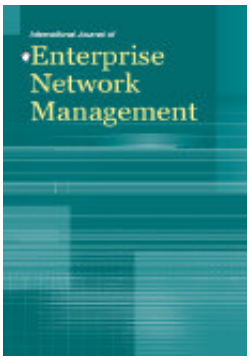


International Journal of Enterprise Network Management

ISSN online: 1748-1260 - ISSN print: 1748-1252
https://www.inderscience.com/ijenm

Ensemble classifiers for bankruptcy prediction using SMOTE and RFECV

T. Shahana, Vilvanathan Lavanya, Aamir Rashid Bhat

DOI: 10.1504/IJENM.2024.10058997

Article History:

Received:	30 March 2022
Last revised:	13 March 2023
Accepted:	06 July 2023
Published online:	19 March 2024

Ensemble classifiers for bankruptcy prediction using SMOTE and RFECV

T. Shahana* and Vilvanathan Lavanya

Department of Management Studies,
National Institute of Technology Tiruchirappalli,
Thuvakkudi, Trichy, 620015, India
Email: shahanadms@gmail.com
Email: lavanya@nitt.edu
*Corresponding author

Aamir Rashid Bhat

Department of Corporate Secretaryship Accounting and Finance,
SRM Institute of Science and Technology,
Kattankulathur, Chennai, 603203, India
Email: aamir.bhat8@gmail.com

Abstract: This research investigates the impact of preprocessing strategies, namely feature selection (utilising correlation and recursive feature elimination with cross-validation) and class imbalance handling (employing synthetic minority oversampling technique), on the performance of prediction models using ensemble-learning techniques (random forest, AdaBoost, gradient boosting decision tree, extreme gradient boosting, bagging, LightGBM and extra tree classifier). The study focuses on the Polish bankruptcy dataset to assess the effectiveness of these preprocessing approaches. Experimental results demonstrate that adopting class imbalance handling significantly influences classifier performance compared to feature selection alone. Interestingly, hyperparameter tuning and feature selection exhibit limited impact on classifier performance. Among the ensemble-learning techniques tested, the adaptive boosting classifier shows consistently poor performance throughout the study period, followed by the bagging classifier with statistical significance. These findings shed light on the importance of selecting appropriate preprocessing strategies to improve the performance of ensemble-based prediction models in bankruptcy prediction tasks.

Keywords: bankruptcy prediction; ensemble classifiers; missing value imputation; SMOTE; correlation; RFECV.

Reference to this paper should be made as follows: Shahana, T., Lavanya, V. and Bhat, A.R. (2024) 'Ensemble classifiers for bankruptcy prediction using SMOTE and RFECV', *Int. J. Enterprise Network Management*, Vol. 15, No. 1, pp.109–132.

Biographical notes: T. Shahana is a PhD scholar in the Department of Management Studies, National Institute of Technology Tiruchirappalli (NITT), Trichy. Her academic journey includes completing an MBA program at Kannur University. Following her MBA, she dedicated three years to serving as an Assistant Professor at Kannur University. Her areas of research focus encompass a range of topics, including accounting fraud, bankruptcy, systematic reviews, and financial inclusion.

Vilvanathan Lavanya has persistently demonstrated herself as an educator and administrator with consistent success. She has over 15 years of academic experience and extensive academic background. Her passion for teaching has made her contribute the finest to the teaching community by unveiling ground-breaking teaching and learning techniques. Her areas of specialisation include; organisational structure and design, human resource management, and training and development. Her research expertise has made her publish several research papers in referred journals.

Aamir Rashid Bhat is an Assistant Professor in the Department of Corporate Secretaryship and Accounting and Finance, SRM Institute of Science and Technology, Kattankulathur, Chennai. He has completed his PhD and Master's degrees from Pondicherry University, Puducherry, and has more than two years of teaching experience. He specialises in finance, and his areas of interest include financial inclusion, corporate social responsibility, bankruptcy, systematic reviews, and bibliometric analysis.

This paper is a revised and expanded version of a paper entitled 'Financial distress prediction: an empirical study' presented at the BMA Conference 2022, Department of Management Studies, NIT Tiruchirappalli, 28 January 2022.

1 Introduction

Researchers and academicians have used several definitions of 'corporate bankruptcy' in their work. A company's inability to generate enough profits to pay off its debtors in terms of interest and the principal sum is known as corporate bankruptcy (Gordon, 1971). Numerous examples of these occurrences we have occasionally seen are Enron's bankruptcy in 2001, WorldCom's in 2002, and Lehman Brothers' in 2008. The topic has attracted the research community because of the significant consequences it could have for business and society at large. Corporate bankruptcy research falls into two categories:

1 predicting bankruptcy (Altman et al., 2017)

2 probing the determinants of bankruptcy (Lukason and Laitinen, 2019).

Financial institutions, investors, and rating agencies have long made bankruptcy prediction a priority research subject, and many statistical and machine-learning models have been developed for predicting the event of bankruptcy.

When it comes to using statistics to foresee business failure, Beaver (1966) was the pioneer. Altman (1968), Wilcox (1973), Altman and Bettina (1976), Deakin (1972) and Laitinen (1991) all followed Beaver (1966). On Beaver's recommendation, Altman (1968) harnessed multiple discriminant analysis (MDA), the gold standard in statistical analysis, to create a bankruptcy prediction problem (BPP, also known as the Z model). Machine learning (ML) has come a long way in insolvency forecasting since the 1990s. Regardless of the approach to their creation, prediction models must be as efficient as possible. Hence, the main research goal when making a prediction model has been to find the model that makes the most accurate predictions or has the slightest prediction error (Tsai et al., 2021). A standard formulation of the BPP asks, "given a set of financial variables that describe the situation of a company over a given period, and a set of companies that have been labelled bankrupt or healthy, predict the likelihood that the

company may become bankrupt during the following years" (Chen et al., 2020). BPP is typically solved using a binary classification task.

Several essential variables impact the ultimate efficacy of prediction models. Feature selection is critical among the many variables contributing to a model's overall predictive efficacy. Improving a model's prediction accuracy often requires performing feature selection to evaluate the feature representativeness of the gathered datasets. An example is the research conducted by Lin et al. (2019), which examined the relative merits of several feature selection and ensemble categorisation strategies for bankruptcy forecasting. Class imbalance handling is another critical consideration while making a real-world application of the insolvency prediction problem. In a case of class inequality, the number of prosperous businesses vastly outnumbers the number of unsuccessful ones. The ability of a model to predict bankruptcy is disturbed when it uses a skewed or imbalanced dataset (Zelenkov and Volodarskiy, 2021). Many classification algorithms used in machine learning presume symmetry between class distributions as part of their objective function, significantly contributing to the performance degradation caused by data imbalance problems (Kim et al., 2016).

Numerous machine learning and computational intelligence techniques, including artificial neural networks (ANN), support vector machines (SVM), decision trees (DT), and many others, have recently been recommended to address BPP (Adisa et al., 2019). This paper focuses on BPP because, regardless of substantial research efforts, sophisticated predictive models are rarely used in practice (Bellovary et al., 2007). According to recent studies, ensemble frameworks which combine multiple classifiers (or prediction models) are a promising method for predicting bankruptcy that can help traditional models overcome flaws like multivariate normality, multicollinearity, and worsened correct classification rates while also giving banks and other financial institutions a reliable model for predicting business failure. For instance, extreme gradient boosting (XGB) was posited by Zięba et al. (2016) to resolve the issue of BPP. They found that the XGB classifier is noticeably more accurate than any previous reference methods to identify the businesses' financial state.

This study seeks to explain whether or not preprocessing the data with ensemble-learning techniques, including feature selection and class imbalance handling, yields superior results to either using just one of these methods or combining any of these methods. Hence, we compared the performance of the seven most common ensemble classifiers – bagging, adaptive boosting (AdaBoost), random forest (RF), gradient boosting decision tree (GBDT), extreme gradient boosting (XGBoost), extra tree classifier (ETC), and light gradient boosting (LightGBM) – in the current study in three distinct ways:

a with feature selection and class imbalance handling

b without class imbalance handling and with feature selection

c with class imbalance handling and without feature selection.

We used synthetic minority oversampling technique (SMOTE) as a class imbalance handling strategy and correlation and recursive feature elimination with cross-validation (RFECV) for feature selection. To the author's knowledge, the ETC and LightGBM are applied for the first time to bankruptcy prediction using a real-world Polish dataset.

The rest of the research is structured as follows: The literature review is discussed in Section 2, and a broad overview of the study methods is provided in Section 3. In Section 4, we looked at the results and talked about what we learned. The significant findings and caveats of the research are presented in Section 5.

