Computational Economics (2024) 64:3161–3179
https://doi.org/10.1007/s10614-023-10537-6

Forecasting Bank Failure in the U.S.: A Cost-Sensitive Approach

Aykut Ekinci[1] · **Safa Sen**[2]

Accepted: 11 December 2023 / Published online: 19 February 2024

Abstract

Preventing bank failure has been a top priority among regulatory institutions and policymakers driven by a robust theoretical and empirical foundation highlighting the adverse correlation between bank failures and real output. Therefore, the importance of creating early signals is an essential task to undertake to prevent bank failures. We used J48, Logistic Regression, Multilayer Perceptron, Random Forest, Extreme Gradient Boosting (XGBoost), and Cost-Sensitive Forest (CSForest) to predict bank failures in the U.S. for 1482 (59 failed) national banks between 2008 to 2010 during the global financial crisis and its aftermath. This research paper stands as a prominent contribution within the existing literature, employing contemporary machine learning algorithms, namely XGBoost and CSForest. Distinguished by its emphasis on mitigating Type-II errors, CSForest, a novel algorithm introduced in this study, exhibits superior performance in minimizing such errors, while XGBoost performed as one of the weakest among the peers. The empirical findings reveal that Logistic Regression maintains its relevance and efficacy, thus underscoring its continued importance as a benchmark model.

Keywords Machine learning models · Banking failure · Off-site monitoring · CSForest · XGBoost

JEL Classification C45 · C53 · G12 · G17

Abbreviations

CAMELS Capital, Asset Quality, Management, Earnings, Liquidity, Sensitivity
CSForest Cost-sensitive forest

✉ Aykut Ekinci
aykut.ekinci@samsun.edu.tr

Safa Sen
safasen2@gmail.com

[1] Department of Economics and Finance, Samsun University, Samsun, Turkey

[2] Department of Accounting and Finance, University of Miskolc, Miskolc, Hungary

FPR	False-positive rate
fsQCA	Fuzzy-set qualitative comparative analysis
MLP	Multi-layer perceptron
RF	Random forest
SDP	Software defect prediction
TPR	True positive rate
WEKA	Waikato Environment for Knowledge Analysis
XGBoost	Extreme Gradient Boosting

1 Introduction

Banks have a vital role in a well-functioning economic system. Many studies in the literature show that bank failures amplify economic downturns. Friedman and Schwartz (1963) illustrated that during the Great Depression, bank collapses further depressed the economy in the United States by contracting the money supply and elevating credit intermediation costs (Bernanke, 1983). Other studies reinforced these findings by providing evidence that unhealthy financial institutions could precipitate financial instability, leading to significant output losses and even triggering a global recession as observed during the Global Financial Crisis (GFC) (Bernanke, Gepositing that the fartler, and Gilchrist 1996; Kang & Stulz, 2000; Hoggarth et al., 2002; Ashcraft, 2005; Anari et al. 2005; Boyd et al., 2005; Kupiec & Ramirez, 2013). These studies were foundational in shaping the "too big to fail" (TBTF) doctrine during the GFC, positing that the fallout of one financial institution's failure can have far-reaching implications on the entire financial ecosystem.

Ramirez and Shively (2012) used the term "bank failure channel" and highlighted four channels to explain the theoretical relationship between bank failures and real economic activity. (i) Direct wealth effect: the loss of uninsured deposits due to a bank failure turns to a contraction in money supply and consumer spending (Calomiris, 1993; Friedman & Schwartz, 1963). (ii) Illiquidity of deposits: spending is adversely affected even if the uninsured depositors do not lose their money during the process of liquidation (Rockoff, 1993, Anari et al., 2005). (iii) Disruption of relationships: a bank failure reduces the effectiveness of the financial system hence increasing the real cost of intermediation and amplifying the length and depths of depression (Ashcraft, 2003; Bernanke, 1983). (iv) A credit crunch: a bank failure increases the economic uncertainty and leads to a contraction of loan supply of other banks (Calomiris & Mason, 2003, Bernanke, 1983).

The theoretical and empirical evidence illustrating the adverse relationship between bank failures and real output has catalyzed prioritizing bankruptcy prevention amongst regulatory institutions and policymakers. As a result, early warning systems for predicting bank failure, underpinned by statistical and time series methodologies, emerged in academic discourse (Martin, 1977; Meyer & Pifer, 1970; Sinkey, 1975; West, 1985). With the advent of advancements in computer processing capabilities and artificial intelligence algorithms, the deployment of machine learning models for predicting banking failure was initiated (Bell, 1997; Ravi et al., 2008; Swicegood & Clark, 2001; Tam, 1991; Tam & Kiang, 1992).

Numerous studies advocate that machine learning models exhibit superior forecasting performance relative to conventional statistical and time series models (Ekinci & Erdal, 2011; Erdal & Ekinci, 2013). Machine learning models present distinct advantages such as the capacity to detect nonlinear relationships, their data-driven nature, the capability to function with extensive data, and obviate the need for assumptions on the distribution of input data. Certain studies have employed recently developed machine learning algorithms such as random forest and XGBoost (Carmona, Climent & Momparler, 2019), or ensemble models (Olmeda & Fernandez, 1997; Ramu & Ravi, 2009; Verikas et al., 2010; Ravi and Promodh 2010; Kima et al. 2010; Paramjeet et al. 2012; Ekinci & Erdal, 2017).

