



An experimental survey of imbalanced learning algorithms for bankruptcy prediction

Peter Gnip¹ · Róbert Kanász¹ · Martin Zoričák² · Peter Drotár¹

Accepted: 7 January 2025 / Published online: 25 January 2025
© The Author(s) 2025

Abstract

Information about imminent bankruptcy is crucial for financial institutions, decision-making managers, and state agencies. Since bankruptcy prediction is a prevalent research topic, many new methods have been continuously proposed. Bankruptcy prediction is frequently approached as a binary classification task. Since bankruptcy datasets are inherently imbalanced, bankruptcy classification is usually performed using class imbalance learning methods. The nature of these methods is very diverse, but they can usually be categorized as ensemble, cost-sensitive, sampling, and hybrid methods. In this paper, we provide a comprehensive experimental comparison of 45 methods. These methods were selected because they cover the approaches and algorithms frequently employed for bankruptcy prediction and imbalanced learning. Extensive experiments on 15 publicly available datasets with different imbalance ratios showed that the methods based on a combination of ensemble learning and undersampling are able to handle data imbalance and achieve the best results for bankruptcy classification.

Keywords Bankruptcy prediction · Financial distress prediction · Machine learning · Ensemble learning · Data sampling · Imbalanced learning

✉ Peter Gnip
peter.gnip@tuke.sk

✉ Martin Zoričák
martin.zoricak@tuke.sk

Róbert Kanász
robert.kanasz@student.tuke.sk

Peter Drotár
peter.drotar@tuke.sk

¹ Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 040 01 Košice, Slovak Republic

² Department of Finance, Faculty of Economics, Technical University of Košice, Nemcovej 32, 040 10 Košice, Slovak Republic

1 Introduction

Corporate bankruptcy can have a very negative impact on society, business, and the economy. If aggregated, company bankruptcy events also profoundly influence the global economy, as in the case of the financial crisis in 2008–2010. Therefore, knowing the risk of bankruptcy is very important not only for investors and creditors who can lose their funds but also for regulators and state agencies that monitor the financial health of individual companies and prevent systematic risks. The effect of business failure can be devastating to firm owners, partners, and creditors. This strongly motivates academics and practitioners to develop models and tools for bankruptcy prediction.

Statistically, bankruptcy prediction is a binary classification task, where a company is identified as either nonbankrupt or bankrupt based on available information captured from financial features. The most common financial features are financial ratios derived from information in balance sheets and income statements. The size and source of capital may affect the quality of this information. Larger companies or those traded on the stock market generally have an obligation to verify financial statements by external auditors and have to follow international accounting standards (such as the International Financial Reporting Standards (IFRS) or Generally Accepted Accounting Principles (GAAP)). On the other hand, small or micro companies may have more lenient reporting duties, which is usually due to simpler capital structure and/or business transactions.

Historically, the most common statistical tool is multiple discriminant analysis, which was first used by Altman (1968) to develop a prediction model known as the Z model. This model is based on Beaver's (1966) methodology. Altman's model is widely accepted and is currently used in some applications. However, with the rise of machine learning, more sophisticated algorithms and methodologies for bankruptcy prediction have been introduced. Models based on decomposition and fusion methods Sun et al. (2021), threshold-based models Véganzones (2022), and many others Kuizinienė et al. (2022) represent only a fraction of the vast range of machine learning approaches for bankruptcy prediction. Unfortunately, most machine learning approaches consider bankruptcy prediction a conventional balanced learning problem. Usually, only a subset of nonbankrupt companies is considered to build a balanced dataset with an equal number of bankrupt and nonbankrupt companies. However, this is different from the real world. In practice, even in a financial crisis, only a fraction of all companies end up bankrupt. Considering only a subset of nonbankrupt companies, a significant part of the data is not used to balance a dataset. Thus, the models are not properly validated. Hence, bankruptcy prediction should be approached as an imbalanced learning problem.

Imbalanced data are characterized by one class, denoted as the majority class, having a much greater number of samples than the others. The classes with limited numbers of samples are denoted as minority classes. The identification of minority samples is usually a challenge for standard classifiers such as logistic regression and support vector machines. They mostly provide suboptimal solutions. Even though the coverage of the majority class is good, the minority class may be treated as noise by a learning model. One of the issues is the usage of performance metrics such as accuracy score to guide the learning process. This leads to a strong bias toward the majority class, while the minority class remains unknown to the model.

Many machine learning approaches have been proposed in recent years to solve imbalanced learning tasks. These approaches can be classified into several categories: sampling methods, ensemble methods, outlier detection methods, and cost-sensitive learning methods. However, the most recent approaches are mostly based on hybrid models that combine two or more categories to build an efficient imbalanced learning model. Authors Dasilas and Rigani (2024) reviewed 207 studies employing mainly hybrid methods for bankruptcy prediction, from which 28 studies addressed the issue of data imbalance in this area of research. Similarly Dovile and Tomas (2022) systematically reviews 232 research articles on the application of artificial intelligence methods for predicting financial distress, with a focus on issues like data imbalance, dimensionality reduction, and performance evaluation metrics. It highlights the importance of proper data preprocessing and finds the frequent use of logistic regression. Another more general, systematic literature review Nazareth and Ramana Reddy (2023) comprehensively analyzes the recent advancements in applying machine learning and deep learning across six financial domains: stock markets, portfolio management, cryptocurrency, forex markets, financial crises, and bankruptcy/insolvency. By examining 126 articles, the review highlights the effectiveness of methods like support vector machine (SVM), decision trees (DTs), and ensemble methods. While these studies serve as excellent references, none of them experimentally compare the performance of different bankruptcy prediction methods using the same or comparable datasets. The only study that provides some experimental analysis is Amirshahi and Lahmiri (2024). The authors offer an overview of recent bankruptcy prediction papers that employ ensemble methods on both balanced and imbalanced data. However, their experiments are limited to ensemble methods and are conducted on a single dataset.

This paper extensively evaluates multiple imbalanced learning methods for the bankruptcy prediction task. Conventional reviews provide information about the models' taxonomy, applications, and theoretical foundations but do not offer a fair comparison of different approaches based on experimental results. We aim to fill the gap between a theoretical review and a practitioner's need to know which method is best for bankruptcy prediction problems. We selected 45 methods representing different sampling methodologies, cost-sensitive learning, ensemble learning, outlier detection, and hybrid methods and evaluated their performance on 15 publicly available bankruptcy classification datasets. Our extensive experiments show that the best performance is achieved using hybrid undersampling ensembles.

The remainder of this paper is organized as follows. The next section briefly reviews bankruptcy prediction and imbalanced learning methods. This is followed by a detailed description of the datasets used in the experiments. Section 4 describes the methods utilized in this study. The experimental settings are outlined in Sect. 5. Finally, the experimental results, followed by a statistical comparison of methods and discussion, are given in Sects. 6 and 7, respectively. Conclusions are drawn in the last section of the paper.

2 Literature review

Bankruptcy prediction has been studied for a very long time. Its origins date back to the beginning of the 20th century. The authors applied conventional statistical methods based on financial ratios from the first models. With increasing computing power, many intelligent

machine learning methods have been proposed. In this section, we provide a brief review of bankruptcy prediction approaches. Additionally, the issue of imbalanced data, one of the most prevalent challenges in bankruptcy prediction, is discussed.

2.1 Bankruptcy prediction

One of the first studies dealing with bankruptcy prediction was Fitzpatrick's (1932) study, where the author compared 13 ratios of failed and successful firms (19 of each firm type). He found that successful companies displayed favorable ratios in most cases, while failed firms had unfavorable ratios when compared with "standard" ratios and ratio trends.

The turning point in bankruptcy prediction research occurred in the second half of the 20th century, when Beaver (1966) published his pioneering study. In this study, Beaver summarized the results of an empirical analysis by applying scientific methods to financial data to predict bankruptcy. The study showed some promising results, but it had drawbacks arising from its univariate approach.

Figini et al. (2017) proposed novel approaches to predict default for small and medium-sized enterprises (SMEs). They used multivariate outlier detection techniques based on the local outlier factor to improve the out-of-sample performance of parametric and nonparametric models for credit risk estimation. The models were tested on a real dataset provided by UniCredit Bank. They compared single and ensemble models and proved that the proposed technique is a suitable competitor to logistic regression.

Another conventional technique used for bankruptcy prediction is k-nearest neighbors (KNN). Chen et al. (2011) proposed a novel modification of KNN based on an adaptive fuzzy KNN method, where the continuous particle swarm optimization approach adaptively specifies the parameters of the method.

Ouenniche et al. (2018) proposed an out-of-sample framework for bankruptcy prediction utilizing the technique for order performance by the similarity to ideal solution (TOPSIS) classifier combined with a KNN model. The authors used a dataset of United Kingdom firms listed on the London Stock Exchange from 2010 to 2014.

In the late 2000 s, conventional methods like SVM gained popularity in bankruptcy prediction (Lin et al. 2011). Compared to the prediction accuracy of SVM with other conventional machine learning methods, the results of the following research papers (Wang and Ma 2012) and (Ding et al. 2008) suggest that SVM's performance is superior.

Currently, ensemble learning is becoming popular in the area of bankruptcy prediction. Chen et al. (2020) proposed two novel prediction methods, bagged-prediction SVM (pSVM) and boosted-pSVM, based on proportion support vector machines and ensemble strategies, including bagging and boosting.

Kim and Upneja (2021) proposed a variation of the ensemble method for bankruptcy prediction. They offered three models: an entire period, an economic downturn, and an economic expansion model. All three methods use a majority voting ensemble method with a DT as a base estimator.

Zelenkov and Volodarskiy (2021) also proposed an ensemble model for bankruptcy prediction. To create an ensemble, they used a multiobjective classifier selection algorithm that only selects classifiers that belong to the Pareto-optimal set in the false-positive rate (FPR)/false-negative rate (FNR) space. The main goal of this algorithm is to minimize the FPR and FNR parameters simultaneously.

Authors Ainan et al. (2024) proposed a novel hybrid approach that combines the strengths of ensemble learning and artificial neural networks called XGBoost+ANN. The XGBoost+ANN approach integrates a comprehensive feature set and optimizes parameters using genetic algorithms instead of feature selection methods.

Most of the research regarding bankruptcy prediction is focused on accounting data from financial statements or derived financial ratios as the primary source of information. The more comprehensive approach used by Jiang et al. (2023) utilized semantic features in the patent text on Chinese manufacturing companies operating between the years 2019 and 2020. They also conducted a feature importance analysis based on the SHAP value, indicating that financial and patent features are essential.

Da Silva Mattos and Shasha addressed the problem of low-quality information in financial reports (Silva Mattos and Shasha 2024) by adding to 17 common financial ratios and 9 dummy variables, such as whether the company has been audited, whether the company has reported all legally needed information, etc. An analysis was conducted on Brazilian companies from 2007 to 2020. They utilized the following intelligent methods: XGBoost, AdaBoost, random forest (RF), bagging, SVM, and logistic regression. The authors concluded that including additional dummy variables led to better results, and RF reached the highest evaluated by area under the receiver operating characteristic curve (AUC) score.

Authors Shakeel et al. (2023) analyzed 36 bankruptcy prediction papers from 2015 to 2021 from a feature importance point of view. They identified that the three main feature selection approaches are filter, wrapper, and embedded, while the filter approach is the most prevalent. Additionally, review analysis shows that researchers commonly use multiple feature selection methods to identify the most relevant economic attributes. According to the review study (El Madou et al. 2024), applying task-specific features in the bankruptcy prediction process can enhance the accuracy and effectiveness of the predictive model.

The field of bankruptcy prediction is a dynamic and continually evolving area of research, resulting in a growing body of literature in the form of comprehensive review papers. These review papers offer a focused examination of the subject matter from various perspectives. In their work Alaka et al. (2018), Alaka et al. analyzed 49 journal papers, evaluating a range of machine learning methods using 13 criteria. The authors concluded that the significance of bankruptcy prediction extends beyond mere accuracy. It also considers the employed variable and model interpretability. The authors assert that choosing an appropriate bankruptcy prediction method should be contingent upon the specific application requirements, and they furnish a robust framework for method selection.

The employment of machine learning methods focused on deep learning was analyzed by Zheng et al. (2023) and Qu et al. (2019). Moreover, Kuiziniene et al. (2022) presented a comprehensive survey scrutinizing 232 articles published between 2017 and February 2022. They concluded that commonly used datasets include the publicly available datasets from Poland, Spain, or Japan. Logistic regression is the most prevalent method and is mainly used as a benchmark; however, artificial neural networks, SVM, DT, RF, and boosting-based methods are also frequently adopted.

From the methodology perspective, Cheraghali et al. (2023) analyzed 145 SME bankruptcy prediction studies. The time span of studies is over 40 years. They concluded that financial ratios are the most common source of information; however, manager-owner characteristics, macroeconomic information, and credit information are also used. To tackle the imbalanced dataset problem, authors in analyzed studies use undersampling (random/

stratified) and oversampling. The most prevalent method for oversampling is the synthetic minority oversampling technique (SMOTE). It is evident that the number of attributes used in models is increasing, and so is the increasing usage of intelligent methods. The top three performing methods, based on accuracy and AUC score, are the hybrid method combining gradient boosting decision tree, convolutional neural network, and LR, as well as Random Forest (RF) and Support Vector Machine (SVM).

Quite a different approach for bankruptcy prediction was proposed by Chen et al. (2023). The authors integrated text-based information from annual reports of United States enterprises that notably increased the prediction performance of four machine learning methods: RF, XGBoost, logistic regression, and SVM. Moreover, they observed that this approach notably decreased Type II errors regarding short-term bankruptcy forecasting.

A cutting-edge approach emerging in bankruptcy prediction involves leveraging generative large language models. Here, authors Loukas et al. (2023) realized a bunch of experiments demonstrating that querying GPT-3.5 and GPT-4 can surpass fine-tuned, non-generative models even with limited examples. Furthermore, they observed that generative models perform better when presented with representative samples chosen by human experts compared to randomly selected ones for the given task.

2.2 Imbalanced learning

Most problems associated with predicting bankruptcy are closely related to the strongly imbalanced datasets used in training the models. Currently, many methods enable the classification of imbalanced data. These methods can generally be divided into data-level, algorithm-level, and hybrid methods.

Data-level methods primarily focus on changing the data distribution to balance individual classes. These methods are based on oversampling of the minority class samples or undersampling of majority class samples. A popular method for oversampling the minority class is the SMOTE (Chawla et al. 2002) and its derivatives, such as reduced noise SMOTE (RN-SMOTE) (Arafa et al. 2022). RN-SMOTE first oversamples the training data using SMOTE, introducing noisy, non-sampled synthetic instances in the minority class. Next, density-based spatial clustering of applications with noise (DBSCAN) is applied to detect and remove noise. Then, the clean artificial instances are combined with the original data. Finally, RN-SMOTE applies SMOTE again to balance the dataset before introducing it to the underlying classifier. Another derivative of SMOTE is its parallel implementation dedicated to big data named Approx-SMOTE (Juez-Gil et al. 2021). It uses an approximated version of the k-nearest neighbor, which makes it highly scalable.

On the other hand, algorithm-level methods are mainly based on new algorithms or modifying existing techniques to improve the model accuracy. This category includes algorithms based on ensemble learning. Niu et al. (2023) proposed a novel anomaly detection approach based on ensemble semisupervised active learning. They proposed a balanced sampling strategy that combines margin sampling and democratic colearning techniques to construct a balanced training set that consists of manually labeled high-information samples and automatically labeled high-confidence samples to train the detection model effectively on a limited budget. Another example of ensemble learning for imbalanced data is the method proposed by Muslim et al. (2023). The proposed model consists of three optimization parts: SMOTE, feature selection, and stacking ensemble learning. The SMOTE method was used

to balance the data, and a feature selection light gradient boosting machine (LightGBM) and stacking ensemble learning with extreme gradient boosting (LGBFS-StackingXGBoost) were used to optimize the prediction performance.

The last group of algorithms combines multiple approaches to address imbalanced data. Shi et al. (2023) proposed a hybrid imbalanced classification model based on data density. They combined a data-level method with an algorithm-level method. At the data level, the density-based resampling method is presented. The data partition algorithm divides the data space into five regions based on the data density. The corresponding subsets are generated by sampling from the divided regions to improve the recognition of different class instances. At the algorithm level, corresponding ensemble models for different class instances are constructed. In the final stage, the model selection algorithm is presented. On this basis, an appropriate model is selected for each instance based on its distribution. Another example of the combination of algorithms is RGAN-EL (Ding et al. 2023), which combines an improved generative adversarial network and ensemble learning to improve the classification performance on imbalanced data. Authors Ding et al. (2023) also utilize generative adversarial networks combined with transfer learning (RVGAN-TL) in tabular data, showing superior classification performance on 20 real datasets from different domains. Review papers such as Kaur et al. (2019), Lin et al. (2023), Rezvani and Wang (2023), and Khan et al. (2023) provide a more comprehensive overview of recent approaches and applications for imbalanced learning.

The literature review reveals three main categories of scientific papers related to bankruptcy prediction. The first category includes studies that propose novel methods, typically compared with classical statistical techniques such as LR or some form of discriminant analysis. The second category consists of review papers that compare various techniques, often utilizing different bankruptcy datasets. The third category comprises papers that focus primarily on addressing the issue of class imbalance in bankrupt/non-bankrupt predictions. Table 1 provides a concise overview of representative papers from each category.

In this study, we present a combination of various datasets with different levels of imbalance and a broad spectrum of methods. The datasets differ in a number of attributes, geographically, and in the time span they cover. This allowed us to test the performance of each method in different conditions, which led to a more robust comparison of methods.

3 Data

In our paper, we employed several publicly available datasets from four countries, namely, the Slovak Republic (Drotár et al. 2019), Bosnia and Herzegovina (Memic and Memic 2020), Taiwan (Liang and Tsai 2020) and Poland (Zięba et al. 2016). The term bankruptcy is generally associated with a legal process defined by law for a specific country. In this process, the debt of the bankrupt entity may be partially or fully waived. Bankruptcy should not be confused with insolvency, which may be only a temporal inability to pay off debt. The datasets span from 1999 to 2016; the period for individual datasets is shown in Fig. 1.

The heterogeneity of the datasets is threefold. First, each dataset covers different countries with different economies and geographical dispositions. For example, Slovakia, Poland, and Bosnia and Herzegovina share some common history and are relatively in close proximity. On the other hand, Taiwan has its specifics within its region. Second, each dataset covers a

Table 1 Overview of scientific publications reviewed in this study

	Paper reference	Dataset	Data source	Category
Bankruptcy data	Ding et al. (2008)	China (2001-2004)	Stock exchange	Conventional (SVM, ANN)
	Chen et al. (2011)	Poland (1997-2001)	Financial statements	Cost-sensitive
		Australia	Credit card applications	
	Wang and Ma (2012)	China (2006-2007)	Financial statements	Ensemble
	Figini et al. (2017)	Italy	Financial statements	Hybrid (ensemble + outlier)
	Ouenniche et al. (2018)	United Kingdom (2010-2014)	Stock exchange	Conventional (kNN + TOPSIS)
	Chen et al. (2020)	Poland (2007-2013)	Financial statements	Ensemble
		Australia		
		Japan		
		Germany		
	Kim and Upneja (2021)	USA (1980-2017)	Stock exchange	Ensemble
	Zelenkov and Volodarskiy (2021)	Poland (2007-2013)	Financial statements	Ensemble
		Russia	Financial statements + central bank	
	Jiang et al. (2023)	China (2019-2020)	Financial statements + patents	Ensemble
	Chen et al. (2023)	USA (1994-2018)	Financial reports	Ensemble
	Muslim et al. (2023)	USA (2007-2015)	Loans information	Hybrid (ensemble + oversampling)
Other domains	Amirshahi and Lahmiri (2024)	Poland (2007-2013)	Financial statements	Ensemble
	Ainan et al. (2024)	Poland (2007-2013)	Financial statements	Ensemble
	Silva Mattos and Shasha (2024)	Brazil (2007-2020)	Financial statements	Ensemble
	Chawla et al. (2002)	9 imbalanced dataset	Various online sources	Undersampling oversampling
	Juez-Gil et al. (2021)	6 imbalanced datasets	UCI Machine Learning Repository	Cversampling
	Arafa et al. (2022)	9 imbalanced datasets	UCI Machine Learning repository	Cversampling
	Niu et al. (2023)	NSL-KDD: network intrusion traffic dataset	–	Sampling
	Shi et al. (2023)	18 imbalanced datasets	KEEL repository	Hybrid (ensemble + sampling)
	Lin et al. (2023)	44 imbalanced datasets	KEEL repository	Combined sampling
	Ding et al. (2023)	20 imbalanced datasets	KEEL repository	Oversampling
	Ding et al. (2023)	21 imbalanced datasets	KEEL repository	Hybrid (sampling + ensemble)

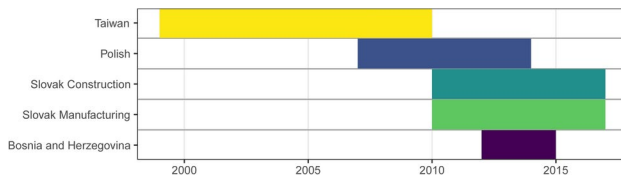


Fig. 1 Time span of the data captured in the datasets used in this paper

different time period. The Taiwan and Polish dataset covers the time of the financial crisis in 2008. Slovak and Bosnian and Herzegovina datasets cover the recovery period after the crisis. Third, each dataset consists of different attributes, which may offer different insights into the company's inner workings.

The most common source of information for company performance and bankruptcy prediction is financial ratios. Financial ratios are derived from financial statement data such as balance sheets, profit and loss statements, income, and cash flow statements. The advantage of financial ratios is comparability across companies of various sizes. All attributes by dataset are listed in the Appendix A. There are five common categories of financial ratios: (a) liquidity ratios, (b) leverage ratios, (c) efficiency ratios, (d) profitability ratios, and (e) market value ratios. Market value ratios apply only to the companies listed on the stock market, which, in our case, are only those listed in the Taiwanese dataset. As mentioned above, there are financial ratios from each category in every dataset; however, some datasets also include variations of some financial ratios. For example, in all datasets except the Bosnia and Herzegovina dataset, there are variations of return on assets, usually expressed as the profit divided by the total assets. However, gross profit, net profit, or profit after depreciation or interest may also be used. Such variations in the same financial ratio lead to a high correlation between attributes. In the Taiwanese dataset, in addition to common financial ratios, growth indicators are used to compare year-to-year changes in profits, assets, equity, and income. Additionally, financial ratios are available up to three years prior to bankruptcy in the Slovak dataset.

