Journal of Economics, Finance and Accounting Studies (JEFAS)
ISSN: 2709-0809
DOI: 10.32996/jefas
Journal Homepage: www.al-kindipublisher.com/index.php/jefas

Predicting Bank Failures with Machine Learning Algorithms: A Comparison of Boosting and Cost-Sensitive Models

Safa SEN[1] ✉ **and Sara Almeida de Figueiredo**[2]
[12]Ph.D. Student at The University of Miskolc, Miskolc, Hungary
✉ **Corresponding Author**: Safa SEN, **E-mail**: safasen2@gmail.com

ARTICLE INFORMATION

Received: August 11, 2020
Accepted: September 01, 2021
Volume: 3
Issue: 2
DOI: 10.32996/jefas.2021.3.2.5

KEYWORDS

Financial crisis, Bank failure, CS-Forest, GLMBoost, XGBoost, Gradient Boosting

JEL Classification C11 · C13 · E37 · E44

ABSTRACT

Predicting bank failures has been an essential subject in literature due to the significance of the banks for the economic prosperity of a country. Acting as an intermediary player of the economy, banks channel funds between creditors and debtors. In that matter, banks are considered the backbone of the economies; hence, it is important to create early warning systems that identify insolvent banks from solvent ones. Thus, Insolvent banks can apply for assistance and avoid bankruptcy in financially turbulent times. In this paper, we will focus on two different machine learning disciplines: Boosting and Cost-Sensitive methods to predict bank failures. Boosting methods are widely used in the literature due to their better prediction capability. However, Cost-Sensitive Forest is relatively new to the literature and originally invented to solve imbalance problems in software defect detection. Our results show that comparing to the boosting methods, Cost-Sensitive Forest particularly classifies failed banks more accurately. Thus, we suggest using the Cost-Sensitive Forest when predicting bank failures with imbalanced datasets.

1. Introduction

Banks are one of the most important key players in a society where they act as an intermediary and channel funds from creditors to debtors efficiently to sustain economic stability. A well-structured financial system is essential for a modern economy and the performance of the banks has a significant impact on society. Therefore, understanding and analyzing bank failures are essential parts of the literature. The study of bank failure is important for two reasons. First, understanding the factors related to a bank's failure enables regulatory authorities to manage and supervise banks more efficiently. Second, the ability to differentiate between sound banks and troubled ones will reduce the expected cost of bank failure. In other words, if examiners can detect problems early enough, regulatory actions can be taken either to prevent a bank from failing or minimize the costs to the public and thus taxpayers (Thomson, 1991).

In early literature, Meyer and Pifer (1970); Sinkey (1975); Martin (1977); Kolari et al. (2002) used statistical models such as linear, quadratic discriminant analysis, factor analysis, and logistic regression methods in the early literature. Later on, Ravi and Promodh (2008), Boyacıoglu et al. (2009), and Erdal and Ekinci (2013) used artificial neural networks, operations research, hybrid intelligent methods, fuzzy logic, and support vector machines, due to their better prediction capabilities for bank failures.

This paper will use machine learning models to predict bank failures in Taiwan between 1999 and 2009. We will implement GLMBoost, XGBoost, and Cost-Sensitive Forest models to feed bank ratios into each model. XGBoost and GLMBoost are both gradient boosting methods, while CS-Forest is a Cost-Sensitive algorithm used to solve the imbalance dataset problems.

This paper is organized as follows:

The first section gives a general outlook on the banking sector and the importance of banks, introductory literature on how bank failure predictions have evolved, and brief information on our datasets and models in our article.

The second section gives a comprehensive list of previous studies related to bank or company failure prediction and several machine learning algorithms and their performance.

The third section initially gives a general description of the data that we used in the research. Afterwards, we defined the methods and performance metrics used in this study and most of the terms that can be found in the results section in our paper.

In the fourth and last section, all the results were displayed together with the selected indicators, which can be found in Table II and their confusion matrix in Table III where we discussed the model performances and our methodology to demonstrate our results.

2.Literature Review

Cleary and Hebb (2015) analyzed the failures of 132 banks for the period of 2002-2009 by using discriminant analysis. The prediction efficiency of bank failure was 92% by the sample data. Furthermore, they did the same analysis to predict bank failures between 2010-2011 and the efficiency range was between 90-95%.