Recent literature predominantly compares machine learning models by altering variables such as attributes, geographic regions, bank classifications, and/or temporal spans (refer to Sect. 2: Related Literature). Certain studies attempt to mitigate bias within the banking sample by narrowing down the bank selection based on asset size, a strategy particularly applicable to nations with substantial numbers of banks, like the United States. For instance, Manthoulis et al. (2020) used a sample of 6,500 banks (including 430 failed banks), whereas Lee and Viviani (2018) utilized a sample of 3,000 US banks (of which 1,438 were failed banks). Similarly, Carmona et al., (2019) operated with 156 U.S. national commercial banks (78 of which were failed banks), maintaining consistency in attributes and periods. A significant portion of these studies emphasize the true positive rate, reporting predictive accuracy rates exceeding 95%. However, this approach of reducing the bank sample and focusing solely on healthy banks or the weighted prediction average may not be feasible for off-site monitoring. Instead, we center our attention on the cost of a Type-II error (i.e., incorrectly predicting a failing bank as non-failing) due to its heightened financial impact on the banking sector compared to a false positive (i.e., a bank being non-failed but predicted as failed).

In this study, the CSForest algorithm developed by Siers and Islam (2015) for software defect prediction (SDP) was applied as a solution for the class imbalance problem in the banking failure/non-failure sample for the first time in the related literature. Furthermore, Extreme Gradient Boosting (XGBoost), one of the newest decision-tree-based ensemble machine learning algorithms, was first used for bank failure classification by Carmona, Climent & Momparler (2019). We also employed J48 as a conventional statistical classifier, logistic regression as a widely used statistical method using the logistic function, multi-layer perceptron (MLP) as the most widely used neural network structure, and random forest (RF) as a bagging-type ensemble classifier. Data were obtained from 1482 national banks (59 failed) operating in the U.S. between 2008 to 2010 during the global financial crisis and its aftermath. 32 Financial ratios are selected according to the CAMELS (Capital, Asset Quality, Management, Earnings, Liquidity, Sensitivity) system as input variables.

This paper is organized as follows: the second section presents the related literature; the third section gives brief information on methods; the fourth section describes the data and experimental settings and presents the model results and discussion. The paper ends with some brief concluding remarks.

2 Related Literature

A considerable amount of recent papers on bank failures in the U.S., particularly around the period of the 2008 Global Financial Crisis and its aftermath, has employed both traditional time series and machine learning models. Lu and Whidbee (2013) applied logistic regression analysis to ascertain the causes of bank failures and assess specific bank-level characteristics. Their dataset, comprised of 6236 U.S. banks (324 of which failed) from 2007 to 2011, suggested an underlying relationship between a bank's financial vulnerability and the probability of failure. In particular, de novo banks and single-bank holding companies showed a higher likelihood of failure, while multibank holding companies demonstrated a lower propensity.

DeYoung and Torna (2013) questioned whether income generated from unconventional banking activities contributed to the failures of U.S. commercial banks during the financial crisis. Using a multi-period logistic regression model on a dataset of 6851 banks from 2008 to 2010—excluding those with assets exceeding 100 billion dollars—the study revealed that pure fee-based nontraditional activities (such as securities brokerage and insurance sales) decreased the probability of a distressed bank failing. Conversely, asset-based nontraditional activities (like venture capital, investment banking, and asset securitization) increased the likelihood of failure.

Berger and Bouwman (2012) sought to determine the effect of bank capital on financial performance during financial crises. Utilizing logit survival and ordinary least squares regression models for the period from Q1 1984 to Q4 2010, their results indicated that robust capital bases lowered the failure probability of small banks and improved the performance of medium and large banks during banking crises.

Through a horse race analysis between the Z-score and CAMELS-related covariates, Chiaramonte et al. (2015) scrutinized the accuracy of the Z Score. Their findings asserted the Z-score's ability to detect distress events matched that of CAMELS throughout the entire period and particularly during the crisis years (2008–2011). However, they also found the Z-score to be more effective for larger and commercial banks with more sophisticated business models.

Cleary and Hebb (2016) employed discriminant analysis to examine the failures of 132 banks from 2002 to 2009. Their model achieved a prediction success rate of 92% for the sample data and continued to perform well in predicting bank failures from 2010 to 2011, with a success rate between 90 and 95%. Chiaramonte et al. (2016) used the Z Score to predict bank failure based on data from US commercial banks between 2004 to 2012. They found that the Z-Score correctly predicted 76% of bank failures and that macro-level indicators did not enhance the level of accuracy.

Le and Viviani (2017) utilized a combination of traditional and machine learning methods to forecast bank failures, using a dataset of 3000 U.S. Banks, including 1438 failures. The study revealed that machine learning algorithms such as artificial neural networks and k-nearest neighbors yielded superior prediction accuracy for bank failures compared to traditional logistic regression and discriminant analysis methods.

Gogas et al. (2018) also employed machine learning models to forecast bank failures. Their dataset, consisting of 1443 U.S. banks (481 of which failed) from 2007–2013, used a two-step feature selection procedure to select the most informative variables, which were then input into an SVM model. The model demonstrated an impressive 99.22% overall forecasting accuracy, outperforming established benchmark scores such as Ohlson's score.