Dataset variability allows us to compare the robustness of results for various methods in different countries and timeframes. The method performance is also evaluated for various levels of imbalanced data and different numbers and types of attributes. A general overview of the datasets is presented in Table 2 and in the following subsections.

3.1 Slovak datasets

The two Slovak datasets each consist of 20 attributes and a binary target variable identifying bankrupt and nonbankrupt companies. They are composed of SMEs in the manufacturing and construction industries and cover four years, from 2013 to 2016. The number of data samples varies over the years, and the imbalance ratio is between 0.23% and 2.07%.

3.2 Polish dataset

The Polish dataset consists of 64 attributes and a binary target variable. Financial ratios cover all general categories: liquidity, profitability, and solvency. The dataset covers five years, from 2007 to 2013. In contrast with the Slovak dataset, the Polish dataset does not

Table 2 Overview of all utilized bankruptcy datasets

Country, references	Dataset	Year	Samples	Attributes	Bankrupt class	Non-bankrupt class	Imbalance ratio
Slovakia, Drotár et al. (2019)	SK_M_13	2013	5007	20	30	4077	1 : 135
	SK_M_14	2014	4480	20	30	4450	1 : 148
	SK_M_15	2015	5045	20	26	5019	1 : 193
	SK_M_16	2016	5854	20	14	5840	1 : 417
	SK_C_13	2013	1230	20	25	1205	1 : 48
	SK_C_14	2014	1448	20	30	1418	1 : 47
	SK_C_15	2015	1769	20	20	1749	1 : 87
	SK_C_16	2016	2174	20	14	2174	1 : 155
Poland, Zięba et al. (2016)	PL_01	2007	7027	64	271	6756	1 : 24
	PL_02	2008	10173	64	400	9773	1 : 24
	PL_03	2009	10503	64	495	10008	1 : 20
	PL_04	2010	9742	64	515	9227	1 : 17
	PL_05	2013	5910	64	410	5500	1 : 13
Taiwan, Liang and Tsai (2020)	TW	1999–2009	6819	96	220	6599	1 : 29
Bosnia and Herzegovina, Memic and Memic (2020)	B&H	2012–2014	150	38	50	100	1 : 2

include the identification of individual companies; thus, it is impossible to track the development of a single company over the available years in the dataset. The number of samples varies over the years, and the imbalance ratio is between 3.85% and 6.93%.

3.3 Taiwanese dataset

The Taiwanese dataset consists of 95 attributes and a binary target variable. Compared to other datasets, the Taiwanese dataset includes financial ratios for standard categories (liquidity, profitability, and solvency) and market value ratios since companies in this dataset are listed on the Taiwan Stock Exchange. There are several studies on the Taiwanese dataset, such as Wang and Liu (2021), where the authors used various undersampling methods in combination with some supervised machine learning algorithms. They achieved the highest F2 score with naive Bayes classification in combination with the edited nearest neighbors method. On the other hand, in Gurnani et al. (2021), authors used oversampling methods to handle data imbalance. They achieved the highest F1 score of 98% with RF. In Lin et al. (2019), authors used the Type I error rate as a metric. The best results (zero Type I error rate) were obtained with SVM optimized by the genetic algorithm.

3.4 Bosnia and Herzegovina dataset

The Bosnia and Herzegovina dataset consists of 38 attributes and a binary target variable. Attributes are divided into five categories: liquidity, activity, profitability, leverage, and others. It contains data from 150 companies, of which 100 are not bankrupt and 50 are bankrupt. This dataset has the lowest imbalance ratio. Because Bosnia and Herzegovina do not have a central database of all bankruptcy cases, all regional courts were contacted to

understand the bankruptcy population better. Then, 50 bankruptcy cases were sampled to represent different regions, bankruptcy phases, and sizes.

4 Class imbalance learning methods

Imbalanced data refers to data in datasets where the target class has an uneven distribution of observations compared to other classes. Several approaches are used to learn patterns in imbalanced data. Here, we selected representative methods from different approaches based on various learning algorithms. We aimed to cover a broad range of frequently used methods. Since it is impossible to cover all existing methods, we aimed for comprehensive coverage, and sometimes, we selected similar methods.

We categorized these methods into five groups based on their characteristics. The first group contains methods specially developed for solving outlier detection problems. This group contains some novel algorithms, such as copula-based outlier detection (COPOD), but we also added some conventional approaches, such as the one-class support vector machine (OCSVM). The second group encompasses eight different methods of sampling. Three of these methods utilize oversampling techniques derived from the widely recognized SMOTE method, three employ undersampling strategies, and the remaining two combine both oversampling and undersampling approaches. The third group includes methods based on pure ensemble learning. Ensemble learning is a method of combining multiple different algorithms to obtain more robust results. The fourth group contains methods that apply a combination of multiple techniques. Most methods are based on ensemble learning with a combination of sampling methods. The last group contains only one method that uses a strategy where different misclassification costs are associated with different classes. The considered taxonomy of the class imbalance learning methods is depicted in Fig. 2.

4.1 Outlier detection methods

Outlier detection methods are designed to recognize rare, significant events. They are expected to perform well, especially on highly imbalanced datasets.

4.1.1 Copula-based outlier detector

COPOD Li et al. (2020) is a relatively new probabilistic algorithm for outlier detection. This method is based on copulas. Copulas describe the dependence structure between random variables. They can be defined as cumulative distribution functions for which the marginal distribution of each variable is uniform on the interval $[0, 1]$. COPOD is a three-stage algorithm. In the first stage, the left-tail cumulative distribution function (CDF), right-tail CDF, and skewness coefficients are computed for each feature in the dataset. In the second stage, three copula values for each observation are computed using empirical CDFs. These three values can be interpreted as the left-tail CDF, right-tail CDF, and skewness-corrected probabilities. Anomaly scores are computed in the last stage using the copula values for each observation. This step relies on the fact that smaller tail probabilities result in larger negative-log values. For every observation, summing the negative log of the left and right tails and correcting the empirical copula is essential. The anomaly score is the maximum

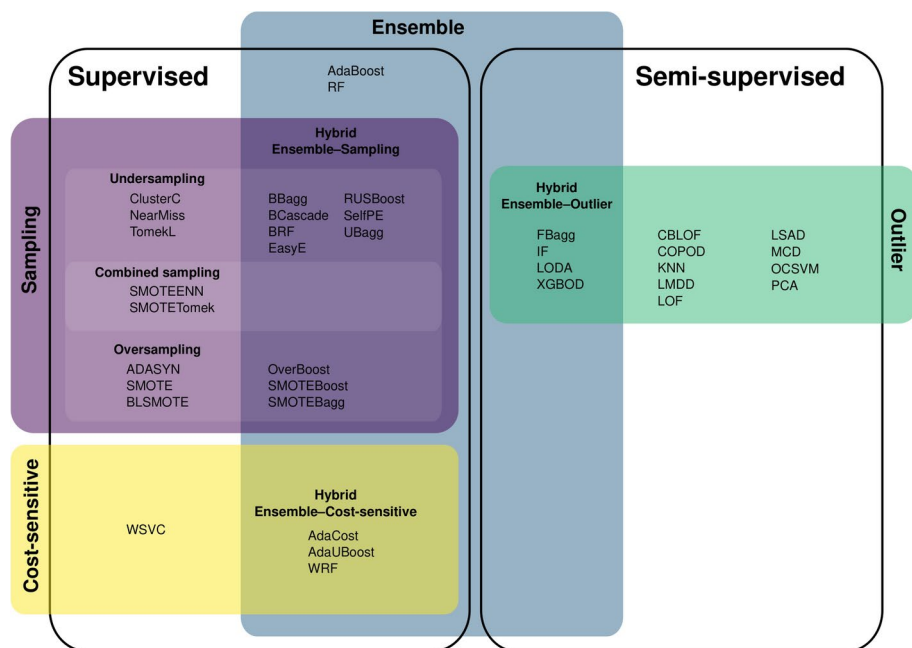


Fig. 2 Taxonomy of the class imbalance learning methods considered in this study

value of these three values. The outlier scores computed in the third stage do not indicate the probability of an observation being an outlier but rather the likelihood of observation relative to other points.

COPOD has some key advantages. It is deterministic and does not utilize hyperparameters. This algorithm is based on empirical CDFs and requires no stochastic training. Moreover, COPOD is interpretable and accessible to visualize using a dimensional outlier graph, and it is efficient in the case of high-dimensional data.

4.1.2 Local outlier factor and cluster-based local outlier factor

The local outlier factor (LOF) is an unsupervised proximity-based anomaly detection algorithm that measures how isolated a point concerning its surrounding neighborhood (Breunig et al. 2000). The LOF is measured by calculating the local density deviation. The local density is determined by estimating the distances between neighboring data points. Data points with less local density than their neighbors are considered outliers.

The cluster-based local outlier factor (CBLOF) is a modification of the LOF that uses K-means clustering (He et al. 2003). Clusters are categorized into small and large clusters. Outlier scores are calculated by using the distances of each sample to its cluster's center multiplied by the number of objects belonging to the cluster. The distance to the closest large cluster is used if a sample belongs to a small cluster.

4.1.3 Minimum covariance determinant

The minimum covariance determinant (MCD) is a linear outlier detection method. It assumes that standard samples are generated from a single Gaussian distribution (Hubert et al. 2018). The MCD uses an elliptic envelope around standard samples to evaluate the parameters of a Gaussian distribution. Next, it searches for the first h samples with the smallest scatter. As a measure of scatter, the determinant of the covariance matrix is used. Computing the MCD is combinatorially complex (Schreurs et al. 2021), so it is primarily feasible for smaller datasets.

4.1.4 One-class support vector machine

An OCSVM is an unsupervised, density-based model for anomaly and outlier detection. It is a derivative of the traditional SVM algorithm. There are two approaches to implementing the OCSVM. The first approach (Schölkopf et al. 1999) is based on separating the samples from the feature space and maximizing the distance from the hyperplane to the origin. This approach results in functions that focus on the space where the probability density is maximum so that the function can return +1 if the observation is in a dense region and -1 if the observation belongs to the low-density space.

The second approach (Tax and Duin 2004) is a spherical, instead of planar, approach. The algorithm creates a hypersphere of minimum volume to cover all the training data. Any sample that falls outside this hypersphere is considered an outlier.

4.1.5 Least-squares approach to anomaly detection

Least-squares anomaly detection (LSAD) (Quinn and Sugiyama 2014) is a probabilistic, nonparametric method for anomaly detection based on a squared-loss objective function. The method is similar to the OCSVM (Schölkopf et al. 1999). LSAD assumes that anomalies occupy the low-density regions of the data space, and a kernel model can be used to characterize the high-density regions given training data.

4.1.6 Linear model deviation-based outlier detection

The linear method for deviation-based outlier detection (LMDD) (Arning et al. 1996) measures the impact of outliers on the data variance, especially how much the variance is reduced when a particular sample is removed. It works with the assumption that outliers lie at the boundary of the data, and removing them significantly reduces the variance. The measure of how much the dissimilarity is reduced when a set of samples E is removed from dataset R is referred to as the smoothing factor. Outliers are defined as exception sets E such that their removal causes the maximum reduction in the data variance.

4.1.7 Principal component analysis

Principal component analysis (PCA) (Pearson 1901) is an unsupervised dimension reduction method where n correlated random variables are transformed into a smaller number of uncorrelated variables (Deepthi 2014). During anomaly detection, PCA measures the

distance of each observation from the center of the data. Samples that are far from the center are considered anomalies.

4.1.8 K-nearest neighbor detector

The K-nearest neighbor (KNN) proximity-based algorithm can be used as an unsupervised anomaly detector (Yang and Huang 2008). The KNN algorithm calculates the k-nearest neighbors from a current sample (data point). This algorithm can use three outlier score approaches: the most significant distance to the k-th neighbor, the average distance to all k neighbors, or the median distance to k neighbors. The final decision on whether the sample is an outlier depends on the chosen threshold, which must be determined experimentally.

4.2 Sampling methods

Sampling methods include machine learning approaches that utilize an oversampling, undersampling, or combined sampling strategy. This group consists of eight methods: three oversampling approaches, three undersampling approaches, and two methods combining previous approaches. In general, sampling methods are considered to be a data preprocessing approach. Accordingly, the sampling techniques were consistently assessed alongside SVM using a nonlinear radial basis function kernel and extreme gradient boosting (XGBoost) (Chen and Guestrin 2016).

4.2.1 SMOTE and related methods

SMOTE (Chawla et al. 2002) is a minority class data augmentation approach. The idea behind this technique is to select examples close to the feature space, drawing a separating line (decision boundary) between the samples in the feature space and drawing a new sample at a point along that line. The first step is to choose a random sample from the minority class. Then, k of the nearest neighbors for that sample are found, and one of the neighbors is chosen. A synthetic sample is created at a randomly selected point between the two samples in the feature space.

Borderline-SMOTE (BLSMOTE) (Han et al. 2005) is a SMOTE variation where synthetic samples are generated only along the decision boundary between the two classes.

Adaptive synthetic sampling (ADASYN) (He et al. 2008) is a SMOTE modification that generates more synthetic samples in regions of the feature space in which the density of minority samples is low. In contrast, only a few or no samples are generated in high-density regions.

SMOTETomek (Batista et al. 2003) is an extension of the SMOTE oversampling algorithm. The basic idea involves combining both oversampling and undersampling techniques to generate representative synthetic samples. First, new minority class samples are synthetically generated using the SMOTE algorithm to balance the original dataset. Then, the dataset is cleaned using the Tomek links (TomekL) undersampling algorithm (Tomek 1976) to reduce the class overlap and remove noisy samples.

SMOTE combined with an edited nearest neighbor (ENN) algorithm denoted as SMO-TEENN (Le 2022) is another sampling approach that combines both oversampling and undersampling strategy. Here, after balancing the original dataset by the SMOTE algorithm,

the ENN approach is used to iteratively check each sample in the dataset to see if its k -nearest neighbors correctly classify it. If a majority of its neighbors misclassify such a sample, then it is removed from the dataset.

4.2.2 Near miss

Near miss (NearMiss) (Mani and Zhang 2003) is a KNN-based undersampling algorithm. It uses the class distribution and random elimination of samples from the majority class. When samples in two different classes are very close to each other, the samples in the majority class are removed to increase the spaces between the two classes. NearMiss has three modifications. NearMiss-1 removes the majority class examples with a minimum average distance to the three closest minority class examples. In NearMiss-2, majority class examples with a minimum average distance to the three furthest minority class examples are removed. The last modification is named NearMiss-3, and in this approach, the majority class examples with the minimum distance to each minority class example are removed.

4.2.3 Tomek links

The TomekL algorithm (Tomek 1976) is an undersampling algorithm based on the condensed nearest neighbors rule. This algorithm finds pairs of samples from different classes with the smallest distance in the feature space. These pairs are referred to as Tomek links. The majority class sample is removed from each pair, which provides a better decision boundary for a classifier.

4.2.4 Cluster centroid-based majority undersampling technique

The cluster centroid-based majority undersampling technique (ClusterC) is a technique that removes the most unimportant samples or samples with less information from the majority class (Pamula et al. 2011). It uses the concept of finding the cluster centroid, where clusters are created by encircling the majority class. The cluster centroid is found by averaging feature vectors for all the features over the samples from the majority class in the feature space. The samples from the majority class farthest from the cluster centroid are considered the most unimportant and are thus removed.

4.3 Ensemble methods

The main idea of ensemble methods is to use multiple base estimators to build a robust classifier. They have achieved excellent results in many machine learning problems Lin et al. (2019), Liang et al. (2018), Liu et al. (2020). The experimental study utilized DT as a base learner in ensemble-based methods.

4.3.1 Random forest

RF (Ho 1995) is an extension of the bootstrap aggregation (bagging) of DTs. Each tree is created from a different bootstrap sample of the training dataset. The algorithm establishes the outcome based on the predictions of the DTs. It makes predictions by averaging the

outputs from various trees. Increasing the number of trees increases the precision of the outcome.

4.3.2 Adaptive boosting

Adaptive boosting (AdaBoost) was one of the first boosting algorithms proposed by Freund and Schapire (1995). It combines multiple base estimators into a single robust classifier. The base estimators are usually DTs with a single split, called decision stumps. AdaBoost iteratively builds an ensemble of base estimators by adjusting the weights of misclassified data during each iteration. For each subsequent base learner, the weights are recalculated such that higher weights are assigned to samples that the current base estimator misclassifies.

4.4 Hybrid methods

Hybrid methods exploit a combination of different approaches. In our case, it mainly combines ensemble methods with sampling techniques. Almost all of the ensemble-based methods use DT as a base estimator.

4.4.1 Random forest-based hybrid techniques

Two modifications of the original RF are much more suitable for imbalanced data. The first is the balanced random forest (BRF) (Chen and Breiman 2004). In every iteration, a training set is created that consists of bootstrap samples from the minority class and the same number of samples, with replacements, from the majority class. Using a classification and regression tree algorithm (CART) (Breiman et al. 1984), a classification tree without pruning is created from the data and grows to its maximum size. A BRF approach can be considered a hybrid technique, but for the sake of consistency, we decided to group it with other RF techniques.

The second modification is the weighted random forest (WRF). This method mitigates the problem of traditional RF being biased toward the majority class. The idea behind the algorithm to mitigate this problem is the implementation of cost-sensitive learning. A heavier penalty is imposed for misclassifying samples from the minority class.

4.4.2 EasyEnsemble and BalanceCascade

EasyEnsemble (EasyE) (Liu et al. 2008) and BalanceCascade (BCascade) (Liu et al. 2008) are undersampling algorithms. Let S be the dataset to be classified, P be the minority training set, and N be the majority training set. The undersampling method randomly samples N' from N , where $|N'| < |N|$. Selecting only one subset of the majority class has a drawback because many potentially useful data are discarded. Using the EasyE method, we can minimize the risk of losing useful data by creating T subsets N_1, N_2, \dots, N_T from N . For each subset N_i (expanded by minority subset P), the classifier H_i is trained. All of the classifiers are combined using AdaBoost.

BCascade is a modification of the EasyE algorithm. If the trained classifiers correctly classify the majority class instances, BCascade removes them from further consideration.

4.4.3 Self-paced ensemble

The self-paced ensemble (SelfPE) (Liu et al. 2020) is based on combining self-paced learning with undersampling to generate a robust ensemble. In self-paced learning, machine learning models, usually DTs, learn at their own pace, considering samples with high hardness first. The hardness of a sample can be interpreted as the probability that the sample is a noisy instance, an outlier, or both. Using the hardness value, the samples from the majority class are split into k bins. Then, these instances are undersampled while retaining the total hardness contribution in each bin. By balancing the hardness contributions, the sample probabilities in the bins with larger populations will decrease (Ding et al. 2023). Initially, the algorithm tends to focus more on outlier samples to improve the model's generalizability. In later iterations, the majority class samples will be more important to prevent overfitting.

4.4.4 Isolation forest

An isolation forest (IF) (Liu et al. 2008) model is an unsupervised anomaly detection technique using DTs similar to RF (Ho 1995). A random feature and a random split value are selected when DTs are constructed. The split value is between a selected feature's minimum and maximum values. This creates sample partitions. Different partitions are separated at the tree's root and deeper into the branches, and subtler distinctions are identified. Then, for prediction, a sample's split value is compared against a node's split value. The number of splits is referred to as the path length. Samples that require more splits have a small probability of being outliers. Samples found on shorter branches are more likely to be outliers since it is simpler to distinguish these samples from the other samples.

4.4.5 Feature bagging

Feature bagging (FBagg) (Lazarevic and Kumar 2005) is an anomaly detection ensemble-based technique in which only a small subset of randomly selected features is used to detect outliers. Each outlier detection classifier in the ensemble assigns an outlier score to all samples. The sample score corresponds to the probability of it being an outlier. Thus, each outlier detection classifier identifies a different set of outliers. The scores from several classifiers are combined to obtain the final outlier score. Importantly, this model uses LOF as a base estimator.

4.4.6 Lightweight online detector of anomalies

The lightweight online detector of anomalies (LODA) approach (Pevný 2016) is an ensemble of k one-dimensional histogram density estimators. Each estimator approximates the probability density of input data projected onto a single projection vector. For every estimator, \sqrt{d} random features are selected, where d is the total number of features. To compute the anomaly score for a particular sample, LODA computes its average log density. LODA is particularly useful when a large amount of data is processed in real time.

4.4.7 Extreme gradient boosting outlier detection

Extreme gradient boosting outlier detection (XGBOD) (Zhao and Hryniewicki 2018) is a semisupervised anomaly detection algorithm. It is a three-phase algorithm. In the first phase, outlier scores are computed using unsupervised anomaly detection methods. These newly computed scores can be interpreted as new augmented features. These new features are used in the second phase, where they are combined with the original features. In this phase, the XGBoost classifier with DTs as base estimators is trained using new data. The output of the XGBoost classifier is considered a result of the entire method.

4.4.8 Balanced bagging and SMOTE bagging

Balanced bagging (BBagg) is a version of the bagging algorithm that uses a random undersampling strategy on the majority class within a bootstrap sample to balance the two classes. BBagg can provide a platform for more specific balancing algorithms, including exact BBagg (Opitz and Maclin 1997), rough BBagg (Hido et al. 2009), overbagging (Opitz and Maclin 1997), and SMOTE bagging (SMOTEBagg) (Chawla et al. 2002).

SMOTEBagg is a supervised anomaly detection technique that is a combination of SMOTE (Chawla et al. 2002) and bagging (Breiman 1996). It uses SMOTE to generate synthetic minority samples until the number of minority samples is equivalent to the number of majority samples. Synthetic data are generated based on the characteristics of the object and k-nearest neighbor. The purpose of the bagging step in this algorithm is to reduce the variance of classifiers.