2 Literature review

2.1 Traditional ML algorithms for BPP

In this section, we give a literature review on BPP, emphasising the application of machine learning. Numerous papers have been written over the past few decades to resolve the issue of bankruptcy prediction. Much recent work investigates the BPP through supervised learning, such as K-nearest neighbour (KNN), DT, SVM, and NN (Ocal et al., 2015; Hosaka, 2019). The classification methods used by Bateni and Asghari (2020) included logit analysis and the genetic algorithm (GA). Their research on Tehran Stock Exchange-listed Iranian businesses discovered that the GA model was supercilious to the logit classifier. To create a robust bankruptcy prediction model that takes into account the feature extraction process, Smiti and Soui (2020) combined the borderline synthetic minority (BSM) oversampling method with the stacked auto encoder (SAE) formulated on the softmax classifier. According to their experimental findings, the BSM-SAES is superior to the other methods regarding area under the curve (AUC).

Ocal et al. (2015) suggested chi-square automatic interaction detector (CHAID) and C5.0 decision tree algorithms to predict business failure. The findings obtained demonstrate the proposed models' sufficient prediction accuracy. Santoso and Wibowo (2018) used linear discriminant analysis (LDA) and SVM with a dimension reduction method to determine how likely Indonesian financial companies would go bankrupt. The results showed that, compared to other classifiers, the hybrid stepwise-SVM model gave an improved performance on the accuracy metric. Using a real-world dataset of Polish companies, Fan et al. (2017) proposed and appraised the performance of one-class SVM, isolation forest, and multivariate Gaussian distribution algorithms for predicting insolvency. When comparing different classifiers based on the AUC measure, isolation forest outperformed the remaining classifiers. Positive results show that the isolation forest can effectively mitigate imbalanced learning issues.

2.2 Ensemble learning algorithms for BPP

The study of BPP has lately benefited from applying a wide variety of ensemble methods, all of which aim to improve BPP performance by pooling the knowledge of many individual learners into a more formidable whole. In order to address the BPP on monetary variables, Jabeur et al. (2021) created a brand new advanced ML algorithm (CatBoost) and compared its performance to that of eight other commonly used algorithms. The findings demonstrate that by using the CatBoost method, the prediction performance was greatly enhanced. Based on experimental findings, Matin et al. (2019) concluded that XGBoost is the best BPP model among ensemble-based BPP approaches for processing unstructured data like audit and management reports of firms. According to the findings of Le et al. (2018), RF coupled with a two-phase preliminary processing method consisting of an oversampling technique (SMOTE) and an information-cleansing

approach (edited nearest neighbour) produces the best results on a highly skewed dataset of Korean companies.

Using a dataset with 449 bankrupt and 449 solvent North American businesses, (Barboza et al., 2017) implemented and evaluated several classification models for bankruptcy prediction. These models included SVM with linear and radial basis functions, ANN, logistic regression, boosting, RF, and bagging. Experiments in this study demonstrate that boosting, bagging, and RF are superior classifiers for bankruptcy prediction. Zięba et al. (2016) proposed XGB to forecast financial distress using a real-world dataset of Polish businesses. The research shows that XGB has accurately predicted corporate insolvency compared to other current methodologies. Further, Wang et al. (2018) examined the attribute bagging (random subspace) method, an ensemble approach to extracting sentiment and textual information to predict the failure of businesses. Using AdaBoost within the context of the boosting framework, Alfaro et al. (2008) predicted the corporate bankruptcy risk of European companies. They demonstrated that, compared to ANN-based BPP models, the generalisation error could be reduced by around 30% using the proposed boosting ensemble method.

2.3 Class imbalance and BPP

A simple definition of the class imbalance problem in binary classification would be that there is a discrepancy between the distributions of the data from the positive and negative classes. In cases of business failure, the group of bankrupt firms (positive class) is much more prominent in number than the group of non-bankrupt firms (negative class). Since it is more costly to categorise a bankrupt business than a non-bankrupt one incorrectly, BPP tasks are of more significant concern to the minority (positive) class. Sampling techniques and cost-sensitive solutions are two ways to accomplish unbalanced BPP (Zou et al., 2022). The former is a data-level approach, and the latter is an algorithmic approach.

Sun et al. (2020) utilised SMOTE in conjunction with an ensemble framework to achieve an imbalanced BPP. Le et al. (2018) offered reference values for fraud detection and other areas by combining a clustering-based under-sampling strategy with a boosting ensemble framework for BPP. To address the class imbalance issue, Zou et al. (2022) modified XGBoost into a weighted variant called XGBoost-W. This change turned XGBoost's error-minimisation-based pattern into a cost-sensitive one. According to experimental findings, compared to traditional balanced BPP approaches such as LDA and LR, GBDT, LightGBM and XGBoost, cost-sensitive BPP models substantially lowered the erroneous classification rate of insolvent firms.

2.4 Feature selection and BPP

Since there is no universally accepted collection of financial ratios to use as input features for model building, feature selection has become the topic of multiple data mining studies preceding the development of models (Liang et al., 2015). On the contrary, using an excessive number of features to analyse the dataset can lead to problems with high dimensionality. Feature selection or dimensionality reduction can be used in data mining to eliminate irrelevant or duplicate features (Powell, 2009). Several methods for choosing which features to use have been suggested. These methods have three classifications:

filter, wrapper and embedded (Lin et al., 2019). Tsai (2009) surveyed and found that BPP literature needs more attention to feature selection pre-processing. Many studies focus on making prediction models that are more accurate and can make better predictions. Some do not even consider how to choose features before building their models. In contrast, Zelenkov et al. (2017) combined GA (genetic algorithm) as a tool for feature selection with different classical and ensemble classifiers, while Yu et al. (2014) used LDA to choose the best attributes while modelling ensemble algorithms.

Comparing the efficacy of classifiers based on machine learning and statistical approaches is a popular topic of scientific study. Nevertheless, more research directly comparing multiple ensemble classifiers is needed. To the author's understanding, efforts have yet to be made to compare the efficacy of different ensemble classifiers using a two-stage pre-processing approach that takes care of class imbalance and selects features. In order to address this research gap, this paper analyses the seven ensemble classifiers mentioned above in conjunction with a two-stage pre-processing approach comprising an over-sampling technique (SMOTE) and a feature selection technique (correlation, RFECV).

3 Research methodology and dataset

3.1 Dataset

We conducted our experiments with real data taken from the public domain. Five real-world datasets from Polish companies are included in the dataset (Zięba et al., 2016). These datasets were found at the University of California, Irvine (UCI) Machine Learning Archive[1]. In Table 1, summaries of the dataset are displayed[2]. There are 43,405 samples in the study, 2,091 of which are insolvent businesses and 41,314 are not bankrupt. From 2000 to 2012, the defunct companies were analysed, and the operating companies were studied between 2007 and 2013. Each observation contains 64 financial ratios (Zięba et al., 2016).

Table 1 The distribution of the dataset used in the experiment

Dataset	*Bankrupt*		*Non-bankrupt*		*Total*
	Number	*Percentage*	*Number*	*Percentage*	
Year 1	271	3.86	6,756	96.14	7,027
Year 2	400	3.94	9,773	96.06	10,173
Year 3	495	4.71	10,008	95.29	10,503
Year 4	515	5.26	9,277	94.74	9,792
Year 5	410	6.94	5,500	93.06	5,910

3.2 Modelling methods

In this study, we use the most popular ML ensemble models to compare the efficacy of seven different approaches to the problem of bankruptcy prediction for Polish companies: bagging, RF, AdaBoost, GBDT, LightGBM, XGBoost and ETC.

3.2.1 Bagging

Bagging, also known as bootstrap aggregation, is a meta-algorithm for ensemble learning that combines multiple classifiers to increase stability and accuracy while reducing the likelihood of over-fitting (Chen et al., 2020). The differences between the classifiers are minimised in the composite model, making it more accurate than its separate components (Han et al., 2012). Using a method dubbed 'row sampling with replacement', multiple classifiers can be trained in parallel using a random subset of the entire training dataset. Bootstrapping is the proper term for this method. Each model created in the 'bootstrapping' procedure predicts a different category of observations. At last, a 'voting classifier' combines all the individual model predictions into a single, accurate forecast. Each primary classifier in the model has its own unique training set. However, some items may overlap in the different training datasets generated for the different classifiers (Breiman, 1996).

3.2.2 Adaptive boosting

AdaBoost is a recursive ML ensemble algorithm that sequentially combines its baseline algorithms via boosting, a technique that incorporates various 'weak' learners into a single 'strong' learner (Freund and Schapire, 1996). Weights are initially uniform across samples in the first iteration of the model. However, they are subsequently adjusted to be higher for incorrectly classified instances and lower for rightly classified instances as the model iterates toward optimal performance. The end outcomes are a tally of all the predictions made by the ensemble classifiers (Kim and Kang, 2010).