Carmona et al., (2019) applied the XGBoost machine learning algorithm to predict the failure of 157 U.S. national commercial banks between 2001 and 2015, with 30 financial ratios included in the model. Their findings linked lower values for certain ratios with a higher risk of bank failure.

Manthoulis et al. (2020) applied both statistical and machine learning methods to a dataset of approximately 60,000 observations for U.S. banks over the period 2006–2015. Their results suggested that the inclusion of diversification attributes in prediction models improved their predictive power, especially for mid to long-term prediction horizons.

Momparler et al., (2020) utilized a fuzzy-set qualitative comparative analysis (fsQCA) to identify the combinations of factors leading to bank failure. Their analysis revealed that when non-performing loans constitute a large proportion of banks' balance sheets, and the levels of risk coverage (loan loss provisioning) and capitalization are low, the likelihood of bank failure is high.

Petropoulos et al. (2020) used a selection of modeling techniques to predict bank insolvencies in a sample of US-based financial institutions. They found that Random Forests (RF) demonstrated superior out-of-sample and out-of-time predictive performance. The performance of Neural Networks was also noteworthy, demonstrating comparable results to RF in out-of-time samples. These conclusions were drawn by comparing with traditional bank failure models like Logistic, as well as with other advanced machine learning techniques. Further investigation into the CAMELS evaluation framework showed that metrics related to earnings and capital had the most significant marginal contribution to predicting bank failures.

In their comprehensive study, Shrivastava et al. (2020) scrutinize the application of machine learning methodologies in constructing an early warning system for predicting bank failures. Recognizing the pivotal role of banks within the financial system, and their crucial importance to the economy's stability, the authors collected data from public and private sector Indian banks, both failed and survived, during the period 2000–2017. They employed both bank-specific and macroeconomic variables, in addition to market structure variables, to ascertain banks' stress levels. Given the disproportionately low number of bank failures compared to surviving banks in India, they grappled with an imbalanced data set, which is notoriously difficult for many machine learning algorithms to handle. To overcome this hurdle, they introduced a novel approach, the Synthetic Minority Oversampling Technique (SMOTE), to balance the data. Lasso regression was utilized to excise redundant features from the predictive model, while random forest and AdaBoost techniques were employed to mitigate bias and overfitting. These were compared with logistic regression to ascertain the most effective predictive model. The results of this study have broad applications for various stakeholders, including shareholders, lenders, and borrowers, facilitating the measurement of financial stress in banks. This study

offers a methodical approach, from the selection of the most significant indicators of bank failure using lasso regression, and balancing data using SMOTE, to the choice of appropriate machine learning techniques for bank failure prediction.

Agrapetidou et al. (2021) meticulously examined the efficacy and application of an automated machine learning (AutoML) methodology, specifically, Just Add Data (JAD), as a tool for predicting bank failures. The scope of the study encompassed the entirety of U.S. bank failures between 2007 to 2013, supplemented by an equivalent sample of financially sound institutions. JAD's potent feature selection capabilities were underscored through its ability to discern significant forecasters autonomously.

Furthermore, a conservative estimation of performance generalization and confidence intervals was obtained via the deployment of a bootstrapping methodology. The outcomes of the research were encouraging, with the leading model boasting an AUC of 0.985. This suggests that AutoML tools such as JAD possess the potential to not only augment the productivity of financial data analysts but also serve as a bulwark against methodological statistical errors, delivering models on par with their manually analyzed counterparts.

The work by Lagasio et al. (2022) emphasizes the burgeoning interest in the capabilities of Artificial Intelligence, particularly machine learning methods, within the financial sector. Their study involves the application of various machine learning algorithms, including innovative use of a graph neural network—a method previously unexplored within the financial context—to identify the key determinants of bank defaults. To ensure a balanced dataset, they customized a heuristic oversampling method, considering factors such as competition among potential default determinants. Using data from all Euro Area banks between 2018 and 2020, their findings corroborate earlier studies suggesting the superior performance of neural networks over other methodologies and offer valuable insights from both micro- and macro-economic perspectives.

3 Methods

3.1 J 48 Algorithm

As a statistical classifier, J48 is a decision tree depending on the C4.5 algorithm (see Quinlan, 1993). After Wu et al. (2008) selected C4.5 as the best algorithm among the most influential data mining algorithms, J48 became one of the most commonly used tools in data mining to construct binary classifiers. J 48 algorithm builds decision trees from a set of training data based on entropy reduction and information gain. In the decision tree, the internal node denotes a test on an attribute, the branch signifies the outcome of the test, and the leaf node denotes the class label. The path from the roof to the leaf is called classification rules (See Yadav & Chandel, 2015).

In pursuit of the optimal performance of the J48 algorithm, hyperparameter tuning was conducted using GridSearchCV in conjunction with tenfold cross-validation throughout all experiments. The hyperparameters that underwent tuning are as follows: the minimum number of samples required in a leaf node, for which values of 5, 10, 20, 50, and 100 were tested. In addition, an optimal confidence factor was

sought by incrementally exploring values within the range [0.01–1] in steps of 0.01 (see Table 1 for hyperparameters used in the training models).