4.4.9 Boost-based outlier detection techniques

In this paper, we used three modifications of the previously mentioned AdaBoost, namely, SMOTEBoost (Chawla et al. 2003), random undersampling boosting (RUSBoost) (Seiffert et al. 2009) and AdaCost (Fan et al. 1999). SMOTEBoost is an oversampling method that uses the SMOTE algorithm to oversample minority classes at each boosting iteration. Conversely, RUSBoost employs a random undersampling method to balance the majority class samples.

Another algorithm similar to SMOTEBoost is OverBoost. The main difference between these two algorithms is the sampling technique they apply. OverBoost uses random oversampling instead of SMOTE. Random oversampling duplicates examples from the minority class in the training dataset.

The last boost-based algorithm we used is the misclassification cost-sensitive boosting method named AdaCost. It uses the cost of misclassified samples to update the training distribution in successive boosting rounds. The goal is to reduce the cumulative misclassification cost compared to AdaBoost.

4.4.10 AdaUBoost

The Leskovec and Shawe-Taylor (2003) algorithm is based on AdaBoost, with a modified unequal loss function and modified weight updating rule. The parameter β is introduced as Karakoulas and Shawe-Taylor (1998). It forces uneven misclassification costs for train-

ing examples. Samples from the minority class are assigned β times higher initial weights than those of samples from the majority class. The parameter β is also used in the weight updating rule. The weights of false-negative samples are increased more than those of false-positive samples, and the weights of true-positive samples are decreased more than those of true-negative samples. This leads to the fact that each base estimator tends to correctly classify more samples from the minority class because they maintain higher weights. The result of AdaUBoost is a linear combination of the base estimators.

4.4.11 Under bagging

An ensemble of DT classifiers creates the under bagging (UBagg) classifier (Opitz and Maclin 1997). When creating classifiers, the training subset for each classifier is constructed by undersampling the majority class. After construction, the majority vote principle is applied to a new instance. The final classification decision follows the most voted class.

4.5 Cost-sensitive methods

Generally, cost-sensitive learning involves reweighting during the model's training, enabling the incorporation of misclassification costs directly into the learning process. The goal is to minimize the cost incurred by misclassifications rather than simply minimizing the error rate. In our specific case, this group includes just one method.

4.5.1 Weighted support vector machine

The weighted support vector machine classifier (WSVM) refers to a modification of the standard SVM where different weights are assigned to different classes or even specific instances (Yang et al. 2005). This is particularly useful in scenarios where there is an imbalance in the class distribution or certain misclassifications are considered more costly than others.

5 Experimental settings

To evaluate the efficiency of the selected machine learning approaches, we utilized five real-world bankruptcy datasets originating from different countries (economies): the Slovak Republic, Bosnia and Herzegovina, Taiwan, and Poland. Before applying the imbalance learning algorithms, it was necessary to perform data cleaning operations to prepare all the datasets. First, the missing values were replaced with the average value of the particular economic attribute. Second, data were standardized per feature to have a zero mean and unit variance.

We used fivefold stratified cross-validation in the supervised machine learning approaches to validate the model performance. Semisupervised learning algorithms were trained only using the samples from the majority class (nonbankrupt) in a fivefold cross-validation manner. All the samples from the minority class (bankrupt) were exclusively utilized in the testing phase.

To tune the performance of the utilized machine learning approaches, we searched through the grid of hyperparameters. These are summarized in Appendix B. We integrated sampling approaches with SVM and XGBoost classifiers. In the case of SVM, we searched through a grid defined by kernel coefficient $\gamma = [0.001, 0.01, 0.1, 1, 10, 100]$ and regularization parameter $C = [0.01, 0.1, 1, 5, 10, 100]$. The grid of hyperparameters of XGBoost was defined by step size shrinkage parameter $\text{learning_rate} = [0.001, 0.01, 0.1, 0.3, 0.5, 0.7, 0.9]$, $\text{max_depth} = [3, 5, 10, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500]$, $\text{min_child_weight} = [1, 3, 5, 10]$ and $\gamma = [0.001, 0.01, 0.1, 1, 10, 100]$. For the remaining hyperparameters, default values specific to each classifier were utilized.

Choosing the right evaluation metric to measure the performance of models derived from imbalanced data is one of the most important steps. Kuizinienė et al. (2022) analyzed the most commonly used evaluation metrics for bankruptcy prediction. The most common metrics were accuracy and area under the receiver operating characteristic curve. Here, the geometric mean (GM) was utilized to measure the model performance since it is considered one of the most reliable techniques when facing issues represented by imbalanced data Al Helal et al. (2016). It can be expressed as a square root of the product of sensitivity and specificity. Sensitivity expresses the proportion of correctly predicted positive cases. The proportion of correctly predicted negative cases is represented by specificity. The GM is defined as follows:

$$GM = \sqrt{\text{sensitivity} \times \text{specificity}} = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}}, \quad (1)$$

where TP (true positive) is the number of correctly predicted positive cases, FN (false negative) is the number of incorrectly predicted positive cases, TN (true negative) is the number of correctly predicted negative cases, and FP (false positive) expresses the number of incorrectly predicted negative cases. In addition to the GM, we reported the results measured by the AUC metric and the error of incorrectly classified bankrupt cases into non-bankrupt class measured by FNR. However, we provided these results for readability in Appendix C and Appendix D, respectively.

6 Experimental evaluation

In this section, we offer a concise evaluation of the results obtained from our experimental study, analyzing the outcomes from various perspectives, such as a diverse number of financial attributes, different data origins, various data distributions, and different dataset sizes. Through our experiments, we aimed to gain insights into the performance of various imbalanced learning approaches on the given datasets and identify the best-performing models for predicting bankruptcy. To evaluate the effectiveness of the selected machine learning approaches, we conducted experiments using five real-world bankruptcy datasets from different economies: the Slovak Republic, Bosnia and Herzegovina, Taiwan, and Poland. The datasets represent diverse business environments and economic conditions. This allowed us to evaluate the robustness of the analyzed imbalanced learning methods across different contexts.

6.1 Effect of dataset origin

In Tables 3, 4, 5, 6, we provide results for the Slovak, Polish, Taiwan, and Bosnia and Herzegovina datasets. We start by evaluating the results with single datasets. The Slovak manufacturing bankruptcy datasets (SK_M_13, SK_M_14, SK_M_15, SK_M_16) are characterized by extremely high imbalanced ratios, ranging from 1:135 to as high as 1:417 for the most imbalanced dataset. In this scenario, conventional classifiers that can not handle data imbalance almost completely fail to predict the minority classes. On the other hand, as we can see from Table 3, most of the ensemble-based methods generate auspicious results. We want to highlight the best-performing methods: BCascade, SelfPE, and RUSBoost. These hybrid methods are combinations of ensemble and undersampling approaches. Their average GMs over the four considered years were the highest. Combined sampling approaches, namely SMOTEENN+XGBoost and SMOTETomek+XGBoost, also demonstrated very promising results. Furthermore, SMOTEENN+XGBoost slightly outperformed all other utilized methods on the SK_M_13 and SK_M_15 datasets, achieving GM scores of 89.9% and 92.2%, respectively.

Similar trends as in the Slovak manufacturing dataset were observed in the Slovak construction bankruptcy datasets (SK_C_13, SK_C_14, SK_C_15, SK_C_16). On average, the best results in terms of the GM score were obtained by the BCascade classifier, followed by the SelfPE and BRF methods. Several other hybrid methods that combine ensemble and sampling techniques, such as BBagg, EasyE, UBagg, and RUSBoost, provided very competitive results that lagged only a few percentage points behind the best results of the BCascade approach. Similarly, XGBoost-based combined sampling approaches, specifically SMOTEENN+XGBoost and SMOTETomek+XGBoost, achieved comparable results. Moreover, the highest average GM score among all sampling algorithms was achieved by the ClusterC+XGBoost approach scoring over 87%. Other sampling scores were notably lower than those achieved by a combination of ensemble and undersampling methods such as BCascade, SelfPE, RUSBoost, UBagg, and OverBoost. The performance of the outlier detection approaches did not match that of the ensemble-based methods, even though the high imbalance ratio of these datasets made the problem very similar to the outlier detection problem. Here, the outlier detection methods include semisupervised and unsupervised methods. Since semisupervised learning is more challenging, we can expect some drop in performance when compared to supervised methods.

Ensemble methods (AdaBoost, RF) and the cost-sensitive learning method (WSVM) failed to identify bankrupt companies with GM scores not exceeding 47%.

Although it is not the most imbalanced dataset, the classification task for the Polish dataset is the most challenging one. The results showed that the best-performing methods were hybrid methods utilizing ensemble and sampling approaches, specifically, the BRF and EasyE and combined sampling approach SMOTEENN applied with the XGBoost classifier. These methods' results were substantially better than those of all other methods, reaching an average GM score of 82% and more.

The imbalance ratio, 1:29, of the Taiwanese dataset (TW) is significantly lower than that of the Slovak datasets and similar to that of the Polish datasets. The performance of the methods aligns with the results achieved on the Slovak datasets. In this case, superior results were obtained for both sampling methods and hybrid methods, which integrate ensembles with either sampling or a cost-sensitive approach. BRF obtained the highest GM score, fol-

Table 3 Overview of the GM scores (%) achieved on the Slovak manufacture bankruptcy datasets in the form of GM_{std}

Category	Method	Data				Avg.
		SK_M_13	SK_M_14	SK_M_15	SK_M_16	
Sampling	ADASYN+SVM	73.1 ₁₁	74.9 ₁₀	81.2 ₈	83.7 ₁₁	78.2 ₁₀
	ADASYN+XGBoost	81.9 ₅	81.2 ₈	90.5 ₇	59.6 ₃₁	78.3 ₁₃
	BLSMOTE+SVM	64.3 ₂₀	68.3 ₁₇	71.9 ₁₆	53.3 ₂₈	64.4 ₁₇
	BLSMOTE+XGBoost	73.5 ₁₀	74.6 ₁₅	72.7 ₁₂	57.0 ₃₃	69.4 ₁₇
	SMOTE+SVM	73.1 ₁₁	73.8 ₁₀	81.1 ₈	83.8 ₁₁	77.9 ₁₀
	SMOTE+XGBoost	81.9 ₅	80.9 ₈	90.4 ₇	63.7 ₃₅	79.3 ₁₄
	SMOTETomek+SVC	73.2 ₁₁	73.9 ₁₀	81.2 ₈	83.7 ₁₁	77.9 ₁₀
	SMOTETomek+XGBoost	82.7 ₅	84.9 ₃	90.7 ₇	67.4 ₃₇	81.5 ₁₃
	SMOTEENN+SVC	74.5 ₁₂	82.5 ₇	85.5 ₆	85.4 ₁₁	81.8 ₉
	SMOTEENN+XGBoost	83.9 ₅	85.3 ₇	92.2 ₅	73.9 ₁₀	83.9 ₇
	ClusterC+SVM	63.2 ₈	53.9 ₁₀	58.2 ₅	74.4 ₁₅	62.4 ₉
	ClusterC+XGBoost	78.2 ₆	83.6 ₄	88.2 ₅	89.3 ₉	84.9 ₂₂
	NearMiss+SVM	69.3 ₁₁	59.7 ₁₆	60.8 ₂₀	85.4 ₈	68.8 ₁₄
	NearMiss+XGBoost	77.7 ₇	78.4 ₄	82.7 ₆	82.4 ₁₂	80.3 ₇
Ensemble	TomekL+SVM	8.2 ₁₇	8.2 ₁₇	8.9 ₁₈	0.0 ₀	6.3 ₆
	TomekL+XGBoost	27.9 ₂₄	39.4 ₂₁	43.9 ₂	25.7 ₃₂	34.2 ₂₀
Outlier	AdaBoost	27.9 ₂₄	32.7 ₁₇	21.6 ₂₈	23.1 ₂₉	26.3 ₂₄
	RF	16.4 ₂₀	27.9 ₂₃	26.9 ₂₂	11.6 ₂₃	20.6 ₂₂
	CBLOF	73.3 ₄	70.6 ₅	69.6 ₅	80.5 ₃	73.5 ₄
	COPOD	79.6 ₁	79.5 ₂	72.9 ₁	86.7 ₂	79.7 ₂
	KNN	74.1 ₃	74.6 ₃	69.7 ₂	84.0 ₃	75.6 ₃
	LMDD	43.8 ₉	39.1 ₂₃	55.5 ₁₂	53.1 ₁₉	47.8 ₁₆
	LOF	69.9 ₃	71.8 ₅	74.6 ₃	84.5 ₂	75.2 ₃
	LSAD	73.3 ₃	73.9 ₄	71.0 ₂	86.9 ₂	76.2 ₃
	MCD	68.0 ₁	78.5 ₂	71.9 ₈	75.6 ₃	73.5 ₄
	OCSVM	37.6 ₆	34.8 ₄	37.6 ₄	33.9 ₅	35.9 ₄
Cost-sensitive	PCA	64.1 ₆	70.6 ₄	67.3 ₄	80.5 ₃	70.6 ₄
	WSVM	16.4 ₂₀	32.7 ₁₇	30.6 ₂₆	11.6 ₂₃	22.8 ₂₂
Hybrid (Ensemble, Sampling)	BBagg	81.5 ₅	84.8 ₁₀	87.6 ₈	85.7 ₁₆	84.9 ₁₀
	BCascade	80.1 ₇	87.2 ₆	90.1 ₄	92.2 ₇	87.4 ₆
	BRF	82.4 ₇	86.1 ₄	87.5 ₄	88.6 ₈	86.1 ₆
	EasyE	81.1 ₅	88.7 ₅	84.8 ₈	87.1 ₈	85.4 ₆
	RUSBoost	81.9 ₅	89.0 ₇	87.7 ₅	87.9 ₉	86.6 ₆
	SelfPE	83.2 ₆	86.8 ₄	90.2 ₄	88.6 ₁₀	87.2 ₆
	UBagg	81.5 ₅	84.8 ₁₀	87.6 ₈	85.7 ₁₆	84.9 ₁₀
	OverBoost	78.9 ₄	86.5 ₇	89.5 ₅	81.3 ₁₇	84.0 ₈
	SMOTEBoost	79.9 ₆	83.1 ₈	89.5 ₇	84.7 ₁₂	84.3 ₈
	SMOTEBagg	16.4 ₂₀	16.4 ₂₀	9.0 ₁₈	11.6 ₂₃	13.3 ₁₈
Hybrid (Ensemble, Cost-sensitive)	AdaCost	78.1 ₇	84.4 ₅	91.7 ₄	83.3 ₁₄	84.3 ₈
	AdaUBoost	78.6 ₄	84.9 ₁₀	89.5 ₅	81.4 ₁₇	83.6 ₉
Hybrid (Ensemble, Outlier)	WRF	71.6 ₁₀	72.8 ₅	80.3 ₅	82.3 ₁₁	76.7 ₈
	FBagg	74.4 ₂	81.0 ₁	75.5 ₃	87.0 ₂	79.4 ₂
	IF	77.4 ₂	77.8 ₃	71.4 ₃	85.7 ₁	78.1 ₂

Table 3 (continued)

Category	Method	Data				Avg.
		SK_M_13	SK_M_14	SK_M_15	SK_M_16	
	LODA	60.2 ₆	50.8 ₈	57.9 ₆	62.4 ₁₂	57.8 ₈
	XGBOD	51.6 ₁₈	73.0 ₆	80.5 ₄	75.6 ₈	70.2 ₉

lowed closely by the scores obtained by SMOTEENN+XGBoost and EasyE. Interestingly, on the Taiwanese dataset, oversampling and combined sampling approaches are competitive with the hybrid ensemble-based methods.

Contrary to the previously mentioned datasets, the dataset from Bosnia and Herzegovina (B&H) is the least imbalanced dataset, characterized by a proportional imbalance ratio of 1:2. This makes the outlier detection-based methods unsuitable for the application, as confirmed by the results. Here, the best result in terms of GM score (over 96%) was achieved by WSVM, the pure cost-sensitive approach. Next, the AdaBoost and SMOTETomek+XGBoost methods scored over 95%. Other methods based on ensemble, cost-sensitive, and sampling techniques yielded GM scores above 90%.

The results for all datasets expressed by the AUC and FNR scores are provided in Appendices C and D, respectively.

6.2 Effect of the dataset imbalance ratio

We also investigated how the dataset imbalance ratio influences the prediction performance of different methods. The imbalance ratio is defined as the ratio of the number of samples in the minority class to the number of samples in the majority class. We again used the datasets described in Sect. 3 for this evaluation. The imbalance ratio ranges from 1:417 to 1:13 for the 14 datasets. The results for all 45 methods are depicted in Fig. 3. Many outlier detection methods showed a negative trend with an increasing imbalance ratio. This confirms our hypothesis that outlier detection methods are more suitable for highly imbalanced scenarios since they are more similar to outlier detection scenarios. In outlier detection, we usually assume that outliers rarely occur, similar to the occurrences of data points from minority classes in extremely imbalanced data. The best-performing methods, such as BRF, SMOTEENN+XGBoost, EasyE, RUSBoost, and UBagg, showed steady performance for different imbalance ratios. It should be noted that even though we present results as a function of the imbalance ratio, the imbalance ratio is not the sole factor affecting prediction outcomes. Other elements, such as the features considered, analyzed period, or country specifics, can significantly influence prediction performance.

6.3 Results summary

Figure 4 shows the aggregated prediction performance of all methods over all datasets. For the Slovak and Polish datasets, we reported the average performance across the years covered by the dataset. As shown in Fig. 4, the top 10 best-performing methods are all ensemble-based. Therefore, it appears that ensemble learning is a way to address imbalanced data. However, we should note that these are not pure ensemble methods but hybrid methods that combine ensemble techniques with other approaches for imbalanced learning, such as sampling and cost-sensitive learning. BRF, EasyE, SMOTEENN+XGBoost, RUS-

Table 4 Overview of the GM scores (%) achieved on the Slovak construction bankruptcy datasets in the form of GM_{std}

Category	Method	Data				Avg.
		SK_C_13	SK_C_14	SK_C_15	SK_C_16	
Sampling	ADASYN+SVM	65.3 ₄	81.1 ₃	85.8 ₆	85.0 ₁₁	79.3 ₆
	ADASYN+XGBoost	83.8 ₁₅	81.9 ₁₀	84.6 ₁₀	67.4 ₁₆	79.5 ₁₂
	BLSMOTE+SVM	71.1 ₁₀	76.2 ₈	85.3 ₁₁	80.4 ₁₆	78.2 ₁₁
	BLSMOTE+XGBoost	78.9 ₁₂	84.7 ₁₀	88.1 ₆	50.2 ₄₂	75.5 ₁₈
	SMOTE+SVM	65.1 ₃	81.1 ₇	85.8 ₆	85.3 ₁₁	79.3 ₇
	SMOTE+XGBoost	84.5 ₁₀	82.9 ₉	85.7 ₁	72.2 ₁₅	81.3 ₉
	SMOTETomek+SVC	66.4 ₆	81.8 ₄	83.2 ₁₁	85.7 ₁₁	79.3 ₈
	SMOTETomek+XGBoost	88.9 ₉	85.3 ₆	89.7 ₇	77.0 ₁₄	85.2 ₉
	SMOTEENN+SVC	74.5 ₁₂	82.5 ₇	85.5 ₆	85.4 ₁₁	81.9 ₉
	SMOTEENN+XGBoost	87.5 ₉	86.5 ₆	92.1 ₇	76.9 ₁₄	85.7 ₉
	ClusterC+SVM	45.3 ₁₃	58.7 ₁₀	79.8 ₆	70.6 ₁₁	63.6 ₁₀
	ClusterC+XGBoost	81.2 ₈	83.7 ₂	89.4 ₃	95.3 ₁	87.4 ₄
	NearMiss+SVM	67.1 ₇	78.4 ₈	84.9 ₇	81.4 ₁₀	77.9 ₈
	NearMiss+XGBoost	78.9 ₈	79.7 ₃	92.9 ₅	66.1 ₂₁	79.4 ₉
	TomekL+SVM	9.0 ₁₈	32.5 ₁₇	8.0 ₁₁	34.7 ₂₈	21.5 ₁₆
	TomekL+XGBoost	50.5 ₂₆	52.2 ₁₆	48.3 ₂₆	32.7 ₄₀	45.9 ₂₇
Ensemble	AdaBoost	46.6 ₂₅	42.7 ₂₃	34.1 ₂₉	27.9 ₃₅	37.8 ₂₈
	RF	30.5 ₂₆	36.0 ₃₂	24.2 ₃₁	23.1 ₂₉	28.4 ₂₉
Outlier	CBLOF	65.7 ₅	70.9 ₂	83.0 ₂	80.2 ₃	74.9 ₃
	COPOD	73.2 ₂	76.1 ₃	84.0 ₃	87.0 ₃	80.0 ₃
	KNN	67.8 ₂	71.3 ₃	85.3 ₂	89.9 ₂	78.5 ₂
	LMDD	38.2 ₁₅	26.9 ₁₇	64.1 ₇	36.5 ₃₁	41.4 ₁₈
	LOF	66.6 ₄	73.2 ₄	80.2 ₂	89.2 ₂	77.3 ₃
	LSAD	86.9 ₂	75.4 ₂	86.3 ₃	88.9 ₄	80.0 ₃
	MCD	71.1 ₃	61.1 ₃	67.6 ₂	63.0 ₄	65.7 ₃
	OCSVM	33.9 ₅	36.7 ₅	45.1 ₄	32.6 ₁	37.9 ₄
Cost-sensitive	PCA	66.1 ₄	66.7 ₅	81.8 ₂	78.1 ₄	73.1 ₄
	WSVM	48.4 ₈	54.9 ₁₈	34.2 ₂₉	27.9 ₃₅	41.3 ₂₂
Hybrid (Ensemble, Sampling)	BBagg	85.7 ₁₁	85.2 ₉	88.5 ₁₂	93.0 ₈	88.1 ₁₀
	BCascade	87.0 ₈	86.2 ₆	95.7 ₁	95.3 ₂	91.0 ₄
	BRF	85.8 ₉	85.1 ₅	90.9 ₇	92.7 ₇	88.6 ₇
	EasyE	87.3 ₉	84.4 ₃	91.8 ₆	89.0 ₁₁	88.1 ₇
	RUSBoost	84.4 ₁₀	85.6 ₄	89.9 ₇	90.5 ₁₁	87.6 ₈
	SelfPE	87.2 ₉	87.3 ₅	92.3 ₆	96.4 ₁	90.7 ₅
	UBagg	85.7 ₁₁	85.2 ₉	88.5 ₁₁	93.0 ₈	88.1 ₁₀
	OverBoost	83.6 ₁₅	84.9 ₆	89.1 ₇	88.0 ₁₂	86.4 ₁₀
	SMOTEBoost	85.6 ₁₀	84.1 ₈	89.4 ₇	88.1 ₁₂	86.8 ₉
	SMOTEBagg	39.4 ₂₁	48.9 ₁₇	24.2 ₃₁	16.4 ₃₃	32.2 ₂₅
Hybrid (Ensemble, Cost-sensitive)	AdaCost	83.9 ₁₃	81.7 ₆	88.0 ₆	85.5 ₁₁	84.8 ₉
	AdaUBoost	84.3 ₁₀	84.8 ₅	88.8 ₇	88.1 ₁₂	86.4 ₈
	WRF	70.2 ₆	82.7 ₆	85.7 ₁₁	84.9 ₁₁	80.8 ₉
Hybrid (Ensemble, Outlier)	FBagg	68.5 ₅	74.6 ₂	82.0 ₂	89.9 ₂	78.7 ₃
	IF	73.9 ₃	71.3 ₂	85.0 ₃	81.5 ₅	77.9 ₃

Table 4 (continued)

Category	Method	Data				Avg.
		SK_C_13	SK_C_14	SK_C_15	SK_C_16	
	LODA	53.5 ₈	57.7 ₈	67.6 ₄	65.5 ₁₂	61.0 ₈
	XGBOD	82.7 ₄	72.4 ₇	73.2 ₉	76.4 ₄	76.1 ₆

Boost, SMOTETomek+XGBoost, and UBagg achieved the highest overall GM scores. All these methods share two common approaches: ensemble learning and data undersampling. The next high-ranking method is OverBoost utilizing oversampling with ensemble learning. AdaUBoost, which also ranks within the top 10 best-performing methods, represents a combination of an ensemble model and cost-sensitive learning. On the other hand, outlier detection methods failed to deliver satisfactory results, even in highly imbalanced scenarios. We hypothesize that the outlier methods are very dataset-specific and may overfit a particular dataset. These findings align with the conclusions drawn in a recent paper Wu and Keogh (2023).