3.2.3 Random forest

Breiman (2001) initially demonstrated RF, a robust supervised ML algorithm for classification and regression. RF combines multiple decision tree classifiers to classify an input vector, each casting a single vote for the most prevalent class. Each tree is built from a random vector drawn independently from the source vector (Breiman, 2001). They are an up-and-coming ensemble technique that combines random subset bagging of predictor variables with trees generated from bootstrap data samples (Breiman, 2001). Low-bias trees are obtained when each tree grows to full maturity without being trimmed. When variables are selected for each tree using a bagging or randomisation procedure, there is little to no correlation between the trees (Chen and Howard, 2015). In order to provide a valuable index of independent variables, random forests can use precise calculations and the Gini index to select features randomly. When variables are randomly assigned weights, the importance index can also reveal their interplay (Vatolkin et al., 2012).

3.2.4 Gradient boosting decision tree

GBDT is a collection of categorisation and regression trees (CART) widely touted as a powerful tool for machine learning (Friedman, 2001). The concept behind GBDT is to combine the outputs from multiple trees into one. As the number of iterations increases, GBDT fits a fresh regression tree in the direction of the gradient of the most recent residual decrease. GB differs from other statistical learning algorithms because it makes

conclusions that can be understood with fewer data preprocessing and parameter tweaking (e.g., ANN and SVM). The method works well with skewed data and can solve classification and regression problems with various distributions (Gaussian, Bernoulli, Poisson and Laplace). Feature selection and the treatment of missing values in predictors contribute to the technique (Guelman, 2012).

3.2.5 Extreme gradient boosting

XGBoost was developed by Chen and Guestrin (2016) to be an enhanced version of Friedman's (2001) GBDT for use in classification and regression. While the GBDT algorithm uses a first-order Taylor expansion of the loss function, XGBoost uses a second-order Taylor expansion, significantly shortening the time required to find the best solution. In addition, the model is improved to avoid the overfitting problem by including standard terms in the objective function and penalising the complexity of each regression tree. In contrast, XGBoost uses a precise greedy algorithm that must iteratively explore the entire training dataset. The technique can find the suitable division condition, but it is time-consuming, requires much memory, and can easily be overfitting (Qian et al., 2022).

3.2.6 Light gradient boosting machine

Microsoft suggested the LightGBM in 2017 to resolve the obstacles of XGBoost (Ke et al., 2017). LightGBM employs a decision tree algorithm based on histograms, a leaf-wise leaf growth strategy with depth constraints, and histogram difference acceleration, among other methods. LightGBM employs a sampling technique called gradient-based one-side sampling (GOSS) to improve training. Its main goal is to ignore data samples with smaller gradients instead of larger ones. GOSS recommended discarding the less-informative data points and using the remaining data to compute the information obtained when deciding optimal splits (Alzamzami et al., 2020). Additionally, LightGBM employs the exclusive feature bundling algorithm to deal with dataset sparsity. It reduces the number of features while retaining the most informative ones by combining mutually exclusive features nearly losslessly.

3.2.7 Extra tree classifier

This ensemble technique uses a base classifier that is either a decision tree or a regression tree that has not been pruned. Compared to other ensembles, this one grows trees using all of the training data and splits nodes at random cut points. In contrast to the RF model, this one employs a random subset for splitting rather than the best split when constructing a DT and does not resample the observation when doing so. The majority vote method combines predictions from multiple trees in classification issues, while the arithmetic means are applied in regression (Geurts et al., 2006).

3.3 Choice of tuning parameters

Python was used to execute every computational method. Using a randomised parameter optimisation method, also known as RandomnisedSearchCV, we adjusted the optimal parameters algorithms to enhance the performance of ensemble classifiers and prevent overfitting issues. By sampling each setting from a distribution of possible parameter

values, RandomisedSearchCV performs a random search over parameters. When training takes a long time and there are many parameters to test, RandomisedSearchCV is handy. Appendix A displays the hyper-parameters used in our analysis.

3.4 Class imbalance handling with SMOTE

SMOTE is a powerful resampling technique utilised to rebalance the sample space for an asymmetrical dataset in order to reduce the impact of the skewed class distribution on the learning process. Chawla et al. (2002) introduced SMOTE as an oversampling technique that generates arbitrary synthetic instances of the minority class in the feature space rather than duplicating the minority sample's existing instances. The SMOTE algorithm utilises KNN to generate these synthetic minority examples. Random neighbours are selected from the KNNs based on the required quantity of oversampling. It is an improved algorithm based on random sampling prone to over-fitting (Ye et al., 2019).

3.5 Feature selection with RFECV and correlation

There are two broad categories of feature selection techniques:

1 supervised or unsupervised

2 wrapper or filter techniques.

RFECV is a wrapper feature selection tool used in the current research. The backward elimination technique used by RFECV begins with a complete set of all features and eliminates the most pointless ones one at a time based on the validation scores (Wang et al., 2019). By removing the features that have no bearing on the accuracy, this process seeks to obtain the ideal number of features to produce the best model accuracy. Before applying RFECV, we analysed correlations and removed variables with high correlations. This step was essential to avoid redundancy in the data caused by specific ratios providing similar information due to their strong correlations.

3.6 Evaluation metrics

We used five evaluation measures to assess the predictive ability of the various ensemble classifiers: accuracy, the area under the ROC curve (AUC), precision, recall, and F-score. A frequently used indicator of classification performance is accuracy, which is defined as follows:

$$Accuracy = \frac{(TP + TN)}{P + N}$$

where P denotes the total number of bankrupt businesses, N is the total number of non-bankrupt businesses, true negative (TN) is the total number of non-bankrupt businesses categorised as non-bankrupt, and true positive (TP) is the total number of bankrupt businesses classified as bankrupt. Recall computes the ratio of predicted to total positive labels, which is defined as:

$$Recall\ or\ sensitivity = \frac{TP}{TP + FN}$$

Precision represents the proportion of accurate positive predictions to the total number of positive predictions.

$$Precision = \frac{TP}{TP + FP}$$

The F-score combines sensitivity and precision and assesses how accurately and robustly the models categorise bankrupt instances. Finally, the area under the ROC curve (AUC) is a common evaluation measure for classification problems.

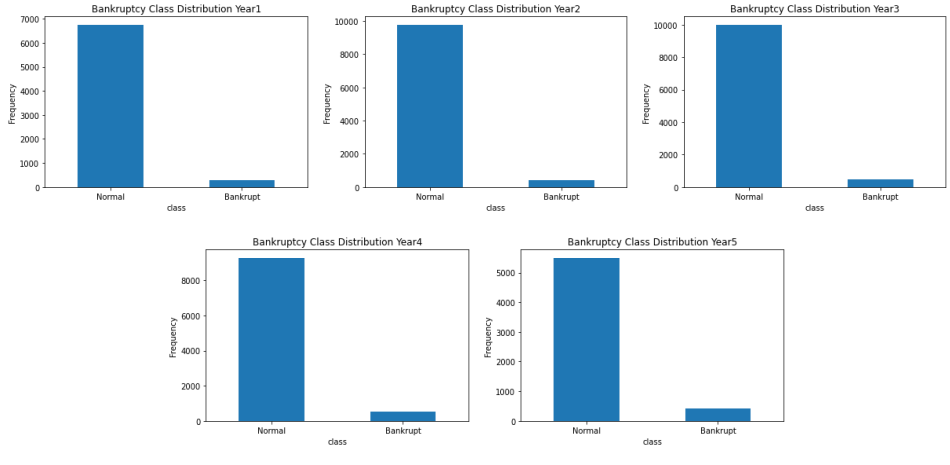
> "For a two-class problem, a ROC curve allows us to visualize the trade-off between the rate at which the model can accurately recognize positive cases versus the rate at which it mistakenly identifies negative cases as positive for different portions of the test set. Any increase in the TP rate occurs at the cost of an increase in the FP (false positive) rate. The area under the ROC curve measures the accuracy of the model." [Han et al., (2012), p.374]

The accuracy with which a model can differentiate between insolvent and solvent examples is quantified by their AUC. The AUC increases as the model become more accurate at separating insolvent from solvent cases. The AUC is preferred for accuracy in skewed datasets (Purda and Skillicorn, 2015) and was also used in this research because it is robust to imbalanced class distributions.

3.7 Result validation

We ran a t-test with a 5% significance threshold to determine if the top performer's AUC and accuracy were significantly higher than the rest of the classifiers.