3.2 Logistic Regression

Logistic regression is a widely used statistical method and uses the logistic function to model a binary dependent variable coded as "0" and "1" or "failure" and "non-failure" as it is applied in this paper. Following Cessie and Houwelingen (1992), we used ridge estimators in logistic regression to improve the parameter estimates and diminish the error made by further predictions. Furthermore, we have two binary classes, i.e. failure and non-failure for "n" instances with "m" attributes; so the parameter matrix B is calculated with the Quasi-Newton Method to search for the optimized values of the attributes.

Hyperparameter tuning was undertaken to ascertain the optimal ridge value, probing the interval [0.1–1.0] in increments of 0.1. Moreover, a search for the maximum number of iterations was conducted within the range [1–100], with a step size of 5 (see Table 1 for hyperparameters).

3.3 Multilayer Perceptron

The Multilayer Perceptron (MLP) constitutes a specific type of feed-forward neural network, leveraging the backpropagation technique for learning and classification purposes. Through the utilization of hidden layers, MLP facilitates a nonlinear mapping between the input and output layers. Notably, Ivakhnenko and Lapa (1966) proffered the first functional learning algorithm for supervised deep feed-forward multilayer perceptrons, characterized by units that feature polynomial activation

Table 1 Hyperparameters used in the training models

Model	Hyperparameters	Value
J48	Min samples in the node	20
	Confidence factor	0.05
Logistic regression	Ridge value	0.1
	Maximum number of iterations	20
MLP	Hidden layers	2
	Learning rate	0.3
	Momentum	0.3
Random forest	Max. depth of tree	7
	The number of trees	100
XGBoost	Max. depth of tree	7
	Eta	0.3
CSForest	Confidence factor	0.25
	Min. required leaf	10
	Number of trees	50
	Separation	0.2

functions. These functions integrate both additions and multiplications in Kolmogorov-Gabor polynomials. Waibel (1989) introduced a time-delay neural network architecture for phoneme recognition, employing sequential delayed inputs for each neuron. This dynamic modeling approach coheres with the feedforward structure, ensuring that there is no feedback loop from subsequent layers to preceding ones. The portrayal of input layers through their temporal lags paves the way for the application of artificial neural networks beyond the scope of time series forecasting. Consequently, it extends to other domains, including but not limited to visual perception, speech recognition, and motor control. This expanded application potential demonstrates the flexible and wide-ranging utility of MLP in diverse problem-solving contexts.

MLP possesses three distinct layers—input, hidden, and output. The input layer is responsible for receiving and processing the initial signal, while the output layer executes the requisite tasks such as prediction and classification. Situated between the input and output layers, the hidden layer constitutes the primary computational mechanism of the MLP. Within this architecture, data flow follows a singular direction, progressing from the input to the output layer, emulating the dynamics of a feed-forward network. The neurons constituting the MLP are trained through the backpropagation learning algorithm, with all nodes in the network exhibiting sigmoid characteristics. A fundamental attribute of MLPs is their ability to approximate any continuous function, enabling them to address problems that are not linearly separable (Abirami & Chitra, 2020; Tamouridou et al., 2018).

Conceptualized as a supplement to the feed-forward neural network, the MLP shares the same structural layers—input, hidden, and output—and enforces the same pattern of data flow. The input layer handles the initial signal, while the output layer takes charge of classification and prediction tasks. The intermediary hidden layer, constituting an indefinite series of directly interconnected mechanisms, is the central feature of the MLP. This intricate network of layers is trained using the backpropagation learning method, facilitating a structure that is capable of approximating any continuous function and addressing non-linearly separable challenges (Tamouridou et al., 2018).

The following hyperparameters were adjusted to achieve the optimal performance of the MLP: The optimal hidden layer size was sought within the range of [1–5]; similarly, both the learning rate and momentum rate were explored within the interval [0.1–0.5] with a step size of 0.1 (see Table 1 for hyperparameters used in the training models).

3.4 Random Forest

Random Forest (RF), as proposed by Breiman (2001), is an ensemble learning method that builds a multiplicity of decision trees using different samples and variables. The construction of RF relies on bootstrapped data sets, randomly selecting subsets of variables to create numerous tree models. Unlike the Classification and Regression Trees (CART) approach, which optimizes a single predictor, RF enhances predictive capacity by relying on multiple tree predictors. This model

proves effective in studying large data sets with numerous explanatory variables, as it can filter out irrelevant variables while retaining the most pertinent ones for analysis.

RF, a bagging technique, typifies the ensemble techniques that amalgamate multiple models to enhance predictive performance. Bagging, also known as bootstrap aggregation, comprises multiple predictive models of the same type, primarily serving to reduce variance rather than bias. Random Forest, an extension of the decision tree classifier, employs the bagging technique to mitigate overfitting issues. Each tree within the forest predicts a classification, and the final decision is determined by the majority vote of these trees. This framework encapsulates the "wisdom of crowds" concept, asserting that a group of uncorrelated trees can outperform any individual model, especially when there's minimal correlation among models. The principal assumptions for an RF classifier include a low correlation among the estimations of individual trees and the availability of actual feature variable data to facilitate accurate prediction (Srivastava et al., 2023).

RF models are potent machine learning models that make predictions by combining outcomes from a sequence of regression decision trees. Each tree depends on a random vector sampled from the input data, with all trees sharing the same distribution. The predictions from these trees are averaged using bootstrap aggregation and random feature selection, demonstrating robust performance with both small sample sizes and high-dimensional data (Biau & Scornet, 2016). RF implements the Gini index to determine the optimal split threshold of input values for given classes, measuring class heterogeneity within child nodes compared to the parent node (Breiman, 2017).