We employed a two-stage statistical evaluation process to assess the performance differences and identify the most effective machine learning approaches. Initially, we utilized Friedman's chi-square test (Friedman 1937), a non-parametric statistical test, to rank utilized classifiers based on their performance metric (GM score) across all datasets. Following Friedman's chi-squared test, to further investigate the specific differences between classifiers, we applied the concept of the critical difference (CDiff) (Demšar 2006). The CDiff is a post-hoc statistical analysis technique used to determine whether the performance differences between pairs of classifiers are statistically significant. By calculating the CDiff score, we can visually and numerically identify groups of classifiers whose performances do not differ significantly, providing a clearer understanding of which classifiers consistently outperform others and which exhibit similar levels of prediction performance.

Combining Friedman's chi-square test with the calculation of the CDiff score, this comprehensive statistical approach allowed us to systematically compare and rank the effectiveness of a wide range of classifiers, considering the variability and complexities inherent in multiple datasets. Through this methodical evaluation, we aimed to offer insights into the performance characteristics of various classifier classes, highlighting the strengths and limitations of each in the context of diverse data challenges. It is crucial to emphasize that a lower CDiff score reflects the robustness and effectiveness of the measured method. The calculated p-value of the chi-square test was 0.05. Comparative analysis of multiple classifiers across various datasets is depicted in Table 7.

Overall, the top 10 best CDiff scores highlight the potential of hybrid methods to build effective and robust models across various datasets. Here, the best results were achieved by combining ensemble learning with sampling or cost-sensitive approaches with CDiff scores ranging from 5.26 to 12. The most consistent and significantly best-performing method in terms of CDiff score across various datasets among the utilized classifiers was achieved with the BRF method with a CDiff score of 5.26, followed by EasyE and SMOTEENN+XGBoost with equal CDiff scores of 6.5. A very promising CDiff score was also achieved by the RUSBoost method with a score of 7. In contrast, integrating pure cost-sensitive learning or pure ensemble methods resulted in CDiff scores ranging from 20.1 to 35.9, demonstrating varied success. These approaches emphasize the complexity of achieving consistent performance across different datasets and the need for meticulous model selection and tuning. Further-

Table 5 Overview of the GM scores (%) achieved on the polish bankruptcy datasets in the form of GM_{std}

Category	Method	Data					Avg.
		PL_01	PL_02	PL_03	PL_04	PL_05	
Sampling	ADASYN+SVM	78.2 ₆	72.1 ₃	73.4 ₂	70.1 ₂	75.8 ₂	73.9 ₃
	ADASYN+XGBoost	81.6 ₃	75.4 ₃	77.3 ₄	77.9 ₃	84.6 ₂	79.4 ₃
	BLSMOTE+SVM	75.0 ₅	69.1 ₄	69.6 ₃	68.3 ₂	75.3 ₄	71.4 ₃
	BLSMOTE+XGBoost	81.6 ₃	74.8 ₃	74.9 ₃	76.1 ₃	83.5 ₃	78.2 ₃
	SMOTE+SVM	78.1 ₅	71.8 ₃	72.5 ₃	70.0 ₂	76.2 ₃	73.7 ₃
	SMOTE+XGBoost	81.9 ₄	75.3 ₅	77.1 ₂	77.3 ₃	84.9 ₂	79.3 ₃
	SMOTETomek+SVC	78.3 ₅	71.4 ₃	72.9 ₃	69.8 ₂	75.5 ₃	73.4 ₃
	SMOTETomek+XGBoost	81.5 ₆	75.4 ₅	78.0 ₂	77.4 ₃	85.2 ₂	79.5 ₄
	SMOTEENN+SVC	78.2 ₄	70.5 ₃	73.2 ₂	70.9 ₂	75.8 ₂	73.7 ₃
	SMOTEENN+XGBoost	83.6 ₄	80.1 ₅	80.5 ₂	80.2 ₃	86.9₃	82.2 ₃
	ClusterC+SVM	65.1 ₃	60.1 ₅	63.7 ₄	60.2 ₃	63.4 ₄	62.5 ₄
	ClusterC+XGBoost	67.8 ₅	56.4 ₇	54.9 ₁₁	65.9 ₆	78.3 ₄	64.7 ₇
	NearMiss+SVM	49.8 ₃	48.7 ₃	59.6 ₃	62.5 ₄	69.7 ₂	58.0 ₃
	NearMiss+XGBoost	74.5 ₄	60.7 ₆	62.4 ₅	57.0 ₃	66.9 ₁	64.3 ₄
	TomekL+SVM	59.3 ₈	44.5 ₅	45.5 ₃	42.1 ₆	51.4 ₄	48.5 ₅
Ensemble	TomekL+XGBoost	77.3 ₆	70.0 ₃	68.1 ₃	71.8 ₄	78.5 ₃	73.2 ₄
	AdaBoost	68.5 ₇	50.0 ₃	53.9 ₄	57.7 ₆	73.4 ₅	60.7 ₅
Outlier	RF	69.8 ₆	59.6 ₅	54.9 ₃	54.0 ₄	72.5 ₆	62.1 ₅
	CBLOF	45.7 ₁	41.2 ₁	44.3 ₃	52.4 ₁	64.4 ₁	49.5 ₁
	COPOD	46.4 ₁	44.8 ₁	46.8 ₁	54.6 ₁	63.9 ₁	51.3 ₁
	KNN	45.4 ₁	45.2 ₁	45.5 ₁	54.1 ₁	67.1 ₂	51.4 ₁
	LMDD	13.8 ₄	18.1 ₇	23.9 ₁₀	34.6 ₁₂	23.6 ₈	22.8 ₈
	LOF	50.8 ₂	46.2 ₂	47.3 ₁	49.7 ₁	61.0 ₂	50.9 ₂
	LSAD	53.5 ₂	48.0 ₄	53.7 ₄	57.8 ₂	68.5 ₂	56.2 ₃
	MCD	65.2 ₁₁	47.0 ₄	44.5 ₁	51.7 ₃	65.0 ₁	54.6 ₄
	OCSVM	47.8 ₁	49.7 ₁	46.7 ₁	41.7 ₁	32.5 ₁	43.6 ₁
	PCA	45.7 ₁	41.1 ₁	43.7 ₁	51.6 ₁	63.9 ₁	49.2 ₁
	WSVM	71.6 ₆	59.8 ₅	56.3 ₅	55.7 ₄	76.2 ₄	63.9 ₅
	Hybrid (Ensemble, Sampling)	BBagg	75.9 ₆	75.5 ₃	76.5 ₂	76.1 ₂	80.9 ₃
		BCascade	72.1 ₉	64.0 ₄	68.8 ₄	68.5 ₃	80.1 ₂
		BRF	83.1 ₂	81.0₄	80.8 ₂	80.4 ₂	86.4 ₂
		EasyE	83.8₂	79.2 ₃	80.9₁	81.6₁	84.9 ₂
Hybrid (Ensemble, Outlier)		RUSBoost	80.4 ₃	77.9 ₄	79.3 ₂	78.8 ₂	84.1 ₃
		SelfPE	78.2 ₃	67.3 ₅	68.2 ₂	69.5 ₅	80.2 ₃
		UBagg	75.9 ₆	75.5 ₃	76.5 ₂	76.1 ₂	80.9 ₃
		OverBoost	81.6 ₄	77.6 ₃	78.3 ₂	80.6 ₁	84.6 ₃
		SMOTEBoost	78.7 ₄	67.8 ₄	73.8 ₃	71.7 ₃	83.0 ₃
		SMOTEBagg	38.1 ₅	35.7 ₅	36.9 ₃	40.6 ₄	64.1 ₄
		AdaCost	79.7 ₄	72.7 ₄	76.2 ₁	74.1 ₄	81.4 ₃
		AdaUBoost	83.3 ₄	77.4 ₄	78.4 ₃	80.3 ₂	84.3 ₃
		WRF	79.2 ₅	71.4 ₂	74.1 ₂	70.2 ₃	75.6 ₃
		FBagg	52.1 ₁	46.9 ₁	48.5 ₁	48.7 ₁	63.8 ₂
		IF	48.9 ₁	46.3 ₂	51.1 ₂	55.5 ₁	67.4 ₁
		LODA	35.5 ₃	29.1 ₃	30.6 ₉	34.1 ₅	48.4 ₉
		XGBOD	0.0 ₀	0.0 ₀	0.0 ₀	0.0 ₀	0.0 ₀

Table 6 Overview of the GM scores (%) achieved on the Taiwan and Bosnia and Herzegovina bankruptcy datasets in the form of GM_{std}

Category	Method	Data	
		TW	B&H
Sampling	ADASYN+SVM	84.4 ₃	92.9 ₅
	ADASYN+XGBoost	85.5 ₃	94.0 ₅
	BLSMOTE+SVM	85.6 ₃	91.9 ₆
	BLSMOTE+XGBoost	83.9 ₄	92.4 ₆
	SMOTE+SVM	86.5 ₂	91.9 ₈
	SMOTE+XGBoost	85.4 ₃	92.5 ₈
	SMOTETomek+SVC	86.4 ₂	92.9 ₆
	SMOTETomek+XGBoost	85.8 ₃	95.0 ₆
	SMOTEENN+SVC	85.9 ₂	91.4 ₆
	SMOTEENN+XGBoost	86.9 ₃	94.5 ₅
	ClusterC+SVM	82.3 ₂	91.5 ₅
	ClusterC+XGBoost	84.3 ₂	95.0 ₅
	NearMiss+SVM	78.7 ₄	88.1 ₉
	NearMiss+XGBoost	84.1 ₄	91.9 ₇
	TomekL+SVM	49.4 ₆	90.3 ₈
	TomekL+XGBoost	58.6 ₅	91.7 ₇
Ensemble	AdaBoost	54.6 ₅	95.5 ₆
	RF	44.5 ₄	94.5 ₇
Outlier	CBLOF	63.4 ₁	45.3 ₈
	COPOD	72.3 ₁	31.9 ₅
	KNN	57.3 ₂	40.5 ₅
	LMDD	23.5 ₁₃	37.9 ₅
	LOF	52.8 ₂	42.8 ₅
	LSAD	53.8 ₂	46.7 ₇
	MCD	62.3 ₂	50.5 ₂
	OCSVM	73.7 ₂	61.5 ₁₁
	PCA	61.5 ₁	40.1 ₆
	WSVM	48.8 ₄	96.5₅
Hybrid (Ensemble, Sampling)	BBagg	86.4 ₃	89.3 ₁₁
	BCascade	85.7 ₂	91.5 ₈
	BRF	87.2₂	94.0 ₇
	EasyE	86.0 ₃	94.0 ₇
	RUSBoost	86.5 ₃	93.0 ₇
	SelfPE	85.9 ₄	91.5 ₈
	UBagg	86.4 ₃	94.7 ₇
	OverBoost	85.8 ₄	94.0 ₇
	SMOTEBoost	84.8 ₃	93.0 ₆
	SMOTEBagg	61.8 ₅	94.0 ₇
Hybrid (Ensemble, Cost-sensitive)	AdaCost	83.6 ₄	93.0 ₅
	AdaUBoost	85.6 ₄	94.0 ₇
	WRF	86.3 ₃	91.4 ₇
Hybrid (Ensemble, Outlier)	FBagg	54.4 ₁	42.2 ₃
	IF	71.7 ₂	45.2 ₄
	LODA	52.8 ₁₁	40.7 ₇
	XGBOD	64.0 ₃	94.4 ₄

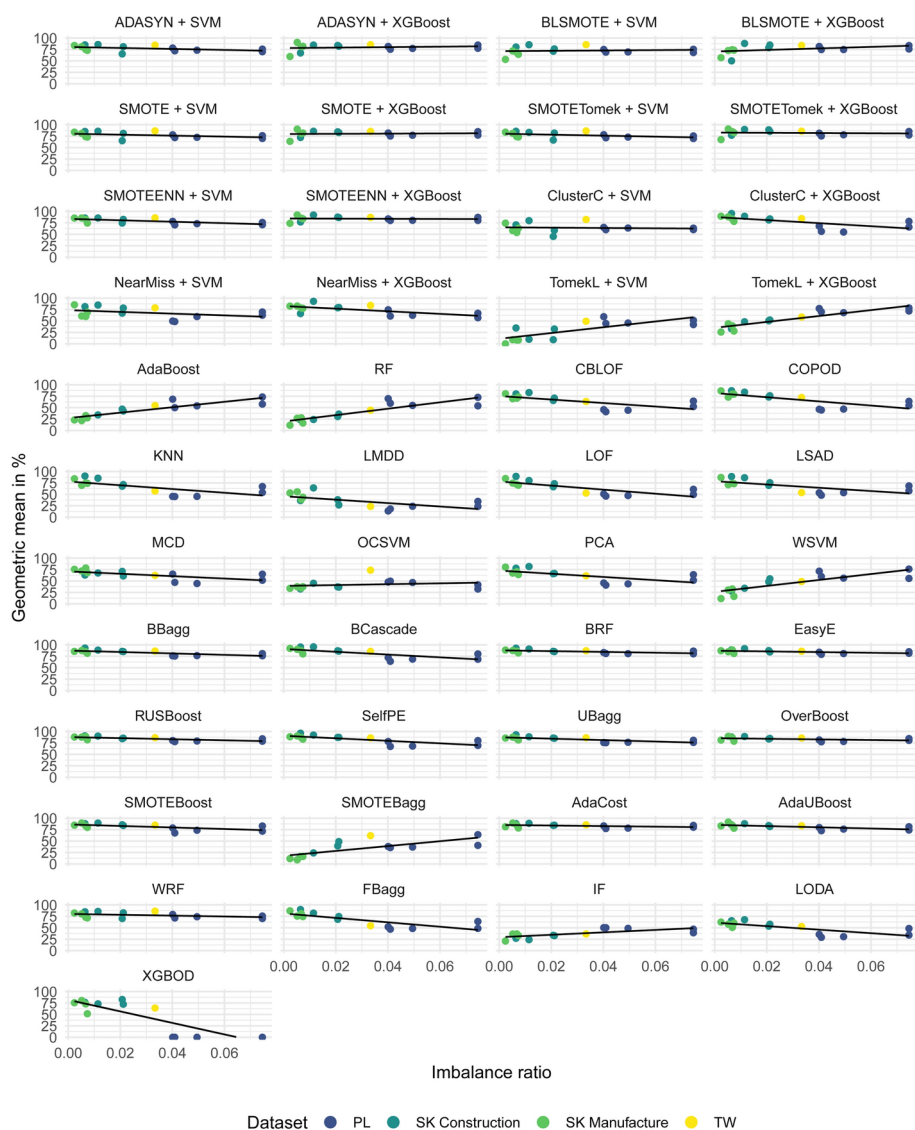


Fig. 3 GM score as a function of the dataset imbalance ratio for different methods

more, outlier detection-based approaches yielded CDiff scores up to 41.6, which indicates their specific application.

These numerical insights highlight that while hybrid classifiers often provide robust solutions across different datasets, the effectiveness of any given approach can vary significantly. The substantial range in CDiff scores across all classifiers reinforces the message that machine learning model selection is highly context-dependent. The practical application of these models requires a deep understanding of both the algorithms and the specific characteristics of the data to which they are applied.

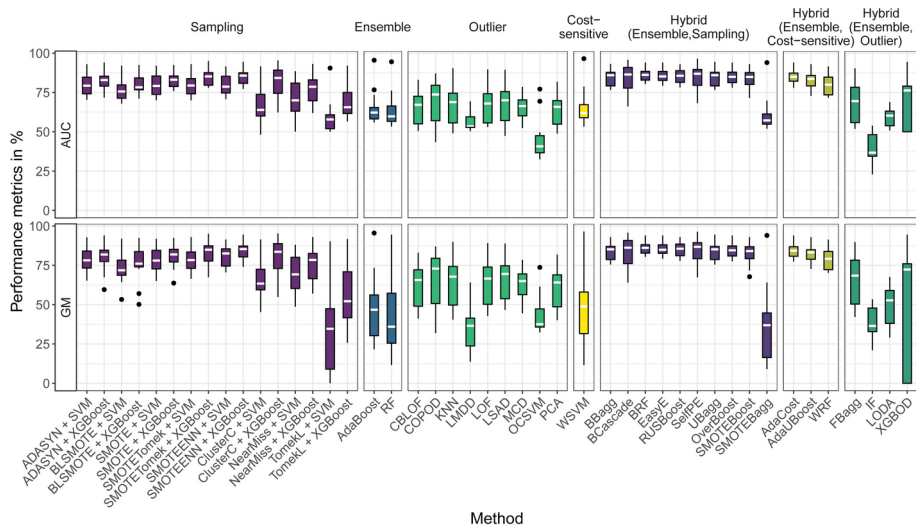


Fig. 4 Averaged AUC and GM scores for all utilized datasets per method

Table 7 Overview of the CDiff scores for all utilized machine learning approaches

Category	Method	CDiff	Category	Method	CDiff
Sampling	ADASYN+SVM	20.5	Sampling	ADASYN+XGBoost	14.5
	BLSMOTE+SVM	25.1		BLSMOTE+XGBoost	19.3
	SMOTE+SVM	20.3		SMOTE+XGBoost	14.1
	SMOTETomek+SVC	20.2		SMOTETomek+XGBoost	9.3
	SMOTEENN+SVC	19.5		SMOTEENN+XGBoost	6.5
	ClusterC+SVM	31.6		ClusterC+XGBoost	15.9
	NearMiss+SVM	28.1		NearMiss+XGBoost	22.2
	TomekL+SVM	41.0		TomekL+XGBoost	30.9
Ensemble	AdaBoost	34.0	Cost-sensitive	WSVM	20.1
	RF	35.9			
Hybrid (Ensemble, Sampling)	BBagg	11.0	Outlier	CBLOF	32.9
	BCascade	12.0		COPOD	27.4
	BRF	5.26		KNN	29.5
	EasyE	6.5		LMDD	41.6
	RUSBoost	7.0		LOF	30.9
	SelfPE	10.6		LSAD	26.5
	UBagg	10.0		MCD	31.8
	OverBoost	10.3		OCSVM	38.1
	SMOTEBoost	13.6		PCA	35.0
	SMOTEBagg	39.4			
Hybrid (Ensemble, Outlier)	FBagg	27.8	Hybrid (Ensemble, Cost-sensitive)	AdaCost	14.7
	IF	27.5		AdaUBoost	10.6
	LODA	39.0		WRF	33.3
	XGBOD	32.7			

A lower CDiff score indicates greater robustness and effectiveness across various datasets

Moreover, the observed variability stresses the importance of rigorous validation and testing in diverse scenarios to ensure the selected model's generalizability and effectiveness in real-world applications. This approach not only aids in achieving optimal performance but also in understanding the limitations and strengths of different classifiers in various contexts.

7 Discussion

The results indicate that a group of methods based on undersampling and different ensemble strategies are able to learn patterns in data with various imbalance ratios. Several limitations must be considered when generalizing the results. We attempted to collect all publicly available datasets used in recent years for bankruptcy classification. These datasets are heterogeneous and cover different types of economies and various geographical locations. Using different datasets may lead to a shift in the generalized results. However, the best-performing methods yielded very satisfying results across all evaluated datasets.