Figure 1 Size of bankrupt and non-bankrupt (normal) class in the original dataset (see online version for colours)

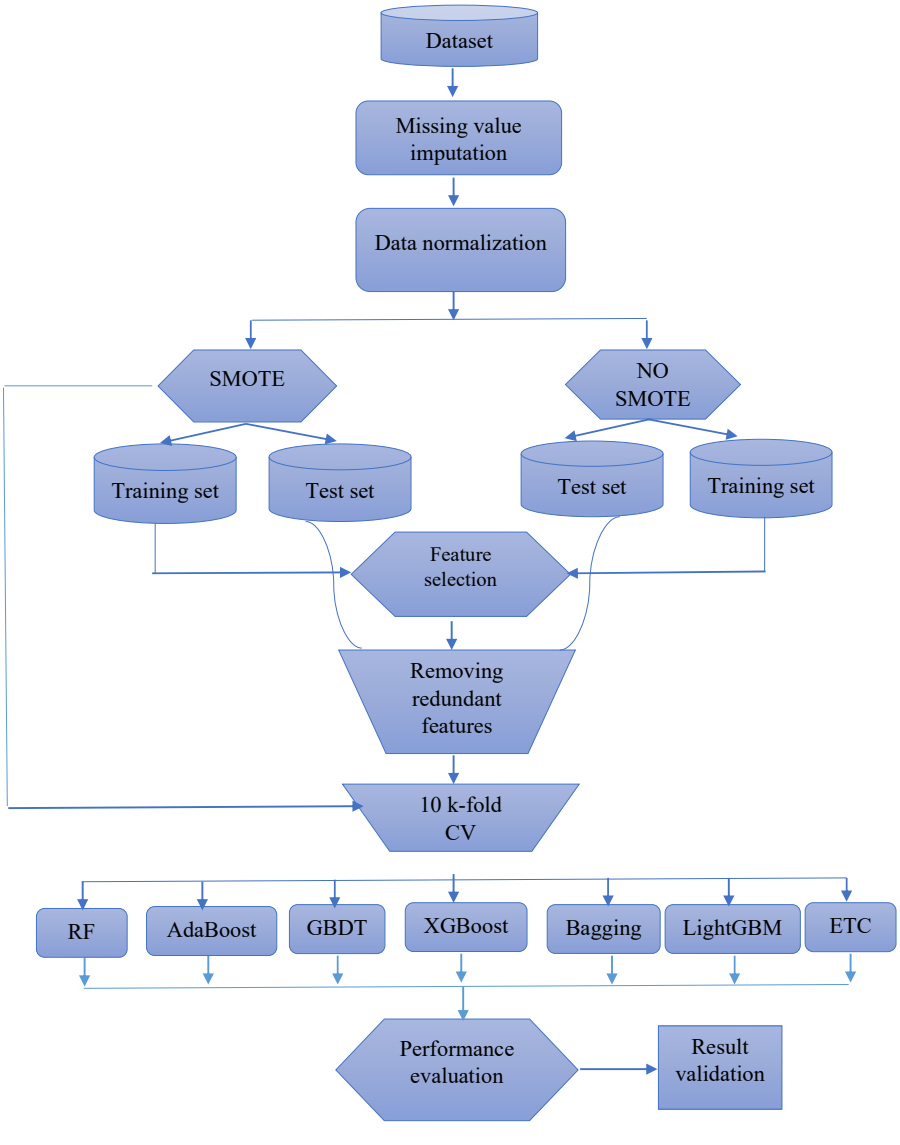


3.8 Experimental setup

As mentioned, we used five real-world datasets from Polish companies to perform our experiments. The datasets were discovered at the University of California, Irvine (UCI) Machine Learning Archive. We first conducted an exploratory data analysis to

understand the dataset thoroughly. We started by using simple imputer to impute missing values from the dataset used in our investigation. We then normalised the dataset using the min-max scaler because the attributes in our dataset are on different scales. The frequency and ratio of bankrupt and non-bankrupt companies in our dataset over the five years under review are shown in Table 1. Our dataset displays a class imbalance for each year (Figure 1), so we used SMOTE to equalise the size of the samples for the positive and negative classes.

Figure 2 Schematic diagram of the experiment conducted in the present study (see online version for colours)



To prevent data leakage, we partitioned the dataset into a set for training and a test set in a proportion of 70:30 before performing a correlation coefficient analysis. Both the training and testing datasets have highly correlated attributes identified and eliminated to control the issue of data leakage. After that, we used RFECV to choose the best set of features for the classifier to achieve the best result. The optimal hyperparameters of each classifier were estimated with a RandomnisedSearchCV. Each classifier is trained and validated with 10-fold cross-validation to produce the most accurate results. The model's accuracy, precision, recall, F1-measure, and AUC performances were then evaluated. The investigation is performed independently on five distinct datasets (years 1, 2, 3, 4, and 5). A t-test with a significance level of 5% was conducted to determine the statistical significance of classifier performance. Figure 2 depicts the schematic diagram of the experiment conducted in the present study.

4 Result analysis and discussion

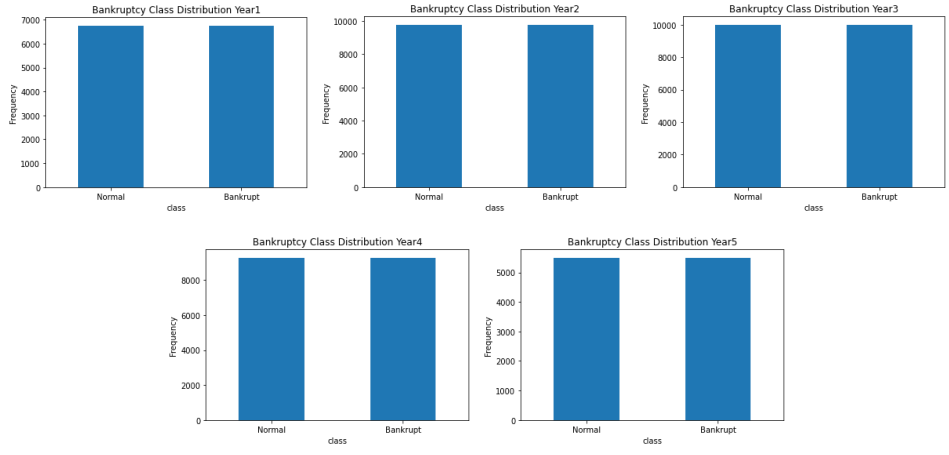
This study aims to determine if using feature selection (correlation and RFECV) and class imbalance handling (SMOTE) to preprocess the data and building prediction models using ensemble-learning techniques (RF, AdaBoost, GBDT, XGBoost, bagging, LightGBM and ETC) is superior to using just one of these methods. In order to answer this research question, we conducted an experiment using five real-world datasets on Polish firms in three distinct ways:

a with feature selection and class imbalance handling

b without class imbalance handling and with feature selection

c with class imbalance handling and without feature selection, as depicted in Figure 2.

Figure 3 Balanced dataset after the implementation of SMOTE (see online version for colours)



In the first set of experiments, we aimed to determine the effect of feature selection (correlation and RFECV) and class imbalance handling (SMOTE) on the performance of ensemble-learning classification models (RF, AdaBoost, GBDT, XGBoost, bagging, LightGBM, ETC). Hence, we performed feature selection and class imbalance handling

strategies during the preprocessing stage and evaluated the performance of classifiers under consideration regarding accuracy, recall, precision, F-score, and area under the curve (AUC). We addressed the class imbalance issue with SMOTE, and the transformed dataset after SMOTE application is shown in Figure 3. A comparison of the distribution of the original and resampled datasets with SMOTE is given in Table 2.

Table 2 The distribution of the dataset before and after SMOTE implementation

Dataset	*Original dataset*					*After SMOTE*				
	Bankrupt		*Non-bankrupt*		*Total*	*Bankrupt*		*Non-bankrupt*		*Total*
	No	*%*	*No*	*%*		*No*	*%*	*No*	*%*	
Year 1	271	3.86	6,756	96.14	7,027	6,756	100	6,756	100	13,512
Year 2	400	3.94	9,773	96.06	10,173	9,773	100	9,773	100	19,546
Year 3	495	4.71	10,008	95.29	10,503	10,008	100	10,008	100	20,016
Year 4	515	5.26	9,277	94.74	9,792	9,277	100	9,277	100	18,554
Year 5	410	6.94	5,500	93.06	5,910	5,500	100	5,500	100	12,000

Table 3 summarises the optimal size of attributes selected in each study period, along with the ID of the selected features after successfully implementing correlation and RFECV[3] on the transformed dataset with SMOTE. We performed the feature selection on the training set to prevent information leakage. Year 5 extracted only fourteen features for model prediction from the initial 64 variables, the least number of optimal features used by the models for differentiating bankrupt firms from non-bankrupt firms. Years 3 and 1 used the highest number of optimal attributes for the prediction task.

Table 3 Results of feature selection tools used in the experiment

Dataset	*Optimal number*	*ID of selected features*[a]
Year 1	27	A1, A3, A4, A5, A8, A9, A10, A13, A15, A16, A19, A21, A27, A28, A29, A33, A37, A41, A45, A47, A49, A53, A55, A57, A59, A60 and A61.
Year 2	20	A4, A5, A15, A20, A21, A26, A27, A29, A30, A37, A39, A40, A41, A45, A55, A57, A58, A59, A60 and A61.
Year 3	28	A2, A3, A5, A13, A15, A19, A20, A21, A22, A24, A27, A29, A32, A34, A35, A36, A37, A39, A41, A45, A55, A56, A57, A58, A59, A60, A61 and A64.
Year 4	18	A5, A6, A15, A21, A24, A27, A28, A29, A34, A37, A39, A41, A42, A51, A55, A58, A61 and A64.
Year 5	14	A6, A15, A21, A27, A29, A32, A34, A37, A39, A41, A42, A55, A58 and A61.