3.5 XGBoost

XGBoost is a scalable and efficient implementation of the gradient boosting algorithm, which has gained popularity due to its speed and performance. Introduced by Chen and Guestrin (2016), XGBoost is an open-source software library that provides a gradient-boosting framework for languages such as Python, R, and Julia.

Gradient boosting is a machine learning technique that produces a prediction model, typically in the form of an ensemble of weak prediction models like decision trees. It builds the model in a stage-wise fashion, and it generalizes them by allowing optimization of an arbitrary differentiable loss function. XGBoost improves upon this method by incorporating a regularized model formalization to control over-fitting, thereby enhancing its performance.

As a decision-tree-based ensemble machine learning algorithm, XGBoost depends on a regularized gradient boosting framework. The reason for adding the "regularized" term is that the XGBoost algorithm allows the formal control of the variable weights in contrast to the standard gradient boosting framework. Regularization of the variables comes with a penalty algorithm using the LASSO and Ridge regularization and pushes the weights of some variables to zero. This enhancement in the algorithm prevents overfitting without a loss in forecasting power and reduces the training time.

XGBoost provides a parallel tree-boosting algorithm that solves many data science problems quickly and efficiently. The core XGBoost algorithm is parallelizable, which means it can harness all of the processing power of modern multi-core computers. Moreover, it is also capable of handling missing values, imbalanced datasets, and a mix of categorical and numerical variables, making it versatile across a broad range of datasets and problem domains (see for more explanation Chen & Guestrin, 2016).

3.6 CSForest

The evolution of machine learning algorithms has led to the emergence of cost-sensitive decision forests (CSFs), a promising approach for dealing with imbalance and varying misclassification costs in data. CSFs consider the cost associated with misclassifications in their construction, hence generating more precise predictive models (Lomax & Vadera, 2013).

Random Forests (RFs), an ensemble learning method built upon decision trees, has shown promising results in handling high-dimensional data (Breiman, 2001). Yet, RFs and traditional decision tree models generally assume equal misclassification costs, which can result in significant errors when the costs are, in reality, unequal (Elkan, 2001). To handle cost imbalance, cost-sensitive decision trees (CSDTs) were introduced (Ling & Li, 1998). They incorporate the cost matrix into the decision tree learning process, producing cost-sensitive splits. However, these models may overfit in response to cost complexities (Lomax & Vadera, 2013). Expanding on this idea, Cost-Sensitive Random Forests (CSRFs) were introduced, combining the benefits of RFs and the cost-sensitive nature of CSDTs (Khan et al., 2010). CSRFs incorporate the cost matrix into the random forest algorithm, resulting in cost-aware ensembles of decision trees. These have shown improved performance in cases with varying misclassification costs (Zhou and Liu, 2005).

4 Results

4.1 Dataset and Experimental Settings

The data was obtained from 1482 national banks operating in the U.S. between 2008 to 2010 during the global financial crisis and its aftermath. 59 banks out of 1482 faced financial failure due to the negative effects of the global financial crisis. Healthy banks were chosen from the 2008 dataset where the banks took the biggest hit from the financial crises and ratios were altered significantly. Therefore, being able to classify failed banks would be more complex and more realistically applicable. 32 Financial ratios are selected according to the CAMELS (Capital, Asset Quality, Management, Earnings, Liquidity, Sensitivity ratios) system and they can be found in Appendix 1. FDIC (Federal Deposit Insurance Corporation) database was used to retrieve the dataset and only national banks were selected for both failed and healthy banks. In order to converge the experiment on a more reliable basis, we

did not remove any instances except those in which there is no data available. The data mining toolkit WEKA (Waikato Environment for Knowledge Analysis) 3.9.5 version was used for all the empirical models. However, for XGBoost, an R package extension is required to install and embed within WEKA.

We used a stratified ten-fold cross-validation to assess model performances since it is generally applied to minimize bias associated with a random sampling of training and hold-out data samples (Chou & Pham, 2013) and it also generates the optimal variance and process time (Chou et al., 2011).

The optimization of hyperparameters of the machine learning models such as confidence factor, ridge value, maximum number of iterations, hidden layers, learning rate, and momentum is presented below in Table 1.

4.2 Results and Discussion

To categorize banks into failing and non-failing, we utilize six distinct models. J48 is a traditional statistical classifier that is founded on the C4.5 decision tree algorithm. Leveraging the sigmoid function, logistic regression serves as a classification algorithm, modeling the likelihood of binary outcomes from predictor variables. The multilayer perceptron stands as the most prevalent neural network structure within related literature.

Random forest is a bagging-type ensemble classifier, first proposed by Breiman (2001), that evolves from decision trees. XGBoost, a recent addition to decision-tree-based ensemble machine learning algorithms, was first employed for bank failure classification by Carmona, Climent & Momparler (2019). Each of these models offers unique strengths and weaknesses, contingent on their underlying machine learning algorithms.