Real-world financial data may contain inaccuracies, and their application may involve practical challenges. The issue of financial reporting quality is of interest to many studies from various points of view Baik et al. (2022), Vander Bauwhede et al. (2015), and Park (2018). Accounting data, sources for financial ratios used in our analysis, may not accurately represent a company. In general, there are two kinds of errors. The first type of error is unintentional clerical errors, which may or may not be immediately obvious and significant. The second type of error occurs from the intentional tampering of accounting data, where shareholders attempt to misrepresent companies' situations based on their personal goals. However, these types of errors should not significantly influence our research.

As mentioned, the selected datasets were from different sources and contained different features. Some of the features overlap across the datasets, but some are unique. Therefore, for all results, as datasets are composed of different features that strongly influence the classification results, the dataset imbalance ratio should not be considered the only main factor impacting the results.

We did not use any feature selection techniques from the methodology perspective since there were no high-dimensional dataset problems. Modern classifiers can handle high-dimensional datasets such as those used in this paper. Applying a feature selection method would increase the chances of overfitting.

We have provided a detailed list of the tuned hyperparameter values in Appendix B. In most cases, we selected two hyperparameters for each method and adjusted them within a reasonable range. We conjecture that more extensive tuning could potentially enhance the prediction performance of specific methods. However, it is important to consider the risk of overfitting that comes with excessive tuning. Generally, there is a preference for methods that exhibit robustness to variations in hyperparameter settings, as these methods tend to achieve higher scores in our experiments. Conversely, algorithms overly sensitive to hyperparameter adjustments often succumb to overfitting and are rarely employed in practical applications.

The comparison presented in our paper focuses primarily on performance. However, it is crucial to consider other factors when selecting the most suitable approach. A key aspect is the interpretability of classification models. It is essential to acknowledge that although

some models provide high predictive accuracy, their black box nature can significantly affect their trustworthiness and practical applicability, especially in sensitive fields like finance. If interpretability is a major concern, one should favor tree-based models, which offer high interpretability when combined with Shapley values. Notably, the best-performing methods in our study are tree-based ensembles, which underscores their high applicability for bankruptcy prediction tasks due to their enhanced interpretability.

8 Conclusion

This paper has provided an in-depth experimental comparison of class imbalance learning methods applied to the task of bankruptcy prediction. Our investigation covered a wide array of 45 distinct methods, evaluated across 15 publicly accessible datasets spanning four diverse countries: the Slovak Republic, Bosnia and Herzegovina, Taiwan, and Poland. The results of these extensive experiments have led us to several key insights and contributions to the field of imbalanced learning for financial distress prediction.

First, among the array of methods tested, the BRF, SMOTEENN+XGBoost, EasyE, and RUSBoost classifiers emerged as the top performers. This superior performance underscores the effectiveness of integrating ensemble learning with undersampling strategies and combined sampling integrated with the XGBoost classifier, showcasing their robustness across a variety of imbalanced datasets.

Secondly, these leading methods consistently achieved high GM and AUC scores across all datasets, demonstrating their adaptability and efficacy regardless of the varying degrees of class imbalance present in the data. This highlights the critical role of methodological flexibility in effectively addressing diverse imbalance ratios.

Thirdly, our findings reveal a distinct advantage of hybrid approaches that blend ensemble learning with sampling techniques over their pure sampling, cost-sensitive learning, and standalone ensemble counterparts. This synergy between ensemble strategies and sampling techniques contributes to a more nuanced handling of imbalanced data, leading to improved predictive performance.

Fourthly, despite the intuitive appeal of outlier detection methods for highly imbalanced scenarios, our analysis indicated that these methods fell short compared to hybrid ensemble-based approaches. Even in situations with very high levels of imbalance, much like those typically handled by outlier detection, the hybrid methods always did better. This indicates that outlier detection might not fully address the complex challenges of predicting bankruptcy.

In addition to these findings, our research has revealed the multifaceted nature of bankruptcy prediction across different economic contexts. The varying performance across datasets from different countries highlights the importance of considering local economic factors and dataset-specific characteristics in developing and applying predictive models.

Furthermore, our study contributes to the methodological discourse on imbalanced learning by demonstrating the utility of combining multiple strategies to tackle class imbalance, a common challenge in many predictive modeling tasks beyond bankruptcy prediction. The insights gained from this comprehensive comparison inform best practices for predicting financial distress and offer valuable guidance for addressing class imbalance in other domains.

In conclusion, this paper identifies effective strategies for bankruptcy prediction in the presence of class imbalance and sets a foundation for future research to explore the nuanced interactions between methodological choices and dataset characteristics. Future work could extend our findings by investigating the impact of feature engineering, deep learning approaches, and the integration of macroeconomic factors, providing a more holistic view of bankruptcy prediction models' performance across varying economic landscapes.

Future research stemming from this paper could explore several key directions to enhance the field of bankruptcy prediction. One promising area is the integration of macroeconomic variables and external factors, such as market trends, industry performance, or geopolitical events, into predictive models. This could improve the accuracy of models, especially in dynamic economic environments where individual financial ratios may not fully capture the risk of bankruptcy. However, the availability of such a diverse dataset is very limited, even though building one could be highly beneficial.

Another avenue for future research is the application of deep learning techniques, such as transformer-based models, which may capture complex temporal patterns in financial data that traditional machine learning models may miss. Neural networks have historically been less favored for imbalanced datasets because they require large amounts of data to perform well, and financial distress datasets are often small and imbalanced. However, recent advancements in techniques such as transfer learning and data augmentation are opening new possibilities.

In our study, hybrid methods consistently achieved high GM scores, even in datasets with extreme imbalance ratios. This suggests that these models can handle significant class imbalances without losing predictive power. Hybrid models combine the strengths of different methodologies, and by utilizing neural networks for deep feature extraction, they could potentially further boost performance. For example, integrating ensemble techniques like XGBoost or RF with a neural network could address class imbalance and improve the prediction of minority class cases, such as bankruptcies.

Appendix A: Comparison of economic attributes in analyzed datasets

All attributes by datasets are listed in Table 8. There are five common categories of financial ratios: (a) liquidity ratios, (b) leverage ratios, (c) efficiency ratios, (d) profitability ratios, and (e) market value ratios.

Table 8 Attributes comparison across datasets

	Bosnia and Herzegovina dataset	Polish bankruptcy dataset	Slovak bankruptcy dataset	Taiwan bankruptcy dataset
Class	Y bankruptcy binary status	Y bankruptcy binary status (last col)	Y bankruptcy binary status	Y bankruptcy binary status
Attribute 1	Net working capital	X1 net profit / total assets	X1 Return On Assets (ROA)	X1 Cost of Interest-bearing Debt
Attribute 2	Adjusted net working capital	X2 total liabilities / total assets	X2 Return On Equity (ROE)	X2 Cash Reinvestment Ratio
Attribute 3	Adjusted net working capital/Sales	X3 working capital / total assets	X3 Return On Sales (ROS)	X3 Current Ratio
Attribute 4	Cash flow from operations	X4 current assets / short-term liabilities	X4 Cash Ratio (L1)	X4 Acid Test
Attribute 5	Current ratio	X5 [(cash + short-term securities + receivables - short-term liabilities) / (operating expenses - depreciation)] * 365	X5 Quick Ratio (L2)	X5 Interest Expenses/Total Revenue
Attribute 6	Acid test	X6 retained earnings / total assets	X6 Current Ratio (L3)	X6 Total Liability/Equity Ratio
Attribute 7	Cash liquidity	X7 EBIT / total assets	X7 Asset Turnover Days (ATD)	X7 Liability/Total Assets
Attribute 8	Days accounts receivables	X8 book value of equity / total liabilities	X8 Days Total Receivables Outstanding (DTR)	X8 Interest-bearing Debt/Equity
Attribute 9	Days inventory	X9 sales / total assets	X9 Total Asset Turnover (TAT)	X9 Contingent Liability/Equity
Attribute 10	Days accounts payables	X10 equity / total assets	X10 Inventory Turnover Days (ITD)	X10 Operating Income/Capital
Attribute 11	Fixed assets/Sales	X11 (gross profit + extraordinary items + financial expenses) / total assets	X11 Debt-to-Assets Ratio (DA)	X11 Pretax Income/Capital
Attribute 12	Equity turnover	X12 gross profit / short-term liabilities	X12 Debt-to-Equity Ratio (DE)	X12 Working Capital to Total Assets
Attribute 13	Assets/Equity	X13 (gross profit + depreciation) / sales	X13 Financial Leverage (FL)	X13 Quick Assets/Total assets
Attribute 14	Gross profit margin	X14 (gross profit + interest) / total assets	X14 Return On Investment (ROI)	X14 Current Assets/Total Assets
Attribute 15	Net profit margin	X15 (total liabilities * 365) / (gross profit + depreciation)	X15 Debt To Income Ratio (DIR)	X15 Cash/Total Assets
Attribute 16	Employee costs/Sales	X16 (gross profit + depreciation) / total liabilities	X16 Debt Service Coverage Ratio (DCR)	X16 Quick Assets/Current Liability

Table 8 (continued)

	Bosnia and Herzegovina dataset	Polish bankruptcy dataset	Slovak bankruptcy dataset	Taiwan bankruptcy dataset
Attribute 17	EBITDA/Sales	X17 total assets / total liabilities	X17 Asset Coverage Ratio (ACR)	X17 Cash/Current Liability
Attribute 18	EBIT/Sales	X18 gross profit / total assets	X18 Labor Productivity (LP)	X18 Current Liability to Assets
Attribute 19	Net result from operations after tax	X19 gross profit / sales	X19 Bank Liabilities to Debt Ratio (BL)	X19 Operating Funds to Liability
Attribute 20	Return on assets	X20 (inventory * 365) / sales	X20 Labor-to-Revenue Ratio (LRR)	X20 Inventory/Working Capital
Attribute 21	Return on equity	X21 sales (n) / sales (n-1)	X21 Wages to Added Value Ratio (WAR)	X21 Inventory/Current Liability
Attribute 22	Retained earnings/Shareholders equity	X22 profit on operating activities / total assets		X22 Current Liabilities/Liability
Attribute 23	Retained earnings/Equity	X23 net profit / sales		X23 Working Capital/Equity
Attribute 24	Gearing ratio	X24 gross profit (in 3 years) / total assets		X24 Current Liabilities/Equity
Attribute 25	Financial leverage	X25 (equity - share capital) / total assets		X25 Long-term Liability to Current Assets
Attribute 26	Operational leverage	X26 (net profit + depreciation) / total liabilities		X26 Current Liability to Current Assets
Attribute 27	Debt to equity	X27 profit on operating activities / financial expenses		X27 One if Total Liability exceeds Total Assets
Attribute 28	Interest coverage	X28 working capital / fixed assets		X28 Equity to Liability
Attribute 29	Debt to assets	X29 logarithm of total assets		X29 Equity/Total Assets
Attribute 30	Loans to debt	X30 (total liabilities - cash) / sales		X30 (Long-term Liability+Equity)/ Fixed Assets
Attribute 31	Years to service debt	X31 (gross profit + interest) / sales		X31 Fixed Assets to Assets
Attribute 32	Dummy loss in excess of equity	X32 (current liabilities * 365) / cost of products sold		X32 Current Liability to Liability
Attribute 33	Loss in excess of equity	X33 operating expenses / short-term liabilities		X33 Current Liability to Equity
Attribute 34	Debt capacity	X34 operating expenses / total liabilities		X34 Equity to Long-term Liability
Attribute 35	Debt	X35 profit on sales / total assets		X35 Liability to Equity

Table 8 (continued)

	Bosnia and Herzegovina dataset	Polish bankruptcy dataset	Slovak bankruptcy dataset	Taiwan bankruptcy dataset
Attribute 36	Dummy EBIDA - CAPEX	X36 total sales / total assets		X36 Degree of Financial Leverage
Attribute 37	EBIDA - CAPEX	X37 (current assets - inventories) / long-term liabilities		X37 Interest Coverage Ratio
Attribute 38	(EBIDA - CAPEX)/Assets	X38 constant capital / total assets		X38 Operating Expenses/Net Sales
Attribute 39		X39 profit on sales / sales		X39 (Research and Development Expenses)/Net Sales
Attribute 40		X40 (current assets - inventory - receivables) / short-term liabilities		X40 Effective Tax Rate
Attribute 41		X41 total liabilities / ((profit on operating activities + depreciation) * (12/365))		X41 Book Value Per Share(B)
Attribute 42		X42 profit on operating activities / sales		X42 Book Value Per Share(A)
Attribute 43		X43 rotation receivables + inventory turnover in days		X43 Book Value Per Share(C)
Attribute 44		X44 (receivables * 365) / sales		X44 Cash Flow Per Share
Attribute 45		X45 net profit / inventory		X45 Sales Per Share
Attribute 46		X46 (current assets - inventory) / short-term liabilities		X46 Operating Income Per Share
Attribute 47		X47 (inventory * 365) / cost of products sold		X47 Sales Per Employee
Attribute 48		X48 EBITDA (profit on operating activities - depreciation) / total assets		X48 Operation Income Per Employee
Attribute 49		X49 EBITDA (profit on operating activities - depreciation) / sales		X49 Fixed Assets Per Employee
Attribute 50		X50 current assets / total liabilities		X50 total assets to GNP price
Attribute 51		X51 short-term liabilities / total assets		X51 Return On Total Assets(C)
Attribute 52		X52 (short-term liabilities * 365) / cost of products sold		X52 Return On Total Assets(A)
Attribute 53		X53 equity / fixed assets		X53 Return On Total Assets(B)
Attribute 54		X54 constant capital / fixed assets		X54 Gross Profit /Net Sales
Attribute 55		X55 working capital		X55 Realized Gross Profit/Net Sales
Attribute 56		X56 (sales - cost of products sold) / sales		X56 Operating Income /Net Sales

Table 8 (continued)

	Bosnia and Herzegovina dataset	Polish bankruptcy dataset	Slovak bankruptcy dataset	Taiwan bankruptcy dataset
Attribute 57		X57 (current assets - inventory - short-term liabilities) / (sales - gross profit - depreciation)		X57 Pre-Tax Income/Net Sales
Attribute 58		X58 total costs / total sales		X58 Net Income/Net Sales
Attribute 59		X59 long-term liabilities / equity		X59 Net Non-operating Income Ratio
Attribute 60		X60 sales / inventory		X60 Net Income-Exclude Disposal Gain or Loss/Net Sales
Attribute 61		X61 sales / receivables		X61 EPS-Net Income
Attribute 62		X62 (short-term liabilities *365) / sales		X62 Pretax Income Per Share
Attribute 63		X63 sales / short-term liabilities		X63 Retained Earnings to Total Assets
Attribute 64		X64 sales / fixed assets		X64 Total Income to Total Expenses
Attribute 65				X65 Total Expenses to Assets
Attribute 66				X66 Net Income to Total Assets
Attribute 67				X67 Gross Profit to Sales
Attribute 68				X68 Net Income to Stockholder's Equity
Attribute 69				X69 One if Net Income is Negative for the Last Two Years; Zero Otherwise
Attribute 70				X70 (Inventory + Accounts Receivables) / Equity
Attribute 71				X71 Total Asset Turnover
Attribute 72				X72 Accounts Receivable Turnover
Attribute 73				X73 Days Receivable Outstanding
Attribute 74				X74 Inventory Turnover
Attribute 75				X75 Fixed Asset Turnover
Attribute 76				X76 Equity Turnover
Attribute 77				X77 Current Assets to Sales
Attribute 78				X78 Quick Assets to Sales

Table 8 (continued)

Bosnia and Herzegovina dataset	Polish bankruptcy dataset	Slovak bankruptcy dataset	Taiwan bankruptcy dataset
Attribute 79			X79 Working Capital to Sales
Attribute 80			X80 Cash to Sales
Attribute 81			X81 Cash Flow to Sales
Attribute 82			X82 No-credit Interval
Attribute 83			X83 Cash Flow from Operating/Curent Liabilities
Attribute 84			X84 Cash Flow to Total Assets
Attribute 85			X85 Cash Flow to Liability
Attribute 86			X86 CFO to Assets
Attribute 87			X87 Cash Flow to Equity
Attribute 88			X88 Realized Gross Profit Growth Rate
Attribute 89			X89 Operating Income Growth
Attribute 90			X90 Net Income Growth
Attribute 91			X91 Continuing Operating Income after Tax Growth
Attribute 92			X92 Net Income-Excluding Disposal Gain or Loss Growth
Attribute 93			X93 Total Asset Growth
Attribute 94			X94 Total Equity Growth
Attribute 95			X95 Return on Total Asset Growth

Appendix B: Hyperparameters

Table 9 summarizes the grid of utilized hyperparameters we searched.

Table 9 Hyperparameter space of classifiers searched during the numerical experiments

Category	Method	Hyperparameter	Grid of tested values
Sampling	ADASYN	Number of nearest neighbors	[2, 3, 4, 5]
	BLSMOTE	Number of nearest neighbors	[2, 3, 4, 5]
	SMOTE	Number of nearest neighbors	[2, 3, 4, 5]
	SMOTETomek	Number of nearest neighbors	[2, 3, 4, 5]
		Sampling_strategy	['all']
	SMOTEENN	Number of nearest neighbors	[2, 3, 4, 5]
		Sampling_strategy	['all']
	ClusterC	Sampling_strategy	['majority']
	NearMiss	Number of nearest neighbors	[2, 3, 4, 5]
		Sampling_strategy	['majority']
Ensemble	TomekL	Sampling_strategy	['majority']
	AdaBoost	Number of estimators	[50 : 50 : 500]
	RF	Number of estimators	[50 : 50 : 500]
		Maximum depth	[50 : 50 : 500]
		Number of features	['auto', 'log2', 5, 10, 15, 20]
		Criterion	['gini', 'entropy']
	CBLOF	Number of clusters	[2, 3, 5, 8, 10, 15]
Outlier		Alpha	[0.6, 0.7, 0.8, 0.9]
		Beta	[2, 5, 10, 15, 20, 50, 100]
		Weights	[True, False]
	KNN	Number of nearest neighbors	[3, 5, 7, 10]
		Method	['largest', 'mean', 'median']
	LMDD	Number of iterations	[50 : 50 : 500]
		Dissimilarity measure	['aad', 'var', 'idr']
	LOF	Number of neighbors	[3, 5, 7, 10, 15, 20]
		Algorithm	['ball_tree', 'kd_tree', 'brute', 'auto']
		Metric	["euclidean", "minkowski"]
	LSAD	Rho	[0.001, 0.01, 0.1, 0.5, 1, 3, 5, 10]

Table 9 (continued)

Category	Method	Hyperparameter	Grid of tested values
Cost-sensitive	MCD OCSVM	Sigma	[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 2, 5, 10]
		Assume centered	[True, False]
	PCA WSVM	Degree	[1, 2, 3]
		Nu	[0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]
		Gamma	[0.01, 0.1, 1, 3, 5]
		SVD solver	['auto', 'full', 'arpack', 'randomized']
		Gamma	[0.001, 0.01, 0.1, 1, 10, 100]
		C	[0.01, 0.1, 1, 5, 10, 100]
		Cass weights	[{0 : 100, 1 : 1}, {0 : 75, 1 : 1}, {0 : 50, 1 : 1}, {0 : 25, 1 : 1}, {0 : 10, 1 : 1}, {0 : 1, 1 : 1}, {0 : 1, 1 : 10}, {0 : 1, 1 : 25}, {0 : 1, 1 : 50}, {0 : 1, 1 : 75}, {0 : 1, 1 : 100}, 'balanced']
			[50 : 50 : 500]
Hybrid (Ensemble, Sampling)	BBagg	Number of estimators	['notminority', 'majority', 'auto']
		Sampling strategy	[5 : 5 : 20]
		Number of features	[50 : 50 : 500]
		Number of estimators	[DT]
	BCascade	Estimator	['gini', 'entropy']
		Estimator_criterion	['best', 'random']
		Estimator_split strategy	[0.1, 0.3, 0.5, 0.7, 0.9]
		Estimator_complexity	[50 : 50 : 500]
	BRF	Number of estimators	['gini', 'entropy']
		Criterion	['auto', 'log2', 5, 10, 15, 20]
EasyE RUSBoost		Number of features	['notminority', 'majority', 'auto']
		Sampling strategy	[50 : 50 : 500]
		Number of estimators	[50 : 50 : 500]
		Number of estimators	[50 : 50 : 500]

Table 9 (continued)

Category	Method	Hyperparameter	Grid of tested values
SelfPE		Learning rate	[0.001, 0.01, 0.1, 0.5, 1, 3, 5, 10, 15, 20]
		Sampling strategy	['notminority', 'majority', 'auto']
		Number of estimators	[50 : 50 : 500]
		Estimator	[DT]
		Estimator_criterion	['gini', 'entropy']
		Estimator_split strategy	['best', 'random']
UBagg		Estimator_complexity	[0.1, 0.3, 0.5, 0.7, 0.9]
		Number of estimators	[50 : 50 : 500]
		Number of features	['auto', 'log2', 5, 10, 15, 20]
		Warm start	[True, False]
		Number of estimators	[50 : 50 : 500]
OverBoost		Learning rate	[0.001, 0.01, 0.1, 1]
		Algorithm	['SAMME', 'SAMME.R']
		Number of estimators	[50 : 50 : 500]
SMOTEBoost		Learning rate	[0.001, 0.01, 0.1, 1]
		Number of nearest neighbors	[3, 4, 5]
		Algorithm	['SAMME', 'SAMME.R']
		Number of estimators	[50 : 50 : 500]
		Number of features	[5 : 5 : 20]
SMOTEBagg		Number of nearest neighbors	[3, 4, 5]
		Warm start	[True, False]
		Number of estimators	[50 : 50 : 500]
AdaCost		Learning rate	[0.001, 0.01, 0.1, 1]
		Algorithm	['SAMME', 'SAMME.R']
		Number of estimators	[50 : 50 : 500]
		Learning rate	[0.001, 0.01, 0.1, 1]
AdaUBoost		Algorithm	['SAMME', 'SAMME.R']
		Number of estimators	[50 : 50 : 500]
		Learning rate	[0.001, 0.01, 0.1, 1]
		Algorithm	['SAMME', 'SAMME.R']

Table 9 (continued)

Category	Method	Hyperparameter	Grid of tested values
Hybrid (Ensemble, Outlier)	WRF	Number of estimators	[50 : 50 : 500]
		Maximum depth	[50 : 50 : 500]
		Number of features	[<i>'auto'</i> , <i>'log2'</i> , 5, 10, 15, 20]
		Criterion	[<i>'gini'</i> , <i>'entropy'</i>]
		Class weights	{0 : 100, 1 : 1}, {0 : 75, 1 : 1}, {0 : 50, 1 : 1}, {0 : 25, 1 : 1}, {0 : 10, 1 : 1}, {0 : 1, 1 : 1}, {0 : 1, 1 : 10}, {0 : 1, 1 : 25}, {0 : 1, 1 : 50}, {0 : 1, 1 : 75}, {0 : 1, 1 : 100} , <i>'balanced'</i>]
	FBagg	Number of estimators	[50 : 50 : 500]
		Number of features	[0.5, 0.7, 1.0]
	IF	Combination	[<i>'average'</i> , <i>'max'</i>]
		Number of estimators	[50 : 50 : 500]
	LODA	Maximum samples	[256, 512, 1024, 2048 , <i>'auto'</i>]
Number of random cuts		[50 : 50 : 500]	
XGBOD	—	—	—

Appendix C: AUC score results

In addition to the GM score that was reported in the main paper, we also calculated the AUC score for all experiments. In Tables [10](#), [11](#), [12](#), [13](#), we provide the AUC values for all datasets.