Chiaramonte et al. (2016) investigated US commercial bank data for 2004 to 2012 to analyze how Z Score can forecast bank failure. The outcome of the investigation showed that Z-Score could forecast 76% of bank failures. Important to state that macro-level indicators did not increase the precision of the forecast. On the other hand, the forecast efficiency of the Z-score to predict bank default remains stable within the three-year forward window.

Ekinci and Erdal (2016) analysed the bank failure prediction of 37 commercial banks in Turkey between 1997 and 2001. In the dataset 20 banks were healthy and 17 of them were failed. They used Logistic Regression, J48, and voted perceptron as the base learners along with different Hybrid Ensembles. Their empirical findings indicated that hybrid ensemble machine learning models perform better than the traditional base and ensemble models.

Le and Viviani (2017) analyzed the failure of banks using both traditional techniques and machine learning for the sample of 3000 US banks, where 1438 of them were failures and 1562 of them were active banks. They used discriminant analysis and Logistic regression for traditional techniques and, as for machine learning, they used artificial neural networks, support vector machines, and k-nearest neighbours. CAMEL ratios were used in the analysis and 31 financial ratios represented them. The empirical findings state that the artificial neural network and k-nearest neighbor methods are the most precise ones to predict bank failures.

Gogas et al. (2018) implemented machine learning models to forecast bank failures. Their data consist of a total of 1443 U.S. banks with 481 banks that have failed between 2007–2013. They implemented a two-step feature selection procedure to define the most informative variables. Afterwards, selected variables were put into an SVM model to proceed with a training-testing learning process. The model showed a 99.22% forecasting accuracy and outperformed the well-established Ohlson’s score.

In their article, Carmona et al. (2019) implemented Extreme Gradient Boosting Method to forecast bank failure of 157 US national commercial banks from 2001 to 2015. They considered 30 financial ratios in their model. Their results state that lower values for retained earnings to average equity, pretax return on assets, and total risk-based capital ratio are linked with bank failure. Additionally, they suggest that retained earnings should be kept within the company in stressful periods and that dividend policies should be reconsidered.

Manthoulis et al. (2020) used statistical and machine learning methods to a sample consisting of 60,000 observations between 2006 and 2015 for the US banks. Their findings showed that diversification variables improve the prediction strength of bank failure forecasting models, mostly for mid to long-term forecast horizons. Furthermore, they indicated that ordinal classification models provide a better description of the state of the banks before failure and are competitive to standard binary classification models.

Momparler et al. (2020) used the dataset of 157 US national commercial banks from 2001 to 2015, considering 30 financial ratios in their model. They implemented the fuzzy-set qualitative comparative analysis to identify the factors that lead a bank to failure. Results showed that non-performing loans account for a large share of banks’ balance sheets and the levels of risk coverage and capitalization are low, which would mean that the chances of bank failure are high.

Petropoulos et al. (2020) used various modeling techniques to forecast bank insolvencies. They used the data from the US financial institutions with a time frame between 2008-2014. CAMELS were analyzed as the main data. Their results show that Random Forests consistently performs better than a series of other models, such as Logistic Regression, Linear Discriminant Analysis, Support Vector Machines, and Neural Networks.

Zizi et al. (2020) used logistic regression to determine the reasons for the financial failures of SMEs. They retrieved the data of healthy and failed companies from Moroccan Bank. Their findings state that the Autonomy ratio, interest to sales, asset turnover, days in accounts receivable, and duration of trade payables increase the probability of financial failure, while repayment capacity and return on assets reduce the probability of failure. On the other hand, given variables show an overall classification rate of healthy and failing SMEs of 91.11% three years before failure and 84.44% two years and one year before failure.

Abidin et al. (2020) compare the power of logit and artificial neural network (ANN) models in predicting the failure of SMEs in the hospitality industry. They used data consisting of 41 failed and 41 non-failed SMEs in the hospitality industry between 2000 and 2016. They used Stepwise logit regression and multilayer perceptron. Their findings stated that the multilayer perceptron model consistently outperformed the logit model in classifying the failed and non-failed SMEs in the hospitality industry.

3. Data and Methods

3.1. Data

The data was collected from the Taiwan Economic Journal for the years 1999 to 2009. Company bankruptcy was defined based on the business regulations of the Taiwan Stock Exchange. In total, there are 1000 banks in which 780 are healthy and 220 are failed banks. Additionally,ratios used for this research can be found in Appendix I.

3.2. Methods

In this experiment, we used three different machine learning algorithms to classify bank failures in Taiwan. We used two boosting and one cost-sensitive algorithms. For boosting, we chose XGBoost and GLMBoost and for a cost-sensitive model, we used CS-Forest.

