. *Machine Learning 46*, 1-3 (2002), 389–422.

[11] Hall, M. *Correlation-based Feature Selection for Machine Learning*. PhD thesis, The University of Waikato, Hamilton, 1999.

[12] Hussain, M., Wajid, S. K., Elzaart, A., and Berbar, M. A comparison of SVM kernel functions for breast cancer detection. In *2011 Eight International Conference Computer Graphics, Imaging and Visualization* (Singapure, 2011), IEEE, pp. 145–150.

[13] James, G., Witten, D., Hastie, T., and Tibshirani, R. *An Introduction to Statistical Learning*. Springer, New York, 2013.

[14] Khemka, D., Kaippada, R., Nikhil, P. S., and Suseela, S. Machine learning based efficient bankruptcy prediction model. In *2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE)* (Chennai, India, 2023), IEEE, pp. 1–10.

[15] Kira, K., and Rendell, L. A. The feature selection problem: traditional methods and a new algorithm. In *AAAI'92: Proceedings of the tenth national conference on Artificial intelligence* (1992), AAAI Press, pp. 129–134.

[16] Kursa, M. B., Jankowski, A., and Rudnicki, W. R. Boruta - a system for feature selection. *Fundamenta Informaticae 101*, 4 (2010), 271–285.

[17] Liang, D., Tsai, C.-F., and Wu, H.-T. The effect of feature selection on financial distress prediction. *Knowledge-Based Systems 73* (2015), 289–297.

[18] Lin, S.-W., Shiue, Y.-R., Chen, S.-C., and Cheng, H.-M. Applying enhanced data mining approaches in predicting bank performance: A case of Taiwanese commercial banks. *Expert Systems with Applications 36*, 9 (2009), 11543–11551.

[19] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics 21*, 6 (1953), 1087–1092.

[20] Mitchell, T. M. *Machine Learning*. MacGraw-Hill, USA, 1997.

[21] Myles, A. J., Feudale, R. N., Liu, Y., Woody, N. A., and Brown, S. D. An introduction to decision tree modeling. *Journal of Chemometrics 18*, 6 (2004), 275–285.

[22] Naganna S. R., and Deka, P. C. Support vector machine applications in the field of hydrology: A review. *Applied Soft Computing 19* (2014), 372–386.

[23] Nelder, J. A., and Wedderburn, R. W. M. Generalized linear models. *Journal of the Royal Statistical Society. Series A: Statistics in Society 135*, 3 (1972), 370–384.

[24] Novaković, J. Toward optimal feature selection using ranking methods and classification algorithms. *Yugoslav Journal of Operations Research 21*, 1 (2016), 119–135.

[25] Patil, T. R., and Sherekar, S. S. Performance analysis of naive bayes and J48 classification algorithm for data classification. *International Journal of Computer Science and Applications 6*, 2 (2013), 256–261.

[26] Quan, J., and Sun, X. Credit risk assessment using the factorization machine model with feature interactions. *Humanities and Social Sciences Communications 11*, 1 (2024), 234.

[27] Quinlan, J. R. Improved use of continuous attributes in C4.5. *Journal of Artificial Intelligence Research 4* (1996), 77–90.

[28] Rish, I. An empirical study of the naive bayes classifier. In *Proceedings of the IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence* (Seattle, WA, USA, 2001), pp. 41–46.

[29] Rodriguez, J. J., Kuncheva, L. I., and Alonso, C. J. Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 10 (2006), 1619–1630.

[30] Saeys, Y., Inza, I., and Larrañaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics 23*, 19 (2007), 2507–2517.

[31] Schapire, R. E. The strength of weak learnability. *Machine Learning 5*, 2 (1990), 197–227.

[32] Subanya, B., and Rajalaxmi, R. R. Feature selection using artificial bee colony for cardiovascular disease classification. In *2014 International Conference on Electronics and Communication Systems (ICECS)* (Coimbatore, India 2014), IEEE, pp. 1–6.

[33] Tsai, C.-F. Combining cluster analysis with classifier ensembles to predict financial distress. *Information Fusion 16* (2014), 46–58.

[34] Tsai, C.-F., Hsu, Y. F., and Yen, D. C. A comparative study of classifier ensembles for bankruptcy prediction. *Applied Soft Computing 24* (2014), 977–984.

[35] Vapnik, V. N. *Statistical Learning Theory*. John Wiley & Sons, Inc., 1998.

[36] Verma, A., and Mehta, S. A comparative study of ensemble learning methods for classification in bioinformatics. In *2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence* (Noida, India, 2017), pp. 155–158.

[37] Wang, Z., Wang, Y., and Srinivasan, R. S. A novel ensemble learning approach to support building energy use prediction. *Energy and Buildings 159* (2018), 109–122.

[38] Welikala, R. A., Fraz, M. M., Dehmeshki, J., Hoppe, A., Tah, V., Mann, S., Williamson, T. H., and Barman, S. A. Genetic algorithm based feature selection combined with dual classification for the automated detection of proliferative diabetic retinopathy. *Computerized Medical Imaging and Graphics 43* (2015), 64–77.

[39] Werbos, P. *Beyond regression: New tools for prediction and analysis in the behavioral sciences*. PhD thesis, Harvard University, Cambridge, MA, 1974.

[40] Xiao, H., Xiao, Z., and Wang, Y. Ensemble classification based on supervised clustering for credit scoring. *Applied Soft Computing 43* (2016), 73–86.

[41] Xie, J., and Wang, C. Using support vector machines with a novel hybrid feature selection method for diagnosis of erythemato-squamous diseases. *Expert Systems with Applications 38*, 5 (2011), 5809–5815.

[42] Youness, G., Phan, N. U. T., and Boulakia, B. C. BootBOGS: Hands-on optimizing Grid Search in hyperparameter tuning of MLP. In *AICCSA 2023: 20th ACS/IEEE International Conference on Computer Systems and Applications* (2023).