Siers and Islam (2015) developed CSForest for software defect prediction (SDP) and stated that:

> "For the conventional classification task in data mining, a classifier is generally built seeking to minimize the number of misclassified records and thereby maximize the prediction accuracy for future records. However, in SDP a classifier is often built in order to minimize the classification cost, which is the cost associated with the classification made. That is, in SDP the classification cost is more important than the number of misclassified records. The cost of a false negative (i.e. a module being actually defective but predicted as non-defective) is generally several times higher than the cost of a false positive (i.e. a module actually being non-defective but predicted as defective). Therefore, it is often better to have several false-positive predictions in order to avoid a single false negative prediction.".

The cost of a false negative (i.e., a bank actually failed but predicted as non-failed) is also much costlier for the banking sector than the cost of a false positive (i.e., a bank actually being non-failed but predicted as failed). Furthermore, Siers and Islam (2015) proposed the CSForest algorithm as a solution to address the class imbalance problem in the field of software defect detection. This parallel is

Table 2 Confusion matrix

		Predicted	
Actual	J48	Non-failure	Failure
	Non-failure	1416	7
	Failure	19	40
	Logistic	Non-failure	Failure
	Non-failure	1414	9
	Failure	11	48
	MLP	Non-failure	Failure
	Non-failure	1417	6
	Failure	14	45
	RF	Non-failure	Failure
	Non-failure	1419	4
	Failure	18	41
	XGBoost	Non-failure	Failure
	Non-failure	1414	9
	Failure	20	39
	CSForest	Non-failure	Failure
	Non-failure	1410	13
	Failure	12	47

particularly relevant to challenges encountered in the banking sector, where similar imbalances exist.

Following Siers and Islam (2015), we focus on *false-negative* but also report the false positive and true positive rates (see Table 3). All other metrics can be extracted from the confusion matrix (see Table 2). False-negative rate (FNR) presents the rate of the failed banks but is predicted (classified) as non-failure to all non-failure banks. False-negative is also known as type-II error and the probability of making a type II error (β) also gives us the FNR. For example, the FNR for the J48 model is 32.20% since J48 classified 19 banks out of 59 failed banks as non-failed. On the other hand, the false-positive rate (FPR) presents the rate of the non-failed bank but is predicted as failed to all non-failed banks. False-positive is also known as Type-I error and the probability of making a type I error (α) also gives us the FPR. In this case, FPR for the j48 model is 0.49% since j48 classified 7 banks out of 1423 non-failed banks as failed. True positive rate (1-α)

Table 3 False negative/positive and true positive rates (%)

	FNR (%)	FPR (%)	TPR (%)
J48	32.20	0.49	99.51
Logistic	18.64	0.63	99.37
MLP	23.73	0.42	99.58
RF	30.51	0.28	99.72
XGBoost	33.90	0.63	99.37
CSForest	20.34	0.91	99.09

represents the rate of correct classification of non-failed banks to all non-failed banks. The true positive rate (TPR) is 99.51% for the J48 model (1416/1423) (Table 3).

Type II error is crucial for players in the financial system such as financial institutions, regulatory institutions, and credit rating agencies. The lower Type II error rate creates a strong early warning signal for regulatory institutions such as FDIC and these banks can be taken under close monitoring of their capital adequacy ratios and their risk profiles.

On the other hand, classifying a non-failure bank as a default bank, type I error, may create mispricing of the assets and liquidities of the bank, increase the average cost of funding, and need more financial regulations. In the end, while type I error in the banking sector can be manageable with low cost, type II error can cause crashes in the whole banking system.

False-negative rates, and Type II error rates, are presented in Fig. 1. Logistic Regression has the smallest false-negative rate with 18.64%. The second-best model is the CSForest model with a 20.34% false-negative rate. In other words, Logistic Regression predicts only 11 "false negatives" out of 59 failure banks, CSForest produces 12 "false negatives" and XGBoost as the worst model has turned 20 "false negatives".

But if we focus on the true positive rate (see Fig. 2), the CSForest model would be the best model with 99.72%. To put it differently, the CSForest has correctly identified 1410 non-failure banks out of 1423 banks. In fact, all models have high true positive rates, the lowest one is J48 with 99.09%.

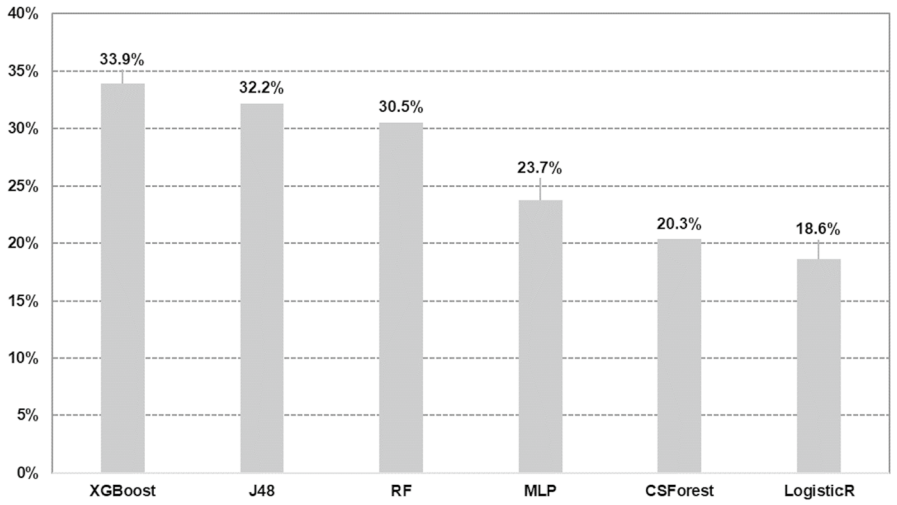

Fig. 1 False negative rates (Type II Error Rates, β)