Note: [a]Description of selected attributes are provided in Appendix B.

The results in Table 4 show the performance of various classifiers in experiment 1 with SMOTE and feature selection techniques over five years. The evaluation is based on tenfold cross-validation. The metrics evaluated include AUC, recall, accuracy, F-score, and precision. To achieve optimal performance, we fitted the experimental models with the optimal parameters determined through RandomisedSearchCV techniques, as explained in Appendix A, rather than with default parameters[4]. Across the experimental period, all classifiers generally demonstrated strong performance regarding AUC, recall,

accuracy, precision, and F-measure metrics. However, the AdaBoost classifier exhibited relatively lower performance during the experimental period.

Table 4 Classifiers performances in experiment one with SMOTE and feature selection

Model	*Metric*	*Year 1*	*Year 2*	*Year 3*	*Year 4*	*Year 5*
RF	AUC	0.9989	0.9985	0.9978	0.9969	0.9967
	Recall	0.9928	0.9872	0.9875	0.9914	0.9822
	Accuracy	0.9873	0.9868	0.9812	0.9760	0.9733
	F-score	0.9884	0.9864	0.9815	0.9766	0.9745
	Precision	0.9850	0.9865	0.9757	0.9622	0.9647
AdaBoost	AUC	0.9664	0.9430	0.9517	0.9364	0.9660
	Recall	0.8725	0.8533	0.8684	0.8538	0.8872
	Accuracy	0.8958	0.8650	0.8799	0.8577	0.8932
	F-score	0.8945	0.8634	0.8785	0.8571	0.8925
	Precision	0.9026	0.8739	0.8890	0.8605	0.8981
GBDT	AUC	0.9967	0.9975	0.9989	0.9970	0.9989
	Recall	0.9726	0.9732	0.9754	0.9757	0.9742
	Accuracy	0.9727	0.9790	0.9820	0.9742	0.9832
	F-score	0.9720	0.9724	0.9745	0.9743	0.9735
	Precision	0.9710	0.9720	0.9724	0.9620	0.9723
XGBoost	AUC	*0.9990*	*0.9991*	*0.9990*	*0.9971*	*0.9989*
	Recall	0.9915	0.9884	0.9884	0.9885	*0.9871*
	Accuracy	0.9875	0.9879	0.9866	0.9768	*0.9862*
	F-score	0.9876	0.9879	0.9866	0.9770	0.9862
	Precision	0.9837	0.9874	*0.9849*	0.9658	0.9654
Bagging	AUC	0.9937	0.9917	0.9921	0.9908	0.9895
	Recall	0.9646	0.9681	0.9681	0.9701	0.9650
	Accuracy	0.9652	0.9678	0.9664	0.9610	0.9604
	F-score	0.9660	0.9692	0.9662	0.9600	0.9612
	Precision	0.9674	0.9648	0.9633	0.9533	0.9547
LightGBM	AUC	0.9986	0.9975	0.9980	0.9942	0.9983
	Recall	0.9792	0.9777	0.9799	0.9769	0.9808
	Accuracy	0.9836	0.9775	0.9796	0.9636	0.9826
	F-score	0.9837	0.9775	0.9796	0.9640	0.9825
	Precision	0.9883	0.9773	0.9793	0.9516	0.9843
ETC	AUC	0.9972	0.9968	0.9958	0.9959	0.9955
	Recall	*0.9933*	*0.9918*	*0.9899*	*0.9893*	0.9812
	Accuracy	*0.9903*	*0.9902*	*0.9870*	0.9797	*0.9722*
	F-score	*0.9903*	*0.9900*	*0.9870*	*0.9789*	0.9721
	Precision	*0.9879*	*0.9888*	0.9826	*0.9705*	0.9633

Further, Table 4 highlights the best classifier for each metric considered throughout the study period. Specifically, the XGBoost classifier achieved the highest AUC, indicating its discriminatory solid power. On the other hand, the ETC classifier outperformed other models in terms of accuracy, precision, recall, and F-measure metrics, showcasing its overall effectiveness in classification tasks. To achieve the best performance, we optimised the XGBoost model using four crucial hyperparameters: min_child_weight, max_depth, learning_rate, and gamma. When combined with class imbalance handling and feature selection techniques, these findings offer valuable insights into the strengths and weaknesses of different ensemble classifiers. By identifying the best-performing classifiers for each metric, our research contributes to selecting appropriate models for future tasks involving similar data characteristics.

We conducted a statistical significance analysis of the classification accuracy (Table 5) and AUC (Table 6) using Student's paired t-test, which is a reliable method for comparing different classifiers based on their mean classification performance, as shown in prior research (Hajek and Henriques, 2017). The asterisk signs indicate the statistical significance of the classifiers' performance comparisons. Analysing the p-values, we find that XGBoost achieved the highest AUC scores and significantly outperformed bagging, LightGBM, AdaBoost, and the extra tree classifier since their p-values are less than 0.05. Similarly, RF significantly outperformed bagging, ETC, and AdaBoost. However, the differences in AUC scores between XGBoost and RF, GBDT or LightGBM are not statistically significant (p-values greater than 0.05). GBDT showed competitive performance, but no statistically significant differences were found between GBDT and other classifiers, including XGBoost, RF, LightGBM, and the extra tree classifier. Finally, our results indicate that the AdaBoost classifier achieved lower performance compared to all the other experimental ensembles in the study.

Table 5 Results of student's paired t-test (AUC)

	GBDT	*XGBoost*	*Bagging*	*LightGBM*	*ETC*	*AdaBoost*
RF	0.961	0.090	0.000*	0.568	0.001*	0.002*
GBDT		0.159	0.004*	0.561	0.103	0.002*
XGBoost			0.000*	0.044*	0.005*	0.001*
Bagging				0.003*	0.001*	0.003*
LightGBM					0.239	0.001*
ETC						0.002*

Note: The asterisk signs indicate the statistical significance of the classifiers' performance comparisons.

In contrast to the AUC metric performance, the accuracy metric reveals different results. RF, GBDT, and XGBoost significantly outperformed Bagging and AdaBoost regarding accuracy. At the same time, the differences in accuracy between RF and GBDT, XGBoost, LightGBM, and extra tree classifier (ETC) are not substantial enough to determine a clear winner among them, as their p-values are not statistically significant. Analysing the statistical significance of accuracy performance between GBDT and XGBoost, bagging, LightGBM, and AdaBoost, we find that XGBoost outperforms the GBDT classifier. Additionally, bagging and AdaBoost exhibit poor performance compared to GBDT. Moreover, apart from GBDT, the XGBoost classifier outperforms the bagging and AdaBoost ensembles in terms of accuracy. Furthermore, the Bagging

classifier outperforms only the AdaBoost classifier in accuracy, while AdaBoost's inferior performance against other ensembles is statistically significant.

Table 6 Results of student's paired t-test (accuracy)

	GBDT	*XGBoost*	*Bagging*	*LightGBM*	*ETC*	*AdaBoost*
RF	0.163	0.548	0.001*	0.462	0.058	0.000*
GBDT		0.042*	0.000*	0.029*	0.748	0.000*
XGBoost			0.005*	0.930	0.299	0.000*
Bagging				0.019*	0.001*	0.000*
LightGBM					0.260	0.000*
ETC						0.000*

Note: The asterisk signs indicate the statistical significance of the classifiers' performance comparisons.

Several data mining techniques, including DT, ANN, and SVM, have been effectively used to predict bankruptcy and typically have high accuracy. Moreover, prior studies have also used some of the ensemble classifiers considered in this study. As a result, we contrast the effectiveness of our classifier with that of comparable classifiers used on the Polish bankruptcy datasets for year 1 (Table 7). This comparison highlights the improved performance of the methodology used in experiment one in the present study.