We used WEKA 3.9.5 (Waikato Environment for Knowledge Analysis) for the experiments for the data processing. All the models we used included as a package of WEKA; however, for XGBoost, we used the R extension of the WEKA where analysis could be done via WEKA,but algorithms will work under R Console. Therefore, it requires an R programming language installed in the computer to proceed with the analysis.

Gradient Boosting

Guelman (2012) defines gradient boosting as an iterative algorithm that combines parameterized functions with poor performance to produce more accurate forecasting rules. While other statistical learning methods provide comparable accuracy, such as support vector machines and neural networks, gradient boosting brings more interpretable results. The method is highly robust and can be applied for both classification and regression problems with a variety of response distributions ,for instance, Gaussian, Poisson, and Laplace.

GLMBoost

Adamu et al. (2019) defined Generalized Linear Models as a generalization of linear regression that allows normally distributed response variables and non-normally distributed variables. GLM is a broad class of models for relating responses to linear combinations of predictive variables. Fitting generalized models using the gradient boosting algorithm is called GLMBoost. GLMBoost can be used to fit linear models through component-wise reinforcement, and each column of the design matrix is individually fitted and selected using a simple linear model. Therefore, the algorithm is a gradient booster to optimize certain loss functions using linear models on a per-component basis. Using GLMBoost for any linear model, the results are more accurate and reliable than the corresponding GLM function.

XG Boost

It is an application of gradient boosting machines created by Tianqi Chen in 2014. Designed for efficiency and scalability, its parallel tree reinforcement capabilities make it significantly faster than other tree-based ensemble algorithms. (Quinto,2020)

Analytics Vidhya Content Team (2018) defines the features of XGBoost as below:

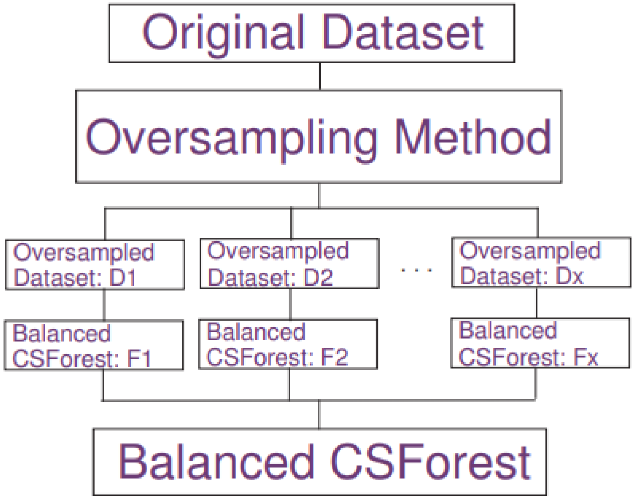
- The model has an option to penalize complex models through both L1 and L2 regularization. Therefore, it prevents overfitting.
- XGBoost incorporates a sparsity-aware split finding algorithm to handle different types of sparsity patterns in the data.
- XGBoost has a distributed weighted quantile sketch algorithm to handle weighted data effectively.

Cost-Sensitive Forest

Siers and Islam (2015) created Cost-Sensitive Forests to prevent imbalance problems in software defect prediction. The main argument in their model was datasets used contained many more non-defective examples than defective examples. This poses a problem to data mining algorithms since there exists an under-representation of the defective examples and an over-representation of the non-defective examples. The data miner can employ resampling techniques in order to account for class imbalance. Resampling techniques involve altering the examples in the dataset. This can be achieved by using either under-sampling or oversampling techniques. However, this method can easily be extended for multi-class cost-sensitive classifications we used for Bank Failure since the nature of bankruptcy datasets is very similar. In most cases, there are always more healthy banks than failure banks and this creates an imbalance problem for bank failure prediction.

In order to incorporate oversampling techniques, they use SMOTE and Safe-Level Smote methods to create new versions of the original dataset. These data sets can be seen in figure X as D1, D2, and Dx. From each oversampled data set, they build a CS-Forest; F1, F2, and F(x). Once new CS-Forests have been built, the trees from each CS-Forest are collected together into a single CS-Forest.

Figure I: Incorporating Oversampling into CS-Forest:



Source: Siers and Islam (2015)

Confusion Matrix: Miao and Zhu (2020) mentioned that the performance of the classification problems is typically measured with a confusion matrix generated by the associated classifier.