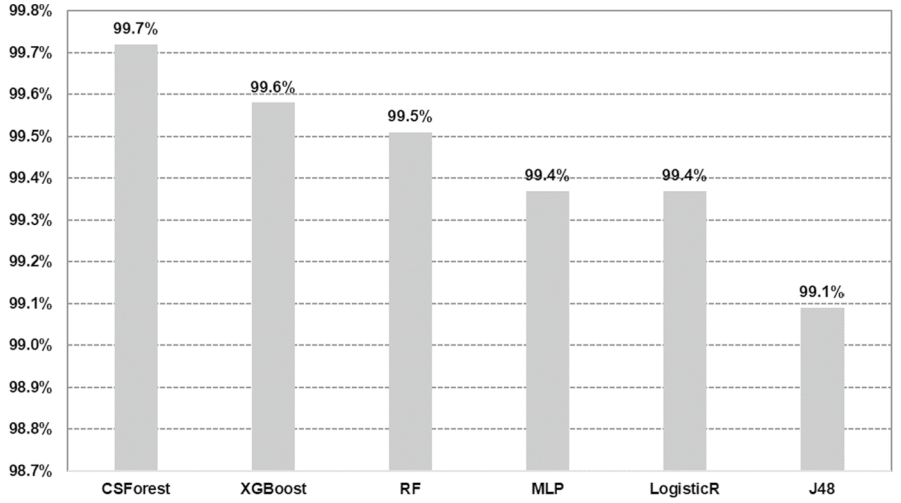

Fig. 2 True positive rates (1-α)

5 Conclusion

The pivotal role of banks in facilitating economic transactions, functioning as an intermediary between agents with surplus and deficit resources, underscores their importance as the linchpin of financial systems and economic stability. Consequently, the examination of bank failure has emerged as a paramount theme within the scholarly discourse. Yet, previous literature predominantly accentuates the general classification success and Area Under the Curve (AUC) ratio, thereby amalgamating the evaluation of robust and fragile banks. This modus operandi renders the essential goal, namely, the precise evaluation of failed banks, increasingly challenging to actualize. Therefore, our primary focus was directed towards the classification capacity of machine learning models to identify failed banks accurately. The data set for our investigation comprised 1482 national banks operative within the U.S. between 2008 to 2010, coinciding with the worldwide financial crisis and its ensuing repercussions. Out of these, 59 banks succumbed to financial failure as a result of the adverse impacts of the global financial crisis.

In the extant literature, the CSForest model has been leveraged to classify software defects within imbalanced data sets. Notwithstanding, we have extended its application to the banking sector for predicting bank failures, given the analogous issue of data set imbalance which necessitates resampling. The findings of our study revealed that Logistic Regression, a conventional classification model, outperformed in minimizing Type II error, followed closely by CSForest. In contrast, XGBoost, despite being highly rated for its accuracy in recent literature, ranked only fifth. However, when considering true positive rates, CSForest emerged as the most accurate model, with an impressive accuracy of 99.72%. In light of the results from our analysis of the various models, we propose employing CSForest to minimize Type II errors in imbalanced bank failure data sets while maintaining high true positive

rates. Additionally, the continued efficacy of Logistic Regression should not be overlooked, underscoring its continued relevance as a benchmark model.

6 Limitations and Recommendations for Further Studies

This study encompasses financial institutions that either faced insolvency or maintained fiscal soundness during the 2008 financial crisis, and as such, the findings are reflective of this specific dataset. Consequently, it is not conclusive to assert that the outcomes of this research comprehensively illuminate prospective banking crises. Instead, the principal objective of this study is to introduce a novel model into the existing literature on forecasting bank failures, an approach heretofore unexplored. This model addresses a prominent challenge in bank failure forecasting, namely the presence of imbalanced datasets wherein the instances of failed banks are typically markedly fewer than their solvent counterparts, thereby resulting in diminished diagnostic performance of models for identifying failed banks.

Appendix 1: The Financial Ratios Used in the Study (CAMELS)

	Ratios
1	Yield on earning assets
2	Cost of funding earning assets
3	Net interest margins
4	Non Interest incomes to average assets
5	Noninterest expenses to average assets
6	Credit loss provision to assets*
7	Net operating income to assets
8	Return on assets (ROA)
9	Pre Tax returns on assets
10	Return on Equity (ROE)
11	Retained earnings to average equity (YTD only)
12	Net charge-offs to loans
13	Loan and lease loss provision to net charge-offs
14	Earnings coverage of net charge-offs (x)
15	Efficiency ratio
16	Assets per employee ($ millions)
17	Earning assets to total assets ratio
18	Loan and lease loss allowance to loans
19	Loan and lease loss allowance to noncurrent loans
20	Noncurrent assets plus other real estate owned to assets
21	Noncurrent loans to loans
22	Net loans and leases to total assets
23	Net loans and leases to deposits

	Ratios
24	Net loans and leases to core deposits
25	Total domestic deposits to total assets
26	Equity capital to assets
27	Leverage (core capital) ratio
28	Total risk-based capital ratio (No CBLR electors)
29	Average total assets
30	Average earning assets
31	Average equity
32	Average total loans

Author Contributions All authors contributed equally to the study conception, design, data, analyses, and the writing of the paper. All authors read and approved the final manuscript.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Declarations

Conflict of Interests The authors have no relevant financial or non-financial interests to disclose.