Table 7 Comparison of results with previous studies

Study	*Method*	*Accuracy*	*AUC*
Present study	RF	0.9873	0.9989
	AdaBoost	0.8958	0.9664
	XGBoost	0.9875	0.9990
	GBDT	0.9727	0.9967
	LightGBM	0.9836	0.9986
	Bagging	0.9652	0.9937
	ETC	0.9903	0.9972
Zięba et al. (2016)	RF	NA	0.851
	AB	NA	0.916
	XGB	NA	0.945
	XGBE	NA	0.953
	EXGB	NA	0.959
Soui et al. (2020)	SAE+softmax	0.98	0.961

In the *second set of experiments*, we only performed the feature selection strategy during preprocessing (ignoring the class imbalance issue). We evaluated the performance of classifiers regarding accuracy, recall, precision, F-score, and AUC with 10-fold cross-validation (Table 8). XGBoost exhibited the highest performance on accuracy and precision metrics, whereas LightGBM performed best on recall and AUC metrics. The best classifiers on different metrics throughout the study period are highlighted in bold in Table 8. One crucial observation is that the AUC metric of all the classifiers decreased significantly in this experiment compared to its performance in experiment 1, along with other metrics except overall accuracy. Compared to the decreased performance of

classifiers on AUC metrics, each classifier achieved greater accuracy in experiment 2 (Table 8). He and Garcia (2009) observed that prediction modelling faces many difficulties when dealing with imbalanced classification issues because classification models are susceptible to this problem. It is because most methods for classifying data using machine learning assume that each class has an equal number of examples. Classification accuracy can be increased without tackling the problem of data unbalance, but the outcomes are more likely to belong to the majority class. Hence, the observed improved performance of the accuracy metric in experiment 2 validates the findings of He and Garcia (2009). Moreover, the potential of the classifier to rightly figure out bankrupt firms from non-bankrupt firms (recall or sensitivity) is significantly lower in experiment 2 than in experiment 1.

Table 8 Classifiers performance in experiment two without SMOTE

Model	*Metric*	*Year 1*	*Year 2*	*Year 3*	*Year 4*	*Year 5*
RF	AUC	0.8332	0.8142	0.8743	0.8686	0.9308
	Recall	0.3878	0.2174	0.2564	0.4348	0.5390
	Accuracy	0.9733	0.9665	0.9615	0.9647	0.9605
	F-score	0.5268	0.3427	0.4060	0.5722	0.6393
	Precision	0.8350	0.7524	0.7908	0.8400	0.8435
AdaBoost	AUC	0.8020	0.7835	0.8066	0.8043	0.8905
	Recall	0.1108	0.0275	0.0747	0.1146	0.4414
	Accuracy	0.9608	0.9599	0.9513	0.9507	0.9489
	F-score	0.1750	0.0499	0.1241	0.1947	0.5409
	Precision	0.5208	0.3330	0.4557	0.7067	0.7193
GBDT	AUC	0.8763	0.8332	0.8724	0.8778	0.9266
	Recall	0.3543	0.2100	0.2586	0.4291	0.5390
	Accuracy	0.9722	0.9662	0.9624	0.9647	0.9615
	F-score	0.4897	0.3269	0.3884	0.5596	0.6587
	Precision	0.8171	0.7568	0.8199	0.8202	0.8564
XGBoost	AUC	0.8781	0.8462	0.8736	0.8818	0.9395
	Recall	0.3802	0.1950	0.2342	0.4115	0.5390
	Accuracy	*0.9756*	*0.9674*	*0.9625*	*0.9652*	*0.9620*
	F-score	0.5429	0.3186	0.3685	0.5527	0.6604
	Precision	*0.9700*	*0.9022*	*0.8874*	*0.8513*	*0.8620*
Bagging	AUC	0.7797	0.7501	0.8019	0.8130	0.8900
	Recall	0.3839	0.2649	0.2827	0.4406	0.5487
	Accuracy	0.9723	0.9664	0.9609	0.9637	0.9602
	F-score	0.5090	0.3562	0.3923	0.5946	0.6515
	Precision	0.7812	0.6945	0.6611	0.7890	0.8135
LightGBM	AUC	*0.8823*	*0.8543*	*0.8967*	*0.8917*	*0.9458*
	Recall	*0.4060*	*0.2750*	*0.3190*	*0.4502*	*0.5634*
	Accuracy	0.9738	0.9673	0.9624	0.9652	0.9617
	F-score	0.5429	0.3968	0.4427	0.5745	0.6703
	Precision	0.8303	0.7342	0.7308	0.8041	0.8315

Table 8 Classifiers performance in experiment two without SMOTE (continued)

Model	*Metric*	*Year 1*	*Year 2*	*Year 3*	*Year 4*	*Year 5*
ETC	AUC	0.8124	0.7511	0.8300	0.8391	0.9165
	Recall	0.3211	0.1624	0.2223	0.2484	0.3902
	Accuracy	0.9711	0.9630	0.9597	0.9562	0.9524
	F-score	0.4616	0.2697	0.3450	0.3601	0.5226
	Precision	0.8005	0.6478	0.7158	0.7706	0.8297

The *third set of experiments* was conducted to study the effect of feature selection (correlation and RFECV) on ensemble classifiers' performance. Hence, we only performed the class imbalance strategy during preprocessing (ignoring the feature selection). The tenfold cross-validated experimental results are given in Table 9. From the results, we observed that, other than AdaBoost, all the other ensemble classifiers performed equally well on all the metrics under consideration. Surprisingly, the results of *experiments one and three* exhibit almost similar performances indicating that the feature selection process does not remarkably affect the performances of ensemble classifiers in predicting the bankruptcy of Polish firms.

Table 9 Classifiers performances in experiment three without feature selection

Model	*Metric*	*Year 1*	*Year 2*	*Year 3*	*Year 4*	*Year 5*
RF	AUC	0.9996	0.9992	0.9983	0.9978	0.9964
	Recall	*0.9945*	*0.9955*	*0.9886*	*0.9904*	*0.9880*
	Accuracy	0.9919	0.9907	0.9825	0.9804	0.9742
	F-score	0.9922	0.9918	0.9827	0.9816	0.9750
	Precision	0.9895	0.9876	0.9757	0.9712	0.9644
AdaBoost	AUC	0.9806	0.9609	0.9532	0.9467	0.9650
	Recall	0.9194	0.8859	0.8774	0.8695	0.8921
	Accuracy	0.9262	0.9810	0.8804	0.8688	0.8989
	F-score	0.9255	0.8899	0.8794	0.8677	0.8975
	Precision	0.9324	0.8960	0.8835	0.8689	0.9048
GBDT	AUC	0.9957	0.9111	0.9865	0.9846	0.9884
	Recall	0.9650	0.9407	0.9312	0.9282	0.9445
	Accuracy	0.9661	0.9505	0.9397	0.9305	0.9456
	F-score	0.9657	0.9495	0.9386	0.9294	0.9452
	Precision	0.9673	0.9597	0.9475	0.9330	0.9468
XGBoost	AUC	*0.9998*	*0.9997*	*0.9994*	*0.9993*	0.9989
	Recall	0.9937	0.9931	0.9774	0.9894	0.9900
	Accuracy	*0.9928*	*0.9925*	*0.9883*	*0.9882*	0.9843
	F-score	*0.9927*	*0.9924*	*0.9894*	*0.9880*	*0.9843*
	Precision	*0.9918*	*0.9919*	0.9875	0.9871	0.9790

Table 9 Classifiers performances in experiment three without feature selection (continued)

Model	*Metric*	*Year 1*	*Year 2*	*Year 3*	*Year 4*	*Year 5*
Bagging	AUC	0.9963	0.9969	0.9929	0.9947	0.9916
	Recall	0.9804	0.9813	0.9708	0.9760	0.9663
	Accuracy	0.9763	0.9780	0.9664	0.9701	0.9585
	F-score	0.9760	0.9792	0.9661	0.9689	0.9633
	Precision	0.9721	0.9791	0.9649	0.9635	0.9574
LightGBM	AUC	0.9997	0.9994	0.9989	0.9985	0.9983
	Recall	0.9936	0.9886	0.9820	0.9824	0.9730
	Accuracy	0.9923	0.9886	0.9840	0.9804	0.9804
	F-score	0.9922	0.9885	0.9838	0.9801	0.9804
	Precision	0.9910	0.9888	0.9861	0.9787	0.9883
ETC	AUC	0.9991	0.9982	0.9967	0.9965	0.9956
	Recall	0.9931	0.9919	0.9778	0.9680	0.9604
	Accuracy	0.9912	0.9903	0.9834	0.9783	0.9720
	F-score	0.9917	0.9908	0.9841	0.9778	0.9723
	Precision	0.9905	0.9883	*0.9897*	*0.9879*	*0.9847*

5 Conclusions

One of the most crucial issues in financial research is bankruptcy prediction. As a result, much research has gone into developing methods of foretelling this kind of danger. The goal of this study is to find out if using feature selection (correlation and RFECV) and class imbalance handling (SMOTE) to preprocess the data and building prediction models using ensemble-learning techniques (RF, AdaBoost, GBDT, XGBoost, bagging, LightGBM, ETC) is better than using just one of these methods or combining both. To answer this research question, we experimented with five real-world datasets on Polish firms in three different ways:

1 with feature selection and class imbalance handling

2 without class imbalance handling and with feature selection

3 with class imbalance handling and without feature selection.