Table I: Confusion Matrix

	Predicted as Positive	Predicted as Negative
Actually Positive	True Positive	False Negative
Actually Negative	False Positive	True Negative

Source: Miao and Zhu (2020)

In the confusion matrix, True Positive and True Negative are correctly classified values. On the other hand, False Negative means that the value was originally positive, however, it is predicted as negative, while False Positive is when a value is originally True Negative but it is predicted as Positive.

Recall Rate

Brownlee (2020) defines Recall as a metric that quantifies the number of correct positive predictions made out of all correct positive predictions that could have been made. Unlike precision that only comments on the correct positive predictions out of all positive predictions, recall indicates missed positive predictions. In this way, recall provides some notion of the coverage of the positive class. For imbalanced learning, recall is typically used to measure the coverage of the minority class.

Formulation: Recall = True Positive /True Positive + False Negative

AUC Term
Hanley and McNeil (1982) defined AUC as a single scalar value that measures the overall performance of a binary classifier. The area under a receiver operating characteristic (ROC) curve is abbreviated as AUC.

Hosmer and Lemeshow (2000) ranked ROC values as:

If ROC = 0.5: this suggests no discrimination (same as flipping a coin).

If 0.7 ≤ ROC < 0.8: This is considered acceptable discrimination.

If 0.8 ≤ ROC < 0.9: This considered excellent discrimination.

If ROC ≥ 0.9: This is considered outstanding discrimination.

K-fold cross-validation

In this experiment, we used 10-fold cross-validation to measure model performance while it minimizes bias associated with a random sampling of training (Chou and Pham (2013)). Yumurtaci et al. (2015) define the 10-fold CV as the dataset is randomly split into ten (k) subsets of the same size in which the class is represented in approximately the same proportions as in the full dataset. Next, each subset is held out in turn and the learning scheme is trained on the remaining nine-tenths (k-1); then, its error rate is calculated on the holdout set. Thus, the learning procedure has been executed a total of ten times on different training sets.

4. Results

Table II: Results of Indicators Selected

Model	Classification Rate (%)	AUC (%)	Recall (Failed Banks) (%)
GLMBoost:	0.87	0.92	0.56
XGBoost:	0.92	0.93	0.71
CS Forest:	0.91	0.97	0.89

Source: Authors own elaboration.

We analyzed three classification models: GLMBoost, XGBoost, and CS-Forest. Our ranking methodology when considering the success of the models is based on three metrics: Classification Rate, Recall, and AUC results. The classification rate stands for the number of correctly classified instances (Both true positive and true negative) in the model, as shown in Table IV. AUC (ROC curve) is a commonly used metric for classification problems. True positive responses converge AUC closer to 1 while false-positive results diverge it to 0.5, bringing the same results with tossing a coin. Finally, Recall Rate measures the accurately predicted instances of the Failed Banks. Therefore, it needs to be taken into consideration when making a comparison between models while the main purpose is to increase the Recall rate as high as possible to reduce the false-positive rate. Best classification rate provided by XGBoost, followed by CS-Forest and GLMBoost as the least successful classifier. On the other hand, CS-Forest performed the best AUC, followed by XGBoost and GLMBoost as the least successful performer.

Finally, CS-Forest once again performed the best results of Recall Rate, followed by XGBoost and GLMBoost as the least performer.

Table III: Results in Confusion Matrix

	CS FOREST		XGBoost		GLMBoost	
	Predicted as non-Failed	Predicted as Failed	Predicted as non-Failed	Predicted as Failed	Predicted as non-Failed	Predicted as Failed
A	712	68	765	15	749	31
B	25	195	63	157	97	123
			A=non-Failed	B=Failed		

CS-Forest correctly classified 712 Non-Failed Bank while it misclassified 68 healthy banks as failed. On the other hand, it correctly classified 195 Failed banks while it misclassified 25 failed banks as healthy.

XGBoost correctly classified 765 Non-Failed Bank while it misclassified 15 healthy banks as failed. On the other hand, it correctly classified 157 Failed banks while it misclassified 63 failed banks as healthy.

GLMBoost correctly classified 749 Non-Failed Bank while it misclassified 31 healthy banks as failed. On the other hand, it correctly classified 123 Failed banks while it misclassified 93 failed banks as healthy.