Table 10 Overview of the AUC scores (%) achieved on the Slovak manufacture bankruptcy datasets in the form of AUC_{std}

Category	Method	Data				Avg.
		SK_M_13	SK_M_14	SK_M_15	SK_M_16	
Sampling	ADASYN+SVM	73.8 ₁₁	75.3 ₁₀	81.6 ₈	84.9 ₁₀	78.9 ₁₀
	ADASYN+XGBoost	82.5 ₄	82.1 ₆	90.9 ₇	71.8 ₁₄	82.7 ₈
	BLSMOTE+SVM	71.8 ₁₃	72.3 ₁₂	75.6 ₁₂	67.9 ₁₁	71.9 ₁₂
	BLSMOTE+XGBoost	76.7 ₇	78.4 ₁₁	76.8 ₉	71.4 ₁₇	75.8 ₁₁
	SMOTE+SVM	73.8 ₁₁	74.6 ₉	81.5 ₈	84.9 ₁₀	78.7 ₁₀
	SMOTE+XGBoost	82.6 ₄	81.8 ₆	90.9 ₇	75.9 ₁₇	85.2 ₉
	SMOTETomek+SVC	74.1 ₁₁	74.7 ₉	81.3 ₉	83.7 ₁₁	78.8 ₁₀
	SMOTETomek+XGBoost	83.1 ₅	85.2 ₃	91.2 ₇	79.2 ₂₀	84.6 ₉
	SMOTEENN+SVC	74.4 ₉	75.0 ₁₀	81.2 ₈	84.8 ₁₀	78.8 ₉
	SMOTEENN+XGBoost	84.4 ₄	85.7 ₇	92.4 ₅	77.3 ₅	84.9 ₆
	ClusterC+SVM	63.5 ₈	55.8 ₉	59.3 ₅	76.2 ₁₃	63.7 ₉
	ClusterC+XGBoost	78.9 ₆	84.3 ₄	88.4 ₅	89.9 ₉	85.4 ₆
	NearMiss+SVM	70.7 ₁₀	63.7 ₁₃	63.7 ₂₀	85.9 ₈	71.0 ₁₃
	NearMiss+XGBoost	78.0 ₇	78.7 ₄	83.0 ₆	83.1 ₁₂	80.6 ₈
	TomekL+SVM	51.7 ₄	51.6 ₄	52.0 ₄	50.0 ₀	51.3 ₃
	TomekL+XGBoost	56.7 ₇	60 ₇	59.6 ₁	58.3 ₁₁	58.6 ₆
Ensemble	AdaBoost	56.6 ₇	56.6 ₄	55.9 ₈	56.6 ₉	56.4 ₇
	RF	53.3 ₄	56.6 ₇	56.0 ₅	53.4 ₇	54.8 ₆
Outlier	CBLOF	73.6 ₃	71.3 ₅	70.4 ₄	80.5 ₃	74.0 ₄
	COPOD	79.6 ₁	79.5 ₂	73.4 ₁	86.9 ₂	79.8 ₂
	KNN	74.3 ₃	74.8 ₃	70.4 ₂	84.1 ₃	75.9 ₃
	LMDD	58.1 ₃	58.7 ₇	62.9 ₅	64.6 ₁₀	61.1 ₆
	LOF	70.7 ₂	72.4 ₄	74.9 ₃	84.7 ₃	75.6 ₃
	LSAD	73.8 ₃	75.1 ₃	71.5 ₂	86.9 ₂	76.8 ₃
	MCD	68.9 ₁	78.5 ₂	72.6 ₇	75.7 ₃	73.9 ₃
	OCSVM	38.1 ₆	35.3 ₃	39.2 ₅	33.9 ₅	36.6 ₅
	PCA	65.9 ₅	71.2 ₄	68.4 ₄	80.5 ₃	71.5 ₄
	WSVM	53.3 ₅	56.6 ₄	58.0 ₈	53.4 ₇	55.3 ₆
Cost-sensitive	BBagg	82.2 ₄	85.5 ₉	88.1 ₈	87.7 ₁₃	85.8 ₉
Hybrid (Ensemble, Sampling)	BCascade	81.1 ₇	87.5 ₆	90.3 ₄	92.5₇	87.8₆
	BRF	82.7 ₇	86.1 ₄	87.7 ₄	89.1 ₈	86.4 ₆
	EasyE	81.3 ₅	88.9 ₅	85.2 ₇	87.6 ₈	85.7 ₆
	RUSBoost	82.2 ₅	89.3₇	87.9 ₅	88.5 ₉	87.0 ₆
	SelfPE	83.6₆	86.9 ₄	90.4 ₄	88.8 ₁₀	87.4 ₆
	UBagg	82.1 ₄	85.5 ₉	88.1 ₈	87.6 ₁₃	85.8 ₉
	OverBoost	79.5 ₃	86.7 ₇	89.6 ₅	84.1 ₁₄	85.0 ₇
	SMOTEBoost	80.3 ₆	83.8 ₇	89.9 ₇	86.0 ₁₀	85.0 ₈
	SMOTEBagg	53.3 ₅	53.3 ₄	52.0 ₄	53.4 ₇	53.0 ₅
	AdaCost	78.6 ₈	84.6 ₅	91.9₄	85.3 ₁₁	85.1 ₇
Hybrid (Ensemble, Cost-sensitive)	AdaUBoost	79.2 ₄	85.6 ₈	89.6 ₅	84.2 ₁₄	84.6 ₈
	WRF	72.6 ₁₀	74.4 ₄	80.5 ₅	83.3 ₁₀	77.7 ₇
Hybrid (Ensemble, Outlier)	FBagg	74.5 ₂	81.0 ₂	75.6 ₃	87.4 ₂	79.6 ₂
	IF	77.5 ₁	77.9 ₃	71.9 ₂	85.8 ₂	78.3 ₂
	LODA	63.8 ₄	60.9 ₄	61.8 ₄	67.0 ₈	63.4 ₅
	XGBOD	64.8 ₇	76.7 ₄	82.5 ₃	78.9 ₆	75.7 ₅

Table 11 Overview of the AUC scores (%) achieved on the Slovak construction bankruptcy datasets in the form of AUC_{std}

Category	Method	Data				Avg.
		SK_C_13	SK_C_14	SK_C_15	SK_C_16	
Sampling	ADASYN+SVM	70.3 ₄	81.4 ₃	86.1 ₆	85.7 ₁₁	80.9 ₆
	ADASYN+XGBoost	85.2 ₁₃	83.1 ₉	85.8 ₈	73.1 ₁₃	81.8 ₁₁
	BLSMOTE+SVM	72.2 ₉	76.4 ₈	85 ₁₁	83.0 ₁₄	79.4 ₁₀
	BLSMOTE+XGBoost	80.8 ₁₁	85.8 ₉	88.7 ₆	71.2 ₂₀	81.6 ₁₁
	SMOTE+SVM	70.1 ₄	81.6 ₇	86.1 ₆	86.0 ₁₁	80.9 ₇
	SMOTE+XGBoost	85.2 ₁₀	83.9 ₈	86.5 ₁	76.5 ₁₂	83.0 ₈
	SMOTETomek+SVC	71.1 ₅	82.2 ₃	83.9 ₁₀	86.4 ₁₁	80.9 ₇
	SMOTETomek+XGBoost	89.3₉	85.7 ₆	90.1 ₇	80.0 ₁₁	86.3 ₈
	SMOTEENN+SVC	76.0 ₁₁	82.8 ₇	85.8 ₆	86.1 ₁₁	82.7 ₉
	SMOTEENN+XGBoost	87.9 ₈	86.9 ₆	92.4 ₇	79.8 ₁₁	86.7 ₈
	ClusterC+SVM	48.1 ₁₄	59.8 ₁₀	80.3 ₆	71.5 ₁₂	64.9 ₁₀
	ClusterC+XGBoost	82.1 ₈	84.5 ₂	90.0 ₃	95.4 ₁	88.0 ₁₃
	NearMiss+SVM	67.7 ₈	78.8 ₉	85.2 ₇	82.4 ₁₀	78.5 ₈
	NearMiss+XGBoost	80.0 ₈	80.8 ₄	93.2 ₅	68.1 ₂₀	80.5 ₉
	TomekL+SVM	52.0 ₄	56.0 ₄	52.4 ₅	59.9 ₉	55.1 ₆
	TomekL+XGBoost	65.7 ₈	64.6 ₁₀	64.9 ₁₀	63.3 ₁₇	64.6 ₁₁
Ensemble	AdaBoost	63.7 ₈	61.2 ₇	59.8 ₁₀	60.0 ₁₄	61.2 ₁₀
	RF	57.8 ₈	61.5 ₁₃	57.3 ₁₀	56.7 ₉	58.3 ₁₀
Outlier	CBLOF	67.2 ₄	71.6 ₂	83.1 ₂	80.2 ₄	75.5 ₃
	COPOD	74.1 ₂	76.7 ₂	83.9 ₃	87.2 ₃	80.5 ₃
	KNN	68.9 ₂	72.0 ₃	85.4 ₂	90.4 ₂	79.2 ₂
	LMDD	56.8 ₆	53.5 ₄	69.4 ₅	60.5 ₁₁	60.0 ₆
	LOF	68.0 ₄	73.6 ₄	80.2 ₃	89.7 ₂	77.8 ₃
	LSAD	70.1 ₃	76.1 ₂	86.3 ₃	89.5 ₃	80.4 ₃
	MCD	71.7 ₃	63.5 ₃	68.7 ₂	64.8 ₄	67.2 ₃
	OCSVM	40.2 ₅	40.7 ₅	45.8 ₄	33.8 ₂	40.1 ₄
	PCA	67.4 ₄	68.1 ₄	81.8 ₂	78.2 ₄	73.8 ₃
Cost-sensitive	WSVM	61.9 ₄	66.5 ₁₁	59.9 ₁₀	60.0 ₁₄	62.1 ₁₀
Hybrid (Ensemble, Sampling)	BBagg	86.4 ₁₀	86.0 ₅	89.3 ₇	93.2 ₈	88.7 ₉
	BCascade	87.5 ₈	86.5 ₅	95.7₁	95.4 ₂	91.2₄
	BRF	86.2 ₈	85.4 ₅	91.1 ₇	93.0 ₆	88.9 ₇
	EasyE	87.8 ₈	84.5 ₃	91.9 ₆	89.8 ₁₀	88.5 ₇
	RUSBoost	85.1 ₉	85.6 ₄	90.4 ₆	91.3 ₁₀	88.0 ₇
	SelfPE	87.5 ₉	87.8₅	92.5 ₆	96.4₂	91.1 ₅
	UBagg	86.4 ₁₀	86.0 ₈	89.3 ₁₀	93.2 ₈	88.7 ₉
	OverBoost	84.9 ₁₃	85.2 ₆	89.4 ₇	89.0 ₁₁	87.1 ₉
	SMOTEBoost	86.3 ₁₀	84.9 ₇	89.7 ₇	89.1 ₁₁	87.5 ₉
Hybrid (Ensemble, Cost-sensitive)	SMOTEBagg	59.8 ₇	63.0 ₁₁	57.4 ₁₁	56.6 ₁₄	59.2 ₁₀
	AdaCost	85.1 ₁₂	82.1 ₆	88.3 ₆	86.4 ₁₀	85.5 ₉
	AdaUBoost	84.9 ₉	85.4 ₅	89.1 ₇	89.1 ₁₁	87.1 ₈
Hybrid (Ensemble, Outlier)	WRF	72.6 ₇	82.9 ₆	86.5 ₁₀	86.2 ₁₀	82.0 ₈
	FBagg	69.5 ₄	74.9 ₂	82.1 ₂	90.4 ₂	79.2 ₂
	IF	74.4 ₃	71.9 ₁	85.1 ₃	81.7 ₅	78.3 ₃
	LODA	60.3 ₅	62.6 ₆	68.9 ₄	68.7 ₈	65.1 ₆
	XGBOD	84.1 ₄	76.3 ₅	77.1 ₇	79.2 ₃	79.2 ₅

Table 12 Overview of the AUC scores (%) achieved on the Polish bankruptcy datasets in the form of AUC_{std}

Category	Method	Data					Avg.
		PL_01	PL_02	PL_03	PL_04	PL_05	
Sampling	ADASYN+SVM	79.3 ₅	72.7 ₃	74.4 ₂	71.8 ₁	75.9 ₃	74.8 ₃
	ADASYN+XGBoost	82.9 ₃	77.9 ₃	78.9 ₃	79.2 ₃	85.1 ₂	82.6 ₃
	BLSMOTE+SVM	76.7 ₄	71.0 ₃	69.7 ₃	69.6 ₂	75.7 ₃	72.5 ₃
	BLSMOTE+XGBoost	82.3 ₃	77.4 ₃	76.1 ₂	78.0 ₃	84.3 ₃	79.6 ₃
	SMOTE+SVM	79.1 ₄	72.5 ₃	73.7 ₃	70.1 ₂	76.2 ₃	74.3 ₃
	SMOTE+XGBoost	83.2 ₃	78.1 ₄	78.6 ₂	79.1 ₃	85.4 ₂	80.9 ₃
	SMOTETomek+SVC	79.4 ₄	72.1 ₃	74.1 ₃	70.0 ₂	75.6 ₃	74.2 ₃
	SMOTETomek+XGBoost	82.9 ₅	77.9 ₄	78.7 ₂	78.6 ₃	85.7 ₂	80.8 ₃
	SMOTEENN+SVC	78.7 ₄	70.7 ₃	73.5 ₂	71.5 ₂	75.9 ₂	74.1 ₂
	SMOTEENN+XGBoost	84.3₄	80.7 ₄	80.9 ₂	80.5 ₂	87.1₂	82.7₃
	ClusterC+SVM	66.0 ₃	60.8 ₄	63.9 ₄	60.4 ₃	64.4 ₃	63.1 ₃
	ClusterC+XGBoost	70.1 ₄	62.4 ₅	62.8 ₆	69.4 ₅	78.9 ₄	62.7 ₅
	NearMiss+SVM	50.6 ₃	50.1 ₄	59.7 ₃	62.5 ₄	70.0 ₂	58.6 ₃
	NearMiss+XGBoost	75.1 ₄	61.9 ₆	64.2 ₄	64.1 ₃	71.5 ₁	67.3 ₄
	TomekL+SVM	67.4 ₅	59.5 ₂	59.4 ₂	57.8 ₃	62.1 ₂	61.2 ₃
	TomekL+XGBoost	79.9 ₄	74.4 ₂	73.0 ₃	75.7 ₃	80.7 ₂	76.8 ₃
Ensemble	AdaBoost	73.5 ₅	62.3 ₂	64.2 ₂	66.5 ₄	76.7 ₄	68.6 ₃
	RF	74.5 ₄	67.8 ₃	65.1 ₂	64.6 ₂	76.2 ₄	69.6 ₃
Outlier	CBLOF	53.1 ₁	52.4 ₂	52.4 ₂	57.1 ₁	65.9 ₁	56.2 ₁
	COPOD	53.9 ₁	52.9 ₁	54.3 ₁	59.8 ₁	67.5 ₁	57.7 ₁
	KNN	52.9 ₁	52.8 ₁	53.0 ₁	58.4 ₁	68.2 ₂	57.0 ₁
	LMDD	50.4 ₁	50.5 ₁	50.6 ₁	53.5 ₄	52.5 ₂	51.5 ₂
	LOF	56.0 ₁	53.1 ₁	53.9 ₁	55.4 ₁	63.3 ₂	56.3 ₁
	LSAD	53.9 ₂	52.2 ₂	55.4 ₁	58.9 ₁	68.6 ₂	57.8 ₂
	MCD	67.4 ₉	53.9 ₃	52.4 ₁	56.7 ₂	66.4 ₁	59.3 ₃
	OCSVM	47.8 ₁	49.7 ₁	47.2 ₁	42.4 ₁	32.5 ₁	43.9 ₁
	PCA	53.1 ₁	50.5 ₁	52.0 ₁	56.7 ₁	65.6 ₁	55.5 ₁
	WSVM	75.6 ₄	67.9 ₃	65.7 ₃	65.3 ₂	78.7 ₃	70.7 ₃
Cost-sensitive	Hybrid (Ensemble, Sampling)	77.1 ₅	76.6 ₃	77.0 ₂	76.7 ₂	81.3 ₃	77.7 ₃
Hybrid (Ensemble, Cost-sensitive)	BCascade	75.3 ₆	66.2 ₄	69.3 ₄	68.6 ₃	80.3 ₂	71.9 ₄
	BRF	83.1 ₂	81.1₄	80.8 ₂	80.4 ₂	86.4 ₂	82.3 ₂
	EasyE	83.9 ₂	79.2 ₃	80.9₁	81.5₁	84.9 ₂	82.1 ₂
	RUSBoost	80.7 ₃	78.2 ₄	79.5 ₁	79.1 ₂	84.4 ₃	80.3 ₃
	SelfPE	78.5 ₃	68.2 ₄	68.6 ₂	70.2 ₅	80.7 ₃	73.2 ₃
	UBagg	77.1 ₅	76.6 ₃	77.0 ₂	76.7 ₂	81.3 ₃	77.7 ₃
	OverBoost	81.8 ₃	78.1 ₂	78.7 ₂	80.7 ₁	84.7 ₂	80.8 ₂
	SMOTEBoost	79.9 ₃	71.6 ₃	75.0 ₃	72.2 ₃	83.4 ₂	76.4 ₃
	SMOTEBagg	57.3 ₂	56.4 ₂	56.6 ₁	57.9 ₂	69.9 ₃	59.6 ₂
	AdaCost	79.8 ₄	72.8 ₄	76.8 ₁	74.2 ₃	81.6 ₂	77.0 ₃
	AdaUBoost	83.7 ₄	78.0 ₃	79.0 ₂	80.7 ₂	84.5 ₃	81.1 ₃
	WRF	80.0 ₄	72.4 ₂	74.9 ₂	71.7 ₂	75.8 ₃	74.9 ₃
	Hybrid (Ensemble, Outlier)	57.0 ₁	53.7 ₁	54.7 ₁	54.8 ₁	65.4 ₂	57.1 ₁
	IF	55.0 ₁	53.4 ₁	56.3 ₂	59.2 ₁	68.4 ₁	58.4 ₁
	LODA	51.8 ₂	51.0 ₁	50.8 ₂	52.5 ₂	58.4 ₅	52.9 ₂
	XGBOD	50.0 ₀	50.0 ₀	50.0 ₀	50.0 ₀	50.0 ₀	50.0 ₀

Table 13 Overview of the AUC scores (%) achieved on the Taiwan and Bosnia and Herzegovina bankruptcy datasets in the form of AUC_{std}

Category	Method	Data	
		TW	B&H
Sampling	ADASYN+SVM	84.5 ₃	93.0 ₅
	ADASYN+XGBoost	85.6 ₃	94.0 ₆
	BLSMOTE+SVM	85.6 ₃	92.0 ₆
	BLSMOTE+XGBoost	84.3 ₄	92.5 ₇
	SMOTE+SVM	86.6 ₂	92.0 ₈
	SMOTE+XGBoost	85.6 ₃	92.5 ₈
	SMOTETomek+SVC	86.4 ₂	93.0 ₆
	SMOTETomek+XGBoost	85.9 ₃	95.0 ₆
	SMOTEENN+SVC	85.9 ₂	91.5 ₆
	SMOTEENN+XGBoost	87.0 ₃	94.5 ₅
	ClusterC+SVM	82.4 ₂	91.5 ₅
	ClusterC+XGBoost	84.8 ₂	95.0 ₅
	NearMiss+SVM	79.6 ₄	88.5 ₉
	NearMiss+XGBoost	84.2 ₄	92.0 ₇
	TomekL+SVM	61.7 ₃	90.5 ₈
	TomekL+XGBoost	66.8 ₃	92.0 ₇
Ensemble	AdaBoost	64.5 ₃	95.5₆
	RF	59.8 ₂	94.5 ₇
Outlier	CBLOF	65.3 ₁	53.0 ₁₁
	COPOD	73.7 ₁	43.4 ₇
	KNN	60.7 ₁	49.1 ₆
	LMDD	53.1 ₃	53.1 ₅
	LOF	57.5 ₂	53.2 ₆
	LSAD	59.6 ₁	47.4 ₇
	MCD	64.4 ₁	56.8 ₂
	OCSVM	77.2 ₂	69.4 ₆
	PCA	63.8 ₁	48.9 ₆
Cost-sensitive	WSVM	61.7 ₂	96.5 ₅
Hybrid (Ensemble, Sampling)	BBagg	86.5 ₃	89.5 ₁₁
	BCascade	86.0 ₂	91.5 ₈
	BRF	87.2₂	94.0 ₇
	EasyE	86.1 ₃	94.0 ₇
	RUSBoost	86.5 ₂	93.0 ₇
	SelfPE	86.1 ₃	91.5 ₈
	UBagg	86.5 ₃	94.5 ₇
	OverBoost	85.8 ₄	94.0 ₇
	SMOTEBoost	84.9 ₃	93.0 ₆
	SMOTEBagg	68.8 ₃	94.0 ₇
Hybrid (Ensemble, Cost-sensitive)	AdaCost	83.7 ₄	93.0 ₅
	AdaUBoost	85.7 ₄	94.0 ₇
	WRF	86.3 ₃	91.5 ₇
Hybrid (Ensemble, Outlier)	FBagg	58.6 ₁	51.9 ₆
	IF	72.3 ₂	56.3 ₅
	LODA	58.2 ₈	55.2 ₅
	XGBOD	70.4 ₂	94.5 ₄

Appendix D: FNR score results

In addition to the GM and AUC score, we also provide a FNR score, representing the error of incorrectly classifying bankrupt cases into non-bankrupt class. In Tables 14, 15, 16, 17, we provide the FNR values for all datasets.