Based on the results, using a feature selection strategy during the preprocessing stage has less of an effect on the performance of the ensemble classifiers than fixing the problem of class imbalance in the skewed dataset of Polish firms used in the study. Adopting a feature selection strategy and tuning the hyperparameters of classifiers may not continuously improve the performance of classifiers. Nevertheless, a combined feature selection and class imbalance strategy method during preprocessing made predicting bankruptcy for Polish firms much more accessible than in previous studies. This paper has some flaws that could be fixed with more work. First, we should have attempted to replicate the results of our experiments on another set of skewed datasets about predicting bankruptcy. Second, we only thought about one way to handle class imbalance and one way to choose features. We should have compared how well other resampling and dimension reduction techniques worked.

References

Adisa, J.A., Ojo, S.O., Owolawi, P.A. and Pretorius, A.B. (2019) 'Financial distress prediction: principle component analysis and artificial neural networks', in *2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, IEEE, pp.1–6.

Alfaro, E. et al. (2008) 'Bankruptcy forecasting: an empirical comparison of AdaBoost and neural networks', *Decision Support Systems*, Vol. 45, No. 1, pp.110–122, DOI: 10.1016/j.dss. 2007.12.002.

Altman, E.I. (1968) 'Financial ratios, discriminant analysis and the prediction of corporate bankruptcy', *The Journal of Finance*, Vol. 34, No. 2, pp.78–86.

Altman, E.I. and Bettina, L. (1976) 'A financial early warning system for over-the-counter broker-dealers', *The Journal of Finan.*, Vol. 31, No. 4, p.1976.

Altman, E.I., Iwanicz-Drozdowska, M., Laitinen, E.K. and Suvas, A. (2017) 'Financial distress prediction in an international context: a review and empirical analysis of Altman's Z-score model', *Journal of International Financial Management & Accounting*, Vol. 28, No. 2, pp.131–171.

Alzamzami, F., Hoda, M. and El Saddik, A. (2020) 'Light gradient boosting machine for general sentiment classification on short texts: a comparative evaluation', *IEEE Access*, Vol. 8, pp.101840–101858, DOI: 10.1109/ACCESS.2020.2997330.

Barboza, F., Kimura, H. and Altman, E. (2017) 'Machine learning models and bankruptcy prediction', *Expert Systems with Applications*, Vol. 83, pp.405–417, DOI: 10.1016/ j.eswa.2017.04.006.

Bateni, L. and Asghari, F. (2020) 'Bankruptcy prediction using logit and genetic algorithm models: a comparative analysis', *Computational Economics*, Vol. 55, No. 1, pp.335–348, DOI: 10.1007/s10614-016-9590-3.

Beaver, W.H. (1966) 'Financial ratios as predictors of failure', *Journal of Accounting Research*, Vol. 4, pp.71–111.

Bellovary, J., Giacomino, D. and Akers, M. (2007) 'A review of bankruptcy prediction studies: 1930 to present', *Journal of Financial Education*, Winter, Vol. 33, pp.1–42.

Breiman, L. (1996) 'Nagging predictors', *Machine Learning*, Vol. 24, pp.123–140, DOI: 10.3390/ risks8030083.

Breiman, L. (2001) 'Random forests', *Machine Learning*, Vol. 45, pp.5–32, DOI: 10.1109/ ICCECE51280.2021.9342376.

Chawla, N.V. et al. (2002) 'SMOTE: synthetic minority over-sampling technique', *Journal of Artificial Intelligence Research*, Vol. 30, No. 2, pp.321–357, DOI: 10.1002/eap.2043.

Chen, F.H. and Howard, H. (2015) 'An alternative model for the analysis of detecting electronic industries earnings management using stepwise regression, random forest, and decision tree', *Soft Computing*, Vol. 20, No. 5, pp.1945–1960, DOI: 10.1007/s00500-015-1616-6.

Chen, T. and Guestrin, C. (2016) 'Diagnosis of tuberculosis--newer tests', *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.785–794.

Chen, Z., Chen, W. and Shi, Y. (2020) 'Ensemble learning with label proportions for bankruptcy prediction', *Expert Systems with Applications*, Vol. 146, p.113155, DOI: 10.1016/j.eswa. 2019.113155.

Deakin, E.B. (1972) 'A discriminant analysis of predictors of business failure', *Journal of Accounting Research*, Vol. 10, No. 1, p.167, DOI: 10.2307/2490225.

Fan, S., Liu, G. and Chen, Z. (2017) 'Anomaly detection methods for bankruptcy prediction', *2017 4th International Conference on Systems and Informatics, ICSAI 2017*, pp.1456–1460, DOI: 10.1109/ICSAI.2017.8248515.

Freund, Y. and Schapire, R.E. (1996) 'Experiments with a new boosting algorithm', *ICML*, pp.148–156.

Friedman, J.H. (2001) 'Greedy function approximation: a gradient boosting machine', *Annals of Statistics*, pp.1189–1232.

Geurts, P., Ernst, D. and Wehenkel, L. (2006) 'Extremely randomized trees', *Machine Learning*, Vol. 63, No. 1, pp.3–42, DOI: 10.1007/s10994-006-6226-1.

Gordon, A.M.J. (1971) 'Towards a theory of financial distress', *The Journal of Finance*, Vol. 26, No. 2, pp.347–356.

Guelman, L. (2012) 'Gradient boosting trees for auto insurance loss cost modeling and prediction', *Expert Systems with Applications*, Vol. 39, No. 3, pp.3659–3667, DOI: 10.1016/j.eswa.2011.09.058.

Hajek, P. and Henriques, R. (2017) 'Mining corporate annual reports for intelligent detection of financial statement fraud – a comparative study of machine learning methods', *Knowledge-based Fraud Detection System*, Vol. 128, pp.139–152, DOI: 10.1016/j.knosys.2017.05.001.

Han, J., Kamber, M. and Pei, J. (2012) *Data Mining: Concepts and Techniques*, Vol. 10, pp.978–981, Morgan Kaufman Publishers, Waltham, MA.

He, H. and Garcia, E.A. (2009) 'Learning from imbalanced data', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, No. 9, pp.1263–1284.

Hosaka, T. (2019) 'Bankruptcy prediction using imaged financial ratios and convolutional neural networks', *Expert Systems with Applications*, Vol. 117, pp.287–299, DOI: 10.1016/j.eswa.2018.09.039.

Jabeur, S.B., Gharib, C., Mefteh-Wali, S. and Arfi, W.B. (2021) 'CatBoost model and artificial intelligence techniques for corporate failure prediction', *Technological Forecasting and Social Change*, Vol. 166, p.120658.

Ke, G. et al. (2017) 'LightGBM: a highly efficient gradient boosting decision tree', *Advances in Neural Information Processing Systems*, December, Vol. 2017, pp.3147–3155.

Kim, M.J. and Kang, D.K. (2010) 'Ensemble with neural networks for bankruptcy prediction', *Expert Systems with Applications*, Vol. 37, No. 4, pp.3373–3379, DOI: 10.1016/j.eswa.2009.10.012.

Kim, Y.J., Baik, B. and Cho, S. (2016) 'Detecting financial misstatements with fraud intention using multi-class cost-sensitive learning', *Expert Systems with Applications*, Vol. 62, pp.32–43, DOI: 10.1016/j.eswa.2016.06.016.

Laitinen, E.K. (1991) 'Financial ratios and different failure processes', *Journal of Business Finance & Accounting*, Vol. 18, No. 5, pp.649–673, DOI: 10.1111/j.1468-5957.1991.tb00231.x.

Le, T., Lee, M.Y., Park, J.R. and Baik, S.W. (2018) 'Oversampling techniques for bankruptcy prediction: novel features from a transaction dataset', *Symmetry*, Vol. 10, No. 4, p.79.

Liang, D., Tsai, C.F. and Wu, H.T. (2015) 'The effect of feature selection on financial distress prediction', *Knowledge-Based Systems*, Vol. 73, No. 1, pp.289–297, DOI: 10.1016/j.knosys.2014.10.010.

Lin, W.C., Lu, Y.H. and Tsai, C.F. (2019) 'Feature selection in single and ensemble learning-based bankruptcy prediction models', *Expert Systems*, Vol. 36, No. 1, pp.1–8, DOI: 10.1111/exsy.12335.

Lukason, O. and Laitinen, E.K. (2019) 'Firm failure processes and components of failure risk: an analysis of European bankrupt firm', *Journal of Business Research*, Vol. 98, pp.380–390.

Matin, R. et al. (2019) 'Predicting distresses using deep learning of text segments in annual reports', *Expert Systems with Applications*, Vol. 132, pp.199–208, DOI: 10.1016/j.eswa.2019.04.071.

Ocal, N., Ercan, M.K. and Kadioglu, E. (2015) 'Predicting financial failure using decision tree algorithms: an empirical test on the manufacturing industry at Borsa Istanbul', *International Journal of Economics and Finance*, Vol. 7, No. 7, pp.189–206, DOI: 10.5539/ijef.v7n7p189.