We can conclude from the above-mentioned results that there is a strong competition between XGBoost and CS-Forest in terms of prediction capability. XGBoost provides the best results in predicting non-Failed banks while CS-Forest provides the best results in predicting failed banks with a strong distinction. It correctly classifies 38 failed banks better than XGBoost and 72 failed banks better than GLMBoost. It is important to state that most of the time purpose of related researches is to misclassify the failed banks the least. Therefore, even though XGBoost provides very strong results in healthy banks, it doesn't particularly focus on reducing Type II error. Thus, we highly suggest using CS-Forest when there are imbalanced datasets for bank failure prediction. Despite the excellent performance of XGBoost, it cannot solve imbalance problems as CS-Forest does.

5. Conclusion

Prediction of bank failures has been a quite interesting study due to the nature of risky behavior of the financial sector, especially in turbulent times. Developing technologies and algorithms help us to advance in our predictions and increase the accuracy of our experiments. In our experiment, we used the Taiwanese bank failure dataset to assess the up-to-date models' performance in predicting capability. Most successful bank failure models are currently based on grading boosting methods due to their advancement in understanding the dataset and predicting better outputs. However, in our article, we used a new technique to predict bank failure and compare it with the best methods in the literature. Cost-Sensitive Forest was created to predict software defects with datasets with imbalance problems. We implemented the same model in bank failure prediction due to similar dataset behaviour in datasets while bank failure datasets mostly include non-failure banks than failed banks and algorithms mostly confused with this matter therefore, they might perform less than they could actually do. Our results show that even though total classification performance is somewhat similar, predicting the Recall rate for failed banks with CS-Forest is much higher than GLMBoost and XGBoost. Thus, we suggest researchers consider the CS-Forest model when dealing with bank failure prediction with imbalanced datasets.

Appendix I:

Ratios used in the research
ROA (Before interest and depreciation before interest.)
Cash flow rate.
Net Value Per Share.
Persistent EPS in the Last Four Seasons.
Cash Flow Per Share.
Revenue Per Share in Yuan.
Cash Reinvestment in PCT.
Interest Expense Ratio.
Debt ratio in PCT.
Net worth To Assets.
Borrowing dependency.
Operating profit To Paid-in capital.
Total Asset Turnover.
Operating Funds to Liability.

Working Capital to Equity.
Retained Earnings to Total Assets.
Cash Flow to Sales.
Fixed Assets to Assets.
Current Liability to Equity.
Equity to Long-term Liability.
Cash Flow to Equity.
Current Liability to Current Assets.
Net Income to Total Assets.
Gross Profit to Sales.
Net Income to Stockholder Equity.
Liability to Equity.
Degree of Financial Leverage.
Interest Coverage Ratio.

References

[1] Abidin, J. - Abdullah, N. - Khaw, K. (2020), *Predicting SMEs Failure: Logistic Regression vs Artificial Neural Network Models*. Capital Markets Review, 28: 29-41.

[2] Adamu, H.A. - Muhammad, M. - Jingi, A.M. (2019): *Application of Gradient Boosting Algorithm In Statistical Modelling*. Journal of Statistics and Mathematical Sciences. 5: 11-18.

[3] Analytics V. Team. (2020, December 23). XGBoost Algorithm: XGBoost In Machine Learning. https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/.

[4] Bank Failure Dataset: https://www.kaggle.com/fedesoriano/company-bankruptcy-prediction

[5] Carmona, P. - Climent, F. - Momparler, A. (2019): *Predicting bank failure in the U.S. banking sector: An extreme gradient boosting approach*. *International Review of Economics & Finance*. 61: 304-323. DOI: 10.1016/j.iref.2018.03.008.

[6] Chiaramonte, L. - Liu, F. -Federica, P. - Zhou, M. (2016): *How Accurately Can Z-score Predict Bank Failure? Financial Markets, Institutions & Instruments*. 25 : 333-360. DOI : 10.1111/fmii.12077.

[7] Chou, J. S. - Pham, A. D. (2013) : Enhanced artificial intelligence for ensemble approach to predicting high performance concrete compressive strength. *Construction and Building Materials*, 49: 554–563.

[8] Cleary, S. - Hebb, G. (2015): *An efficient and functional model for predicting bank distress: In and out of sample evidence*. *Journal of Banking & Finance*. 64. DOI: 10.1016/j.jbankfin.2015.12.001.

[9] Ekinci, A. - Erdal, H. (2016): Forecasting Bank Failure: Base Learners, Ensembles and Hybrid Ensembles. *Computational Economics*. 49. DOI: 10.1007/s10614-016-9623-y.

[10] Erdal, H. I., & Ekinci, A. (2013): A comparison of various artificial intelligence methods in the prediction of bank failures. *Computational Economics*, 42(2): 199–215.