Table 14 Overview of the FNR scores (%) achieved on the Slovak manufacture bankruptcy datasets in the form of FNR_{std}

Category	Method	Data				Avg.
		SK_M_13	SK_M_14	SK_M_15	SK_M_16	
Sampling	ADASYN+SVM	26.7 ₂₀	23.3 ₁₇	18.7 ₁₇	23.3 ₂₀	23.0 ₁₈
	ADASYN+XGBoost	26.7 ₈	26.7 ₁₃	14.7 ₁₃	53.3 ₂₇	30.3 ₁₅
	BLSMOTE+SVM	53.3 ₂₄	43.3 ₂₅	42 ₂₂	63.3 ₂₂	50.5 ₂₃
	BLSMOTE+XGBoost	43.3 ₁₃	40.0 ₂₃	45.3 ₁₇	56.7 ₃₃	46.3 ₂₁
	SMOTE+SVM	26.7 ₂₀	33.3 ₁₅	18.7 ₁₇	23.3 ₂₀	25.5 ₁₈
	SMOTE+XGBoost	26.7 ₈	26.7 ₁₃	14.7 ₁₃	46.7 ₃₄	28.7 ₁₇
	SMOTETomek+SVC	26.7 ₂₀	33.3 ₁₅	18.7 ₁₇	23.3 ₂₀	25.5 ₁₈
	SMOTETomek+XGBoost	23.3 ₈	20.0 ₇	14.7 ₁₃	40.0 ₃₉	24.5 ₁₇
	SMOTEENN+SVC	23.3 ₁₇	23.3 ₁₇	18.7 ₁₇	23.3 ₂₀	22.2 ₁₈
	SMOTEENN+XGBoost	23.3 ₈	20.0 ₁₂	11.3 ₉	43.3 ₁₃	24.5 ₁₁
	ClusterC+SVM	33.3 ₁₁	50.0 ₂₁	50.0 ₉	30.0 ₂₇	40.8 ₁₇
	ClusterC+XGBoost	13.3 ₁₂	6.7 ₈	7.3 ₉	13.3 ₁₆	10.2 ₁₂
	NearMiss+SVM	40.0 ₁₇	40.0 ₁₇	49.3 ₁₇	13.3 ₈	35.7 ₁₇
	NearMiss+XGBoost	23.3 ₁₃	20.0 ₇	11.3 ₉	16.7 ₂₁	17.8 ₁₃
	TomekL+SVM	96.7 ₇	96.7 ₇	96.0 ₈	100.0 ₀	97.3 ₅
	TomekL+XGBoost	86.7 ₁₂	80.0 ₁₂	80.7 ₁	83.3 ₂₁	82.7 ₁₂
Ensemble	AdaBoost	86.7 ₁₂	86.7 ₇	88.0 ₁₆	86.7 ₁₆	87.0 ₁₃
	RF	93.3 ₈	86.7 ₁₂	88.0 ₁₀	93.3 ₁₃	90.3 ₁₁
Outlier	CBLOF	32.7 ₆	37.8 ₈	39.2 ₈	19.3 ₅	32.7 ₇
	COPOD	23.3 ₂	23.3 ₄	34.7 ₂	7.2 ₃	22.1 ₂
	KNN	31.7 ₄	31.1 ₅	39.1 ₄	12.1 ₄	28.5 ₄
	LMDD	78.6 ₉	77.4 ₁₉	61.4 ₁₇	64.7 ₂₅	70.6 ₁₈
	LOF	39.0 ₄	36.0 ₈	30.1 ₅	10.7 ₄	28.9 ₅
	LSAD	32.7 ₈	37.8 ₅	37.2 ₂	9.6 ₃	29.3 ₅
	MCD	42.2 ₂	23.4 ₃	34.4 ₁₃	28.6 ₄	32.1 ₅
	OCSVM	63.5 ₁₁	69.4 ₆	50.8 ₁₀	65.4 ₄	62.3 ₈
	PCA	48.3 ₉	37.8 ₆	43.0 ₇	19.3 ₅	37.1 ₇
	WSVM	20.0 ₁₉	40.0 ₈	23.3 ₈	23.3 ₂₀	26.7 ₁₄
Cost-sensitive	BBagg	26.7 ₈	20.0 ₁₆	15.3 ₁₅	20.0 ₂₇	20.5 ₁₇
	BCascade	10.0 ₁₃	6.7 ₈	7.3 ₉	6.7 ₁₃	7.7 ₁₁
Hybrid (Ensemble, Cost-sensitive)	BRF	20.0 ₁₂	13.3 ₇	11.3 ₉	13.3 ₁₆	14.5 ₁₁
	EasyE	16.7 ₁₁	6.7 ₈	15.3 ₁₅	13.3 ₁₆	13.0 ₁₂
	RUSBoost	23.3 ₈	10.0 ₁₃	7.3 ₉	13.3 ₁₆	13.5 ₁₂
	SelfPE	13.3 ₁₂	10.0 ₈	7.3 ₉	13.3 ₁₆	11.0 ₁₂
	UBagg	26.7 ₈	20.0 ₁₆	15.3 ₁₅	20.0 ₂₇	20.5 ₁₇
	OverBoost	30.0 ₇	13.3 ₁₂	11.3 ₉	30.0 ₂₇	21.2 ₁₄
	SMOTEBoost	23.3 ₁₃	23.3 ₁₃	14.7 ₁₃	23.3 ₂₀	21.2 ₁₅
	SMOTEBagg	93.3 ₈	93.3 ₈	96.0 ₈	93.3 ₁₃	94.0 ₉
	AdaCost	26.7 ₁₃	16.7 ₁₁	7.3 ₉	13.3 ₁₄	16.0 ₁₅
	AdaUBoost	30.0 ₇	20.0 ₁₆	11.3 ₉	30.0 ₂₇	22.8 ₁₅
Hybrid (Ensemble, Outlier)	WRF	93.3 ₈	86.7 ₇	84.0 ₁₅	93.3 ₁₃	89.3 ₁₁
	FBagg	31.1 ₃	18.4 ₁	28.8 ₅	4.7 ₂	20.7 ₃
	IF	76.7 ₆	63.5 ₇	65.1 ₁₈	84.3 ₉	72.4 ₁₀
	LODA	55.6 ₁₀	71.7 ₉	59.2 ₈	53.5 ₁₇	60.0 ₁₁
	XGBOD	70.4 ₁₄	46.5 ₈	35.0 ₆	42.2 ₁₂	48.5 ₁₀

Table 15 Overview of FNR scores (%) achieved on the Slovak construction bankruptcy datasets in the form of FNR_{std}

Category	Method	Data				Avg.
		SK_C_13	SK_C_14	SK_C_15	SK_C_16	
Sampling	ADASYN+SVM	4.0 ₈	23.3 ₈	10.0 ₁₂	10.0 ₂₀	11.8 ₁₂
	ADASYN+XGBoost	20.0 ₂₅	26.7 ₁₇	25.0 ₁₆	50.0 ₂₆	30.4 ₂₁
	BLSMOTE+SVM	28.0 ₂₀	23.3 ₁₃	15.0 ₂₀	30.0 ₂₇	24.1 ₂₀
	BLSMOTE+XGBoost	32.0 ₂₀	23.3 ₁₇	20.0 ₁₀	56.7 ₃₉	33.0 ₂₂
	SMOTE+SVM	4.0 ₈	23.3 ₁₇	10.0 ₁₂	10.0 ₂₀	11.8 ₁₃
	SMOTE+XGBoost	20.0 ₁₈	23.3 ₁₃	25.0 ₀	43.3 ₂₅	27.9 ₁₅
	SMOTETomek+SVC	4.0 ₈	23.3 ₈	15.0 ₂₀	10.0 ₂₀	13.1 ₁₄
	SMOTETomek+XGBoost	12.0 ₁₆	20.0 ₁₂	15.0 ₁₂	36.7 ₂₂	20.9 ₁₆
	SMOTEENN+SVC	16.0 ₂₃	20.0 ₁₂	10.0 ₁₂	10.0 ₂₀	14.0 ₁₇
	SMOTEENN+XGBoost	12.0 ₁₆	13.3 ₁₂	10.0 ₁₂	36.7 ₂₂	18.0 ₁₆
	ClusterC+SVM	40.0 ₂₂	46.7 ₁₆	15.0 ₁₂	23.3 ₂₀	31.2 ₁₀
	ClusterC+XGBoost	12.0 ₁₆	6.7 ₈	0.0 ₀	0.0 ₀	4.7 ₆
	NearMiss+SVM	28.0 ₁₆	13.3 ₁₂	10.0 ₁₂	10.0 ₂₀	15.3 ₁₅
	NearMiss+XGBoost	12.0 ₁₆	6.7 ₈	5.0 ₁₀	36.7 ₃₁	15.1 ₁₆
	TomekL+SVM	96.0 ₈	86.7 ₇	95.0 ₁₀	18.0 ₁₆	89.4 ₁₀
	TomekL+XGBoost	68.0 ₁₆	70.0 ₁₉	70.0 ₁₉	73.3 ₃₃	70.3 ₂₂
Ensemble	AdaBoost	72.0 ₁₆	76.7 ₁₃	80.0 ₁₉	80.0 ₂₇	77.2 ₁₉
	RF	84.0 ₁₅	76.7 ₂₅	85.0 ₂₀	86.7 ₉	83.1 ₁₉
Outlier	CBLOF	46.5 ₆	37.8 ₃	13.4 ₃	19.3 ₅	29.2 ₄
	COPOD	37.4 ₃	32.7 ₄	16.5 ₅	8.5 ₅	23.8 ₄
	KNN	43.2 ₄	37.8 ₄	10.0 ₂	0.0 ₀	22.8 ₃
	LMDD	81.6 ₁₅	89.1 ₁₀	55.8 ₁₁	74.5 ₂₅	75.3 ₁₅
	LOF	45.4 ₅	33.0 ₇	20.2 ₄	1.1 ₂	24.9 ₅
	LSAD	27.3 ₆	18.7 ₈	11.7 ₃	1.2 ₂	14.7 ₅
	MCD	37.9 ₃	53.8 ₄	43.2 ₄	50.1 ₆	46.3 ₄
	OCSVM	48.0 ₁₃	42.9 ₁₀	51.9 ₁₂	66.4 ₉	52.3 ₁₁
Cost-sensitive	PCA	45.8 ₆	44.6 ₈	16.6 ₂	23.9 ₅	32.7 ₅
	WSVM	12.0 ₁₆	16.7 ₁₁	20.0 ₁₉	23.3 ₂₀	18.0 ₁₆
Hybrid (Ensemble, Sampling)	BBagg	16.0 ₂₀	16.7 ₁₈	15.0 ₂₀	6.7 ₁₃	13.6 ₁₈
	BCascade	12.0 ₁₆	16.7 ₁₁	15.0 ₂₀	6.7 ₁₃	13.6 ₁₈
	BRF	12.0 ₁₆	13.3 ₁₂	10.0 ₁₂	6.7 ₁₃	10.5 ₁₄
	EasyE	8.0 ₁₆	13.3 ₇	5.0 ₁₀	10.0 ₂₀	9.1 ₁₃
	RUSBoost	20.0 ₁₈	13.3 ₇	15.0 ₁₂	10.0 ₂₀	14.6 ₁₄
	SelfPE	12.0 ₁₆	10.0 ₁₃	10.0 ₁₂	0.0 ₀	8.0 ₁₀
	UBagg	16.0 ₂₀	16.7 ₁₈	15.0 ₂₀	6.7 ₁₃	13.6 ₁₈
	OverBoost	20.0 ₂₅	13.3 ₁₂	15.0 ₁₂	16.7 ₂₁	16.3 ₁₈
	SMOTEBoost	16.0 ₂₀	20.0 ₁₆	15.0 ₁₂	16.7 ₂₁	16.9 ₁₇
	SMOTEBagg	80.0 ₁₃	73.3 ₂₀	85.0 ₂₀	86.7 ₂₇	81.2 ₂₀
Hybrid (Ensemble, Cost-sensitive)	AdaCost	16.0 ₂₃	20.0 ₁₂	15.0 ₁₂	16.7 ₂₁	16.9 ₁₇
	AdaUBoost	20.0 ₁₈	10.0 ₁₃	15.0 ₁₂	16.7 ₂₁	15.4 ₁₆
	WRF	76.0 ₈	66.7 ₂₁	80.0 ₁₉	80.0 ₂₇	75.7 ₁₉
Hybrid (Ensemble, Outlier)	FBagg	42.3 ₆	31.8 ₃	16.8 ₄	0.0 ₀	22.7 ₃
	IF	75.6 ₁₀	63.8 ₁₁	80.3 ₁₁	61.8 ₈	70.4 ₁₀
	LODA	66.6 ₉	60.8 ₉	44.2 ₆	47.8 ₁₇	54.8 ₁₀
	XGBOD	31.2 ₆	47.0 ₉	45.6 ₁₂	41.5 ₅	41.4 ₈

Table 16 Overview of FNR scores (%) achieved on the Polish bankruptcy datasets in the form of FNR_{std}

Category	Method	Data					Avg.
		PL_01	PL_02	PL_03	PL_04	PL_05	
Sampling	ADASYN+XGBoost	31.4 ₅	41.5 ₄	36.6 ₆	36.5 ₄	23.7 ₃	33.9 ₄
	BLSMOTE+SVM	38.8 ₈	45.0 ₆	34.7 ₅	43.7 ₃	31.0 ₆	38.6 ₅
	BLSMOTE+XGBoost	28.0 ₅	42.3 ₅	37.2 ₅	39.0 ₄	26.6 ₅	34.6 ₅
	SMOTE+SVM	32.8 ₉	37.0 ₃	39.4 ₅	33.8 ₃	25.9 ₆	33.8 ₅
	SMOTE+XGBoost	31.0 ₆	42.2 ₇	37.0 ₃	37.3 ₅	23.4 ₃	34.2 ₅
	SMOTETomek+SVC	32.5 ₉	37.8 ₃	38.6 ₅	34.2 ₃	26.6 ₇	33.9 ₅
	SMOTETomek+XGBoost	31.3 ₉	41.0 ₇	31.5 ₃	35.1 ₅	23.4 ₃	32.5 ₆
	SMOTEENN+SVC	28.4 ₈	31.0 ₆	32.5 ₄	36.5 ₄	28.0 ₃	31.3 ₅
	SMOTEENN+XGBoost	26.5 ₆	28.7 ₇	26.9 ₃	27.2 ₄	19.8 ₃	25.8 ₅
	ClusterC+SVM	24.0 ₇	43.8 ₁₁	35.4 ₈	40.4 ₈	45.9 ₇	37.9 ₈
	ClusterC+XGBoost	13.3 ₄	12.0 ₆	9.7 ₅	9.9 ₂	13.7 ₇	11.7 ₅
	NearMiss+SVM	41.3 ₂	39.5 ₉	42.0 ₅	38.1 ₆	24.1 ₄	37.0 ₅
	NearMiss+XGBoost	18.8 ₄	26.0 ₆	21.2 ₃	6.6 ₄	3.4 ₁	15.2 ₄
	TomekL+SVM	63.8 ₉	79.8 ₄	78.8 ₂	81.6 ₄	72.7 ₃	75.3 ₅
	TomekL+XGBoost	39.9 ₈	50.8 ₄	53.1 ₄	48.0 ₅	37.8 ₄	45.9 ₅
Ensemble	AdaBoost	52.4 ₉	74.8 ₃	70.5 ₄	66.0 ₇	45.1 ₇	61.8 ₆
	RF	50.9 ₈	64.2 ₅	69.7 ₃	70.7 ₄	46.8 ₈	60.5 ₆
Outlier	CBLOF	73.9 ₁	78.8 ₁	75.5 ₃	65.7 ₁	48.3 ₂	68.5 ₁
	COPOD	73.5 ₁	75.3 ₁	73.2 ₁	64.7 ₁	54.5 ₁	68.2 ₁
	KNN	74.2 ₁	74.6 ₁	74.1 ₃	63.7 ₁	43.9 ₄	66.1 ₂
	LMDD	97.9 ₁	96.2 ₃	92.7 ₅	85.1 ₁₂	93.8 ₃	93.1 ₅
	LOF	67.5 ₁	73.1 ₂	72.0 ₁	69.2 ₁	53.5 ₃	67.0 ₂
	LSAD	51.6 ₄	56.8 ₁₇	43.2 ₁₃	47.5 ₁₀	30.4 ₄	45.9 ₁₀
	MCD	45.2 ₁₇	72.2 ₅	75.2 ₁	66.4 ₄	47.2 ₁	61.3 ₆
	OCSVM	54.3 ₀	50.7 ₁	45.8 ₁	65.4 ₁	65.3 ₁	56.3 ₁
	PCA	73.9 ₁	79.0 ₁	76.1 ₁	66.8 ₁	49.0 ₁	69.0 ₁
Cost-sensitive	WSVM	30.6 ₈	39.2 ₄	35.8 ₄	41.9 ₅	28.8 ₆	35.3 ₅
Hybrid (Ensemble, Sampling)	BBagg	34.3 ₁₀	36.0 ₅	31.7 ₂	32.8 ₂	27.1 ₅	32.4 ₅
	BCascade	37.9 ₂₁	18.2 ₈	23.2 ₇	30.1 ₅	22.2 ₇	26.3 ₁₀
	BRF	17.3 ₄	18.2 ₇	16.6 ₃	17.9 ₃	12.9 ₃	16.6 ₄
	EasyE	15.1 ₄	19.0 ₆	18.4 ₁	17.3 ₂	15.9 ₄	17.1 ₃
	RUSBoost	26.6 ₅	28.5 ₇	26.1 ₂	27.2 ₃	21.5 ₅	26.0 ₅
	SelfPE	21.0 ₈	29.2 ₁₂	30.9 ₈	27.4 ₁₃	27.8 ₅	27.3 ₉
	UBagg	34.3 ₁₀	36.0 ₅	31.7 ₂	32.8 ₂	27.1 ₅	32.4 ₅
	OverBoost	22.9 ₇	30.5 ₄	28.9 ₄	25.0 ₂	20.0 ₄	25.5 ₄
	SMOTEBoost	33.6 ₆	51.2 ₆	38.2 ₅	35.5 ₆	24.1 ₄	36.5 ₅
	SMOTEBagg	85.3 ₄	87.0 ₄	86.3 ₂	83.3 ₃	57.8 ₅	79.9 ₃
	AdaCost	18.4 ₆	24.7 ₆	14.3 ₃	28.5 ₄	23.7 ₅	21.9 ₅
	AdaUBoost	23.9 ₇	31.0 ₆	30.5 ₄	27.4 ₃	21.2 ₄	26.8 ₅
	WRF	48.3 ₈	64.0 ₆	67.9 ₅	68.7 ₄	41.0 ₅	58.0 ₆
	FBagg	66.2 ₁	72.5 ₁	70.6 ₁	70.4 ₁	49.1 ₄	65.8 ₂
	IF	48.0 ₇	50.4 ₃	50.6 ₅	49.3 ₁₂	66.4 ₁₂	52.9 ₈
Hybrid (Ensemble, Outlier)	LODA	85.7 ₃	90.8 ₂	88.7 ₆	87.1 ₄	72.1 ₁₃	84.9 ₅
	XGBOD	100.0 ₀	100.0 ₀	100.0 ₀	100.0 ₀	100.0 ₀	100.0 ₀

Table 17 Overview of FNR scores (%) achieved on the Taiwan and Bosnia and Herzegovina bankruptcy datasets in the form of FNR_{std}

Category	Method	Data	
		TW	B&H
Sampling	ADASYN+SVM	12.3 ₅	6.0 ₈
	ADASYN+XGBoost	15.5 ₆	6.0 ₅
	BLSMOTE+SVM	15.0 ₅	8.0 ₇
	BLSMOTE+XGBoost	22.7 ₇	10.0 ₉
	SMOTE+SVM	11.4 ₄	10.0 ₁₁
	SMOTE+XGBoost	18.2 ₇	8.0 ₁₀
	SMOTETomek+SVC	11.8 ₄	8.0 ₇
	SMOTETomek+XGBoost	15.9 ₇	4.0 ₅
	SMOTEENN+SVC	10.0 ₄	8.0 ₇
	SMOTEENN+XGBoost	11.8 ₆	4.0 ₅
	ClusterC+SVM	15.0 ₃	6.0 ₅
	ClusterC+XGBoost	7.7₄	4.0 ₅
	NearMiss+SVM	31.4 ₇	18.0 ₁₃
	NearMiss+XGBoost	18.6 ₇	10.0 ₉
	TomekL+SVM	75.0 ₅	14.0 ₁₂
	TomekL+XGBoost	65.0 ₅	12.0 ₁₂
Ensemble	AdaBoost	69.5 ₅	4.0 ₅
	RF	80.0 ₃	6.0 ₈
Outlier	CBLOF	50.4 ₁	71.9 ₇
	COPOD	40.7 ₁	86.0 ₂
	KNN	59.5 ₁	77.9 ₅
	LMDD	92.8 ₆	84.0 ₃
	LOF	65.3 ₂	78.0 ₄
	LSAD	66.0 ₂	52.9 ₆
	MCD	51.7 ₂	69.0 ₃
	OCSVM	45.7 ₃	61.2 ₁₁
	PCA	53.3 ₁	78.3 ₅
	WSVM	12.7 ₅	12.0 ₁₀
Cost-sensitive	WSVM	12.7 ₅	12.0 ₁₀
Hybrid (Ensemble, Sampling)	BBagg	16.4 ₆	14.0 ₁₅
	BCascade	11.8 ₈	10.0 ₉
	BRF	10.9 ₄	6.0 ₈
	EasyE	12.3 ₅	6.0 ₈
	RUSBoost	14.5 ₅	6.0 ₈
	SelfPE	17.3 ₇	10.0 ₉
	UBagg	16.4 ₆	6.0 ₈
	OverBoost	13.2 ₇	6.0 ₈
	SMOTEBoost	14.1 ₆	8.0 ₇
	SMOTEBagg	60.9 ₆	6.0 ₈
Hybrid (Ensemble, Cost-sensitive)	AdaCost	14.1 ₅	4.0 ₅
	AdaUBoost	14.5 ₇	6.0 ₈
	WRF	75.9 ₄	4.0 ₅
Hybrid (Ensemble, Outlier)	FBagg	63.3 ₀	77.3 ₅
	IF	66.0 ₆	45.7 ₇
	LODA	64.4 ₁₄	81.6 ₅
	XGBOD	58.8 ₃	2.7₄

Acknowledgements This work was supported by the Slovak Research and Development Agency under the contract APVV-21-0318 and by the Ministry of Education, Science, Research, and Sport of the Slovak Republic under contract no. VEGA 1/0174/24. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Author contributions P.G. conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared tables, authored and reviewed drafts of the article, and approved the final version of the manuscript. R.K. conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, and approved the final version of the manuscript. M.Z. analyzed state-of-the-art literature, prepared figures, authored and reviewed article drafts, and approved the final draft. P.D. conceived and designed the experiments, authored and reviewed article drafts, and approved the final version of the manuscript.