Powell, W.B. (2009) 'What you should know about approximate dynamic programming', *Naval Research Logistics*, Vol. 31, No. 11, pp.535–536, DOI: 10.1097/00152193-198712000-00022.

Purda, L. and Skillicorn, D. (2015) 'Accounting variables, deception, and a bag of words: assessing the tools of fraud detection', *Contemporary Accounting Research*, pp.1–32, DOI: 10.1111/1911-3846.12089.

Qian, H. et al. (2022) 'Financial distress prediction using a corrected feature selection measure and gradient boosted decision tree', *Expert Systems with Applications*, February, Vol. 190, p.116202, DOI: 10.1016/j.eswa.2021.116202.

Santoso, N. and Wibowo, W. (2018) 'Financial distress prediction using linear discriminant analysis and support vector machine', *Journal of Physics: Conference Series*, DOI: 10.1088/1742-6596/979/1/012089.

Smiti, S. and Soui, M. (2020) 'Bankruptcy prediction using deep learning approach based on borderline SMOTE', *Information Systems Frontiers*, Vol. 22, No. 5, pp.1067–1083, DOI: 10.1007/s10796-020-10031-6.

Soui, M., Smiti, S., Mkaouer, M.W. and Ejbali, R. (2020) 'Bankruptcy prediction using stacked auto-encoders', *Applied Artificial Intelligence*, Vol. 34, No. 1, pp.80–100.

Sun, J. et al. (2020) 'Class-imbalanced dynamic financial distress prediction based on AdaBoost-SVM ensemble combined with SMOTE and time weighting', *Information Fusion*, December, Vol. 54, pp.128–144, DOI: 10.1016/j.inffus.2019.07.006.

Tsai, C.F. (2009) 'Feature selection in bankruptcy prediction', *Knowledge-Based Systems*, Vol. 22, No. 2, pp.120–127, DOI: 10.1016/j.knosys.2008.08.002.

Tsai, C.F. et al. (2021) 'Combining feature selection, instance selection, and ensemble classification techniques for improved financial distress prediction', *Journal of Business Research*, Vol. 130, No. 300, pp.200–209, DOI: 10.1016/j.jbusres.2021.03.018.

Vatolkin, I. et al. (2012) 'Multi-objective evolutionary feature selection for instrument recognition in polyphonic audio mixtures', *Soft Computing*, Vol. 16, No. 12, pp.2027–2047, DOI: 10.1007/s00500-012-0874-9.

Wang, C., Xiao, Z. and Wu, J. (2019) 'Functional connectivity-based classification of autism and control using SVM-RFECV on rs-fMRI data', *Physica Medica*, April, Vol. 65, pp.99–105, DOI: 10.1016/j.ejmp.2019.08.010.

Wang, G., Chen, G. and Chu, Y. (2018) 'A new random subspace method incorporating sentiment and textual information for financial distress prediction', *Electronic Commerce Research and Applications*, Vol. 29, pp.30–49, DOI: 10.1016/j.elerap.2018.03.004.

Wilcox, J. (1973) 'A prediction of business failure using accounting data', *Journal of Accounting Research*, Vol. 11, pp.163–179, https://doi.org/10.2307/2490035.

Ye, H., Xiang, L. and Gan, Y. (2019) 'Detecting financial statement fraud using random forest with SMOTE', *IOP Conference Series: Materials Science and Engineering*, DOI: 10.1088/1757-899X/612/5/052051.

Yu, Q. et al. (2014) 'Bankruptcy prediction using extreme learning machine and financial expertise', *Neurocomputing*, Vol. 128, pp.296–302, DOI: 10.1016/j.neucom.2013.01.063.

Zelenkov, Y. and Volodarskiy, N. (2021) 'Bankruptcy prediction on the base of the unbalanced data using multi-objective selection of classifiers', *Expert Systems with Applications*, Vol. 185, p.115559, DOI: 10.1016/j.eswa.2021.115559.

Zelenkov, Y., Fedorova, E. and Chekrizov, D. (2017) 'Two-step classification method based on genetic algorithm for bankruptcy forecasting', *Expert Systems with Applications*, Vol. 88, pp.393–401, DOI: 10.1016/j.eswa.2017.07.025.

Zięba, M., Tomczak, S.K. and Tomczak, J.M. (2016) 'Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction', *Expert Systems with Applications*, Vol. 58, pp.93–101, DOI: 10.1016/j.eswa.2016.04.001.

Zou, Y., Gao, C. and Gao, H. (2022) 'Business failure prediction based on a cost-sensitive extreme gradient boosting machine', *IEEE Access*, Vol. 10, pp.42623–42639, DOI: 10.1109/ACCESS.2022.3168857.

Notes

1 https://archive.ics.uci.edu/ml/datasets/Polish+companies+bankruptcy+data.

2 The dataset used in the present investigation is described in great depth by Zięba et al. (2016).

3 We used a decision tree estimator to perform RFECV, and other essential parameters used are step = 1, scoring = 'neg_mean_squared_error', cv = 4, verbose = 1, and n_jobs = –1).

4 AdaBoost, bagging, ETC classifiers are fitted with default parameters throughout the study period, whereas GBDT and XGBoost are fitted with optimal parameters. Except for year 5, RF and LightGBM are fitted with default parameters, as default parameters provide better performance against the best parameters identified through randomised search CV.

Appendix A

Results of hyperparameter tuning

Classifier	*GBDT*	*XGBoost*	*RF*	*LightGBM*
Year 1	(learning_ rate = 0.02, n_estimators = 500, max_depth = 6, subsample = 0.1)	(min_child_ weight = 3, learning_ rate = 0.05, max_depth = 12, gamma = 0.1)	Default	Default
Year 2	(learning_ rate = 0.01, n_estimators = 500, max_depth = 9)	(min_child_ weight = 1, learning_rate = 0.3, max_depth = 8, gamma = 0.1)	Default	Default
Year 3	(learning_ rate = 0.01, max_depth = 7, n_estimators = 500)	(min_child_ weight = 3, learning_ rate = 0.25, max_depth = 12, gamma = 0.0)	Default	Default
Year 4	(learning_ rate = 0.03, n_estimators = 400, max_depth = 5, subsample = 0.1)	(min_child_ weight = 3, learning_rate = 0.3, max_depth = 8, gamma = 0.3)	Default	Default
Year 5	(learning_ rate = 0.02, n_estimators = 500, max_depth = 6, subsample = 0.1)	(min_child_ weight = 3, learning_ rate = 0.25, max_depth = 12, gamma = 0.4)	(max_features = 0.2, min_samples_ leaf = 2, max_samples = 0.5, Min_samples_ split = 5, N_estimators = 120)	(feature_ fraction = 1, max_depth = 10, n_estimators = 1,000, Num_leaves = 6)

Appendix B

Variables in the original dataset taken from

ID	*Description*	*ID*	*Description*
A1	Net profit/total assets	A32	(Current liabilities ∗ 365)/cost of products sold
A2	Total liabilities/total assets	A34	Operating expenses/total liabilities
A3	Working capital/total assets	A35	Profit on sales/total assets
A4	Current assets/short-term liabilities	A36	Total sales/total assets
A5	[(Cash + short-term securities + receivables – short-term liabilities)/(operating expenses – depreciation)] * 365	A37	(Current assets – inventories)/long-term liabilities
A6	Retained earnings/total assets	A39	Profit on sales/sales
A8	Book value of equity/total liabilities	A40	(Current assets – inventory – receivables)/short-term liabilities
A9	Sales/total assets	A41	Total liabilities/((profit on operating activities + depreciation) ∗ (12/365))
A10	Equity/total assets	A42	Profit on operating activities/sales
A13	(Gross profit + depreciation)/sales	A45	Net profit/inventory
A15	(Total liabilities ∗ 365)/(gross profit + depreciation)	A47	(Inventory ∗ 365)/cost of products sold
A16	(Gross profit + depreciation)/total liabilities	A49	EBITDA (profit on operating activities – depreciation)/sales
A19	Gross profit/sales	A51	Short-term liabilities/total assets
A20	(Inventory ∗ 365)/sales	A53	Equity/fixed assets
A21	Sales(n)/sales(n – 1)	A55	Working capital
A22	Profit on operating activities / total assets	A56	(Sales – cost of products sold)/sales
A24	Gross profit (in 3 years)/total assets	A57	(Current assets – inventory – short-term liabilities)/(sales – gross profit – depreciation)
A26	(Net profit + depreciation)/total liabilities	A58	Total costs/total sales
A27	Profit on operating activities/financial expenses	A59	Long-term liabilities/equity
A28	Working capital/fixed assets	A60	Sales/inventory
A29	Logarithm of total assets	A61	Sales/receivables
A30	(Total liabilities – cash)/sales	A64	Sales/fixed assets