[11] Gogas, P. - Papadimitriou, T. - Agrapetidou, A. (2018): Forecasting bank failures and stress testing: A machine learning approach. International Journal of Forecasting. 34: 440-455. DOI: 10.1016/j.ijforecast.2018.01.009.

[12] Guelman, L. (2012): Gradient boosting trees for auto insurance loss cost modeling and prediction. *Expert Syst. Appl.*. 39: 3659-3667. DOI: 10.1016/j.eswa.2011.09.058.

[13] Hanley, J.A. - Mcneil, B. (1982): The Meaning and Use of the Area Under a Receiver Operating Characteristic (ROC) Curve. *Radiology*. 143: 29-36. DOI: 10.1148/radiology.143.1.7063747.

[14] Hosmer, S Lemeshow -Applied Logistic Regression, *John Wiley and Sons, New York, NY* (2000)

[15] J. Brownlee, Imbalanced classification with python: Better metrics, balance skewed classes, cost-sensitive learning, Machine Learning Mastery, 2020.

[16] Kolari, J. - Glennon, D. - Shin, H. - Caputo, M. (2002): Predicting large US commercial bank failures*. Journal of Economics and Business*, 54(4): 361–387.

[17] Le, H. - Viviani, J. L. (2017): Predicting bank failure: an improvement by implementing Machine learning approach on classical financial ratios. *Research in International Business and Finance*. 44. DOI: 10.1016/j.ribaf.2017.07.104.

[18] Manthoulis, G. - Doumpos, M. - Zopounidis, C. - Galariotis, E. (2020): An Ordinal Classification Framework for Bank Failure Prediction: Methodology and Empirical Evidence for US Banks. *European Journal of Operational Research*. 282: 786-801. DOI: 10.1016/j.ejor.2019.09.040.

[19] Martin, D. (1977): Early warning of bank failure: A logit regression approach. *Journal of Banking & Finance*, 1 (3): 249-276, https://EconPapers.repec.org/RePEc:eee:jbfina:v:1:y:1977:i:3:p:249-276.

[20] Meyer, P. A., & Pifer, H. W. (1970): Prediction of bank failures. *The Journal of Finance*, 25(4): 853–858.

[21] Miao, J., & Zhu, W. (2020). Precision-Recall Curve (PRC) Classification Trees. ArXiv, abs/2011.07640.

[22] Momparler, A. - Carmona, P. - Climent, F. (2020): Revisiting bank failure in the United States: a fuzzy-set analysis. *Economic Research-Ekonomska Istraživanja*. 33: 1-17. DOI: 10.1080/1331677X.2019.1689838.

[23] Petropoulos, A. - Siakoulis, V. - Stavroulakis, E. - Vlachogiannakis, N. E. (2020): Predicting bank insolvencies using machine learning techniques. *International Journal of Forecasting*, Elsevier, 36(3): 1092-1113. DOI: 10.1016/j.ijforecast.2019.11.005

[24] Quinto B. (2020): *Supervised Learning*. In: Next-Generation Machine Learning with Spark. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-5669-5_3

[25] Thomson, J.B., 1991. Predicting bank failure in the 1980s. *Economic Review* 27.

[26] Ravi, V. - Promodh, C. (2008): Threshold accepting trained principal component neural network and feature subset selection: Application to bankruptcy prediction in banks. *Applied Soft Computing*, 8(4): 1539–1548.

[27] Siers, M. & Islam, M. (2015). Software defect prediction using a cost sensitive decision forest and voting, and a potential solution to the class imbalance problem. *Information Systems*. 51. 62-71. 10.1016/j.is.2015.02.006.

[28] Sinkey,J. F. (1975): A Multivariate Statistical Analysis of the Characteristics of Problem Banks*, Journal of Finance*, 30 (1): 21-36, https://EconPapers.repec.org/RePEc:bla:jfinan:v:30:y:1975:i:1:p:21-36.

[29] Yumurtaci-Aydogmus, H. - Erdal, H. I. - Karakurt, O. - Namli, E. - Turkan, Y. S. - Erdal, H. (2015): A comparative assessment of bagging ensemble models for modeling concrete slump flow. *Computers and Concrete*, 16(2): 741–757.

[30] Zizi, Y. - Mohamed, O. - Moudden, A. (2020): Determinants and Predictors of SMEs' Financial Failure: A Logistic Regression Approach. *Risks*. 8. DOI: 10.3390/risks8040107.