Funding Open access funding provided by The Ministry of Education, Science, Research and Sport of the Slovak Republic in cooperation with Centre for Scientific and Technical Information of the Slovak Republic

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no Conflict of interest

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Ainan UH, Por LY, Chen Y-L, Yang J, Ku CS (2024) Advancing bankruptcy forecasting with hybrid machine learning techniques: insights from an unbalanced polish dataset. *IEEE Access*
- Al Helal M, Haydar MS, Mostafa SAM (2016) Algorithms efficiency measurement on imbalanced data using geometric mean and cross validation. In: 2016 International workshop on computational intelligence (IWCi), pp 110–114. <https://doi.org/10.1109/IWCI.2016.7860349>. IEEE
- Alaka HA, Oyedele LO, Owolabi HA, Kumar V, Ajayi SO, Akinade OO, Bilal M (2018) Systematic review of bankruptcy prediction models: towards a framework for tool selection. *Expert Syst Appl* 94:164–184. <https://doi.org/10.1016/j.eswa.2017.10.040>
- Altman EI (1968) Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *J Financ* 23(4):589–609
- Amirshahi B, Lahmiri S (2024) Bankruptcy prediction using optimal ensemble models under balanced and imbalanced data. *Expert Syst*, 13599
- Arafa A, El-Fishawy N, Badawy M, Radad M (2022) Rn-smote: Reduced noise smote based on dbscan for enhancing imbalanced data classification. *J King Saud Univ-Comput Inform Sci* 34(8):5059–5074. <https://doi.org/10.1016/j.jksuci.2022.06.005>
- Arning A, Agrawal R, Raghavan P (1996) A linear method for deviation detection in large databases. *KDD* 1141:972–981
- Baik B, Han S-Y, Kim BH, Oh S (2022) Financial reporting quality of privately held firms: evidence from private corporations versus limited companies. *Asia-Pac J Account Econ* 29(5):1184–1207
- Batista GE, Bazzan AL, Monard MC et al (2003) Balancing training data for automated annotation of keywords: a case study. *Wob* 3:10–18
- Beaver WH (1966) Financial ratios as predictors of failure. *J Account Res*, pp 71–111

- Breiman L (1996) Bagging predictors. *Mach Learn* 24:123–140. <https://doi.org/10.1007/BF00058655>
- Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and regression trees. Wadsworth and Brooks, Monterey
- Breunig MM, Kriegel H-P, Ng RT, Sander J (2000) Lof: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD international conference on management of data, pp 93–104. <https://doi.org/10.1145/342009.335388>
- Chawla NV, Lazarevic A, Hall LO, Bowyer KW (2003) Smoteboost: Improving prediction of the minority class in boosting. In: Knowledge Discovery in Databases: PKDD 2003: 7th European conference on principles and practice of knowledge discovery in databases, Cavtat-Dubrovnik, Croatia, September 22–26, 2003. Proceedings 7, pp 107–119. https://doi.org/10.1007/978-3-540-39804-2_12. Springer
- Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) Smote: synthetic minority over-sampling technique. *J Artif Intell Res* 16:321–357. <https://doi.org/10.1613/jair.953>
- Chen C, Breiman L (2004) Using random forest to learn imbalanced data. University of California, Berkeley
- Chen H-L, Yang B, Wang G, Liu J, Xu X, Wang S-J, Liu D-Y (2011) A novel bankruptcy prediction model based on an adaptive fuzzy k-nearest neighbor method. *Knowl-Based Syst* 24(8):1348–1359. <https://doi.org/10.1016/j.knosys.2011.06.008>
- Chen Z, Chen W, Shi Y (2020) Ensemble learning with label proportions for bankruptcy prediction. *Expert Syst Appl* 146:113155. <https://doi.org/10.1016/j.eswa.2019.113155>
- Chen T-K, Liao H-H, Chen G-D, Kang W-H, Lin Y-C (2023) Bankruptcy prediction using machine learning models with the text-based communicative value of annual reports. *Expert Syst Appl* 233:120714
- Chen T, Guestrin C (2016) Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd Acm Sigkdd international conference on knowledge discovery and data mining, pp 785–794
- Cheraghali H, Molnár P (2023) Sme default prediction: a systematic methodology-focused review. *J Small Business Manag*, pp 1–59
- Dasilas A, Rigani A (2024) Machine learning techniques in bankruptcy prediction: a systematic literature review. *Expert Syst Appl* 255:124761
- Deepthi AS (2014) Anomaly detection using principal component analysis
- Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7:1–30
- Ding Y, Song X, Zen Y (2008) Forecasting financial condition of Chinese listed companies based on support vector machine. *Expert Syst Appl* 34(4):3081–3089
- Ding H, Sun Y, Wang Z, Huang N, Shen Z, Cui X (2023) Rgan-el: a gan and ensemble learning-based hybrid approach for imbalanced data classification. *Inform Process Manag* 60(2):103235. <https://doi.org/10.1016/j.ipm.2022.103235>
- Ding H, Sun Y, Huang N, Shen Z, Wang Z, Iftexhar A, Cui X (2023) Rvgan-tl: A generative adversarial networks and transfer learning-based hybrid approach for imbalanced data classification. *Inf Sci* 629:184–203
- Ding R, Zhou Y, Xu J, Xie Y, Liang Q, Ren H, Wang Y, Chen Y, Wang L, Huang M (2023) Cross-hospital sepsis early detection via semi-supervised optimal transport with self-paced ensemble. *IEEE J Biomed Health Inform*. <https://doi.org/10.1109/JBHI.2023.3253208>
- Dovile Kuiziniene RD, Tomas Krilavicius Maskeliunas R (2022) Systematic review of financial distress identification using artificial intelligence methods. *Appl Artif Intell* 36(1):2138124. <https://doi.org/10.1080/08839514.2022.2138124>
- Drotár P, Gnip P, Zoričák M, Gazda V (2019) Small-and medium-enterprises bankruptcy dataset. *Data Brief*. <https://doi.org/10.1016/j.dib.2019.104360>
- El Madou K, Marso S, El Kharrim M, El Merouani M (2024) Evolutions in machine learning technology for financial distress prediction: a comprehensive review and comparative analysis. *Expert Syst* 41(2):13485
- Fan W, Stolfo SJ, Zhang J, Chan PK (1999) Adacost: misclassification cost-sensitive boosting. *Icml* 99:97–105
- Figini S, Bonelli F, Giovannini E (2017) Solvency prediction for small and medium enterprises in banking. *Decis Support Syst* 102:91–97. <https://doi.org/10.1016/j.dss.2017.08.001>
- Fitzpatrick PJ (1932) A comparison of the ratios of successful industrial enterprises with those of failed companies
- Freund Y, Schapire RE (1995) A decision-theoretic generalization of on-line learning and an application to boosting. In: European conference on computational learning theory, pp 23–37. https://doi.org/10.1007/3-540-59119-2_166. Springer
- Friedman M (1937) The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *J Am Stat Assoc* 32(200):675–701
- Gurnani I, Tandian FS, Anggreainy MS et al (2021) Predicting company bankruptcy using random forest method. In: 2021 2nd international conference on artificial intelligence and data sciences (AiDAS), pp 1–5. <https://doi.org/10.1109/AiDAS53897.2021.9574384>. IEEE

- Han H, Wang W-Y, Mao B-H (2005) Borderline-smote: a new over-sampling method in imbalanced data sets learning. In: International conference on intelligent computing. Springer, pp 878–887. <https://doi.org/10.1007/11538059>
- He Z, Xu X, Deng S (2003) Discovering cluster-based local outliers. *Pattern Recogn Lett* 24(9–10):1641–1650. [https://doi.org/10.1016/S0167-8655\(03\)00003-5](https://doi.org/10.1016/S0167-8655(03)00003-5)
- He H, Bai Y, Garcia EA, Li S (2008) Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE international joint conference on neural networks (IEEE World Congress on Computational Intelligence). IEEE, pp 1322–1328. <https://doi.org/10.1109/IJCNN.2008.4633969>
- Hido S, Kashima H, Takahashi Y (2009) Roughly balanced bagging for imbalanced data. *Stat Anal Data Min: ASA Data Sci J* 2(5–6):412–426. <https://doi.org/10.1002/sam.10061>
- Ho TK (1995) Random decision forests. In: Proceedings of 3rd international conference on document analysis and recognition. IEEE, 1, pp 278–282. <https://doi.org/10.1109/ICDAR.1995.598994>
- Hubert M, Debruyne M, Rousseeuw PJ (2018) Minimum covariance determinant and extensions. *Wiley Interdiscip Rev: Comput Stat* 10(3):1421. <https://doi.org/10.1002/wics.1421>
- Jiang C, Zhou Y, Chen B (2023) Mining semantic features in patent text for financial distress prediction. *Technol Forecast Soc Chang* 190:122450
- Juez-Gil M, Arnaiz-Gonzalez A, Rodriguez JJ, Lopez-Nozal C, Garcia-Orsorio C (2021) Approx-smote: fast smote for big data on apache spark. *Neurocomputing* 464:432–437. <https://doi.org/10.1016/j.neucom.2021.08.086>
- Karakoulas G, Shawe-Taylor J (1998) Optimizing classifiers for imbalanced training sets. *Adv Neural Inform Process Syst*, 11
- Kaur H, Pannu HS, Malhi AK (2019) A systematic review on imbalanced data challenges in machine learning: applications and solutions. *ACM Comput Surv (CSUR)* 52(4):1–36. <https://doi.org/10.1145/3343440>
- Khan AA, Chaudhari O, Chandra R (2023) A review of ensemble learning and data augmentation models for class imbalanced problems: combination, implementation and evaluation. *Expert Syst Appl*, 122778
- Kim SY, Upneja A (2021) Majority voting ensemble with a decision trees for business failure prediction during economic downturns. *J Innov Knowl* 6(2):112–123. <https://doi.org/10.1016/j.jik.2021.01.001>
- Kuiziniienė D, Krilavičius T, Damaševičius R, Maskeliūnas R (2022) Systematic review of financial distress identification using artificial intelligence methods. *Appl Artif Intell* 36(1):2138124
- Kuiziniienė D, Krilavičius T, Damaševičius R, Maskeliūnas R (2022) Systematic review of financial distress identification using artificial intelligence methods. *Appl Artif Intell* 36(1):2138124. <https://doi.org/10.1080/08839514.2022.2138124>
- Lazarevic A, Kumar V (2005) Feature bagging for outlier detection. In: Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining. KDD '05. Association for Computing Machinery, New York, NY, USA, pp 157–166. <https://doi.org/10.1145/1081870.1081891>
- Le T (2022) A comprehensive survey of imbalanced learning methods for bankruptcy prediction. *IET Commun* 16(5):433–441
- Leskovec J, Shawe-Taylor J (2003) Linear programming boosting for uneven datasets. In: Proceedings of the 20th international conference on machine learning (ICML-03), pp 456–463
- Liang D, Tsai C-F, Dai A-J, Eberle W (2018) A novel classifier ensemble approach for financial distress prediction. *Knowl Inf Syst* 54:437–462. <https://doi.org/10.1007/s10115-017-1061-1>
- Liang D, Tsai C-F (2020) UCI machine learning repository - Taiwanese bankruptcy prediction data set. <https://archive.ics.uci.edu/ml/datasets/Taiwanese+Bankruptcy+Prediction>
- Lin W-Y, Hu Y-H, Tsai C-F (2011) Machine learning in financial crisis prediction: a survey. *IEEE Trans Syst, Man, Cybern, Part C (Appl Rev)* 42(4):421–436
- Lin W-C, Lu Y-H, Tsai C-F (2019) Feature selection in single and ensemble learning-based bankruptcy prediction models. *Expert Syst* 36(1):12335. <https://doi.org/10.1111/exsy.12335>
- Lin C, Tsai C-F, Lin W-C (2023) Towards hybrid over-and under-sampling combination methods for class imbalanced datasets: an experimental study. *Artif Intell Rev* 56(2):845–863. <https://doi.org/10.1007/s10462-022-10186-5>
- Liu FT, Ting KM, Zhou Z-H (2008) Isolation forest. In: 2008 Eighth IEEE international conference on data mining. IEEE, pp 413–422. <https://doi.org/10.1109/ICDM.2008.17>
- Liu X-Y, Wu J, Zhou Z-H (2008) Exploratory undersampling for class-imbalance learning. *IEEE Trans Syst, Man, Cybern, Part B (Cybern)* 39(2):539–550. <https://doi.org/10.1109/TSMCB.2008.2007853>
- Liu Z, Cao W, Gao Z, Bian J, Chen H, Chang Y, Liu T-Y (2020) Self-paced ensemble for highly imbalanced massive data classification. In: 2020 IEEE 36th international conference on data engineering (ICDE). IEEE, pp 841–852. <https://doi.org/10.1109/ICDE48307.2020.00078>
- Li Z, Zhao Y, Botta N, Ionescu C, Hu X (2020) Copod: copula-based outlier detection. In: 2020 IEEE international conference on data mining (ICDM). IEEE, pp 1118–1123. <https://doi.org/10.1109/ICDM50108.2020.00135>

- Loukas L, Stogiannidis I, Malakasiotis P, Vassos S (2023) Breaking the bank with chatgpt: Few-shot text classification for finance. arXiv preprint [arXiv:2308.14634](https://arxiv.org/abs/2308.14634)
- Mani I, Zhang I (2003) knn approach to unbalanced data distributions: a case study involving information extraction. In: Proceedings of workshop on learning from imbalanced datasets. ICML, 126, pp 1–7
- Memic N, Memic D (2020) Financial traits of bankruptcy, empirical evidence from Bosnia and Herzegovina. *Int J Bus Perform Manag* 21(1–2):76–94. <https://doi.org/10.1504/IJBPM.2020.106111>
- Muslim MA, Nikmah TL, Pertiwi DAA, Dasril Y et al (2023) New model combination meta-learner to improve accuracy prediction p2p lending with stacking ensemble learning. *Intell Syst Appl* 18:200204. <https://doi.org/10.1016/j.iswa.2023.200204>
- Nazareth N, Ramana Reddy YV (2023) Financial applications of machine learning: a literature review. *Expert Syst Appl* 219:119640. <https://doi.org/10.1016/j.eswa.2023.119640>
- Niu Z, Guo W, Xue J, Wang Y, Kong Z, Huang L (2023) A novel anomaly detection approach based on ensemble semi-supervised active learning (Adessa). *Comput Secur* 129:103190. <https://doi.org/10.1016/j.cose.2023.103190>
- Opitz DW, Maclin RF (1997) An empirical evaluation of bagging and boosting for artificial neural networks. In: Proceedings of international conference on neural networks (ICNN'97). IEEE, 3, pp 1401–1405. <https://doi.org/10.1109/ICNN.1997.613999>
- Ouenniche J, Pérez-Gladish B, Bouslah K (2018) An out-of-sample framework for topsi-based classifiers with application in bankruptcy prediction. *Technol Forecast Soc Change* 131:111–116
- Pamula R, Deka JK, Nandi S (2011) An outlier detection method based on clustering. In: 2011 Second international conference on emerging applications of information technology, pp 253–256. <https://doi.org/10.1109/EAIT.2011.25>
- Park K (2018) Financial reporting quality and corporate innovation. *J Bus Finance Account* 45(7–8):871–894
- Pearson, K (1901) Li. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science* 2(11):559–572 <https://doi.org/10.1080/14786440109462720>
- Pevný T (2016) Loda: Lightweight on-line detector of anomalies. *Mach Learn* 102:275–304. <https://doi.org/10.1007/s10994-015-5521-0>
- Qu Y, Quan P, Lei M, Shi Y (2019) Review of bankruptcy prediction using machine learning and deep learning techniques. *Procedia Comput Sci* 162:895–899. <https://doi.org/10.1016/j.procs.2019.12.065>
- Quinn JA, Sugiyama M (2014) A least-squares approach to anomaly detection in static and sequential data. *Pattern Recogn Lett* 40:36–40. <https://doi.org/10.1016/j.patrec.2013.12.016>
- Rezvani S, Wang X (2023) A broad review on class imbalance learning techniques. *Appl Soft Comput*, 110415
- Schölkopf B, Williamson RC, Smola A, Shawe-Taylor J, Platt J (1999) Support vector method for novelty detection. *Adv Neural Inform Process Syst*, 12
- Schreurs J, Vranckx I, Hubert M, Suykens J, Rousseeuw P (2021) Outlier detection in non-elliptical data by kernel mrd. *Stat Comput*. <https://doi.org/10.1007/s11222-021-10041-7>
- Seiffert C, Khoshgoftar TM, Van Hulse J, Napolitano A (2009) Rusboost: a hybrid approach to alleviating class imbalance. *IEEE Trans Syst, Man, Cybern-Part A: Syst Humans* 40(1):185–197. <https://doi.org/10.1109/TSMCA.2009.2029559>
- Shakeel MR, Siddiqui TA, Alam S (2023) Feature selection in corporate bankruptcy prediction using ml techniques: a systematic literature review. *Adv Signal Process, Embedded Syst IoT*, pp 345–363
- Shi S, Li J, Zhu D, Yang F, Xu Y (2023) A hybrid imbalanced classification model based on data density. *Inf Sci* 624:50–67. <https://doi.org/10.1016/j.ins.2022.12.046>
- Silva Mattos E, Shasha D (2024) Bankruptcy prediction with low-quality financial information. *Expert Syst Appl* 237:121418
- Sun J, Fujita H, Zheng Y, Ai W (2021) Multi-class financial distress prediction based on support vector machines integrated with the decomposition and fusion methods. *Inf Sci* 559:153–170. <https://doi.org/10.1016/j.ins.2021.01.059>
- Tax DM, Duin RP (2004) Support vector data description. *Mach Learn* 54:45–66. <https://doi.org/10.1023/B:MACH.0000008084.60811.49>
- Tomek I (1976) Two modifications of cnn. *IEEE Trans Syst Man Cybern SMC*–6(11):769–772. <https://doi.org/10.1109/TSMC.1976.4309452>
- Vander Bauwhede H, De Meyere M, Van Cauwenberge P (2015) Financial reporting quality and the cost of debt of smes. *Small Bus Econ* 45:149–164
- Veganzones D (2022) Corporate failure prediction using threshold-based models. *J Forecast* 41(5):956–979. <https://doi.org/10.1002/for.2842>
- Wang H, Liu X (2021) Undersampling bankruptcy prediction: Taiwan bankruptcy data. *PLoS ONE* 16(7):0254030. <https://doi.org/10.1371/journal.pone.0254030>

- Wang G, Ma J (2012) A hybrid ensemble approach for enterprise credit risk assessment based on support vector machine. *Expert Syst Appl* 39(5):5325–5331
- Wu R, Keogh EJ (2023) Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. *IEEE Trans Knowl Data Eng* 35(3):2421–2429. <https://doi.org/10.1109/TKDE.2021.3112126>
- Yang P, Huang B (2008) Knn based outlier detection algorithm in large dataset. In: 2008 International workshop on education technology and training & 2008 international workshop on geoscience and remote sensing. IEEE, 1, pp 611–613. <https://doi.org/10.1109/ETTandGRS.2008.306>
- Yang X, Song Q, Cao A (2005) Weighted support vector machine for data classification. In: Proceedings. 2005 IEEE international joint conference on neural networks, 2, pp 859–8642. <https://doi.org/10.1109/IJCNN.2005.1555965>
- Zelenkov Y, Volodarskiy N (2021) Bankruptcy prediction on the base of the unbalanced data using multi-objective selection of classifiers. *Expert Syst Appl* 185:115559. <https://doi.org/10.1016/j.eswa.2021.115559>
- Zhao Y, Hryniewicki MK (2018) Xgbod: improving supervised outlier detection with unsupervised representation learning. In: 2018 International joint conference on neural networks (IJCNN). IEEE, pp 1–8. <https://doi.org/10.1109/IJCNN.2018.8489605>
- Zheng Y, Xu Z, Xiao A (2023) Deep learning in economics: a systematic and critical review. *Artif Intell Rev*. <https://doi.org/10.1007/s10462-022-10272-8>
- Zięba M, Tomczak SK, Tomczak JM (2016) Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction. *Expert Syst Appl* 58:93–101. <https://doi.org/10.1016/j.eswa.2016.04.001>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.