Ethical approval No particular ethical approval was required for this study because it does not entail human participation or personal data.

References

Abirami, S., & Chitra, P. (2020). Energy-efficient edge based real-time healthcare support system. *Advances in Computers.* Elsevier.

Agrapetidou, A., Charonyktakis, P., Gogas, P., Papadimitriou, T., & Tsamardinos, I. (2021). An AutoML application to forecasting bank failures. *Applied Economics Letters, 28*(1), 5–9.

Anari, A., Kolari, J., & Mason, J. (2005). Bank asset liquidation and the propagation of the U.S. great depression. *Journal of Money, Credit and Banking, 2005*, 753–773.

Ashcraft, A. B. (2003). Are banks really special? New evidence from the FDIC-induced failure of healthy banks. *American Economic Review, 95*, 1712–1730.

Ashcraft, A. B. (2005). Are banks really special? New evidence from the FDIC-induced failure of healthy banks. *The American Economic Review, 95*(5), 1712–1730.

Bell, T. B. (1997). Neural nets or the logit model: A comparison of each model's ability to predict commercial bank failures. *International Journal of Intelligent Systems in Accounting, Finance and Management, 6*, 249–264.
Berger, A. N., & Bouwman, C. H. (2012). How Does Capital affect bank performance during financial crises? *Journal of Financial Economics (JFE)., 109*(1), 146–176.
Bernanke, B. S. (1983). Nonmonetary effects of the financial crisis in the propagation of the great depression. *American Economic Review, 73*, 257–276.
Bernanke, B., Gertler, M., & Gilchrist, S. (1996). The financial accelerator and the flight to quality. *The Review of Economics and Statistics, 78*(1), 1–15.
Biau, G., & Scornet, E. (2016). A random forest guided tour. *TEST, 25*, 197–227.
Boyd, J. H., Kwak, S., & Bruce, D. (2005). The real output losses associated with modern banking crises. *Journal of Money, Credit, and Banking, 37*, 977–999.
Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32.
Breiman, L. (2017). *Classification and regression trees*. Routledge.
Calomiris, C. W. (1993). Financial factors in the great depression. *Journal of Economic Perspectives, 7*, 61–85.
Calomiris, C. W., & Mason, J. R. (2003). Consequences of bank distress during the great depression. *American Economic Review, 93*(3), 937–947.
Carmona, P., Climent, F., & Momparler, A. (2019). Predicting failure in the U.S. banking sector: An extreme gradient boosting approach. *International Review of Economics & Finance, Elsevier, 61*(C), 304–323.
Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. arXiv:1603.02754v3 [cs. LG].
Chiaramonte, L., Croci, E., & Poli, F. (2015). Should we trust the Z-score? Evidence from the European Banking Industry. *Global Finance Journal, 28*, 111–131.
Chiaramonte, L., Liu, H., Poli, F., & Zhou, M. (2016). How accurately can Z-score predict bank failure? *Financial Markets, Institutions & Instruments, 25*(5), 333–360.
Chou, J. S., Chiu, C. K., Farfoura, M., & Al-Taharwa, I. (2011). Optimizing the prediction accuracy of concrete compressive strength based on a comparison of data-mining techniques. *Journal of Computing in Civil Engineering, 25*(3), 242–263.
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.
Cleary, S., & Hebb, G. (2016). An efficient and functional model for predicting bank distress: In and out of sample evidence. *Journal of Banking & Finance, 64*, 101–111.
Ekinci, A., & Erdal, H. I. (2011). An application on prediction of bank failure in Turkey. *Iktisat Işletme Ve Finans, 26*(298), 21–44.
Ekinci, A., & Erdal, H. I. (2017). Forecasting bank failure: Base learners, ensembles and hybrid ensembles. *Computational Economics, 49*(4), 677–686.
Elkan, C. (2001). The foundations of cost-sensitive learning. In *International Joint Conference on Artificial Intelligence* (Vol. 17, No. 1, pp. 973–978). Lawrence Erlbaum Associates Ltd.
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.
Friedman, M., & Schwartz, A. J. (1963). *A monetary history of the United States, 1867–1960*. Princeton University Press.
Gogas, P., Papadimitriou, T., & Agrapetidou, A. (2018). Forecasting bank failures and stress testing: A machine learning approach. *International Journal of Forecasting, 34*(3), 440–455.
Hoggarth, G., Reis, R., & Saporta, V. (2002). Costs of banking system instability: Some empirical evidence. *Journal of Banking & Finance, 26*, 825–855.
Ivakhnenko, A. G., & Lapa, V. G. (1966). Cybernetic Predicting Devices, Purdue University Lafayette and Ind. School of Electrical Engineering.
Kang, J. K., & Stulz, R. M. (2000). Do banking shocks affect borrowing firm performance? An analysis of the Japanese experience. *The Journal of Business, 73*(1), 1–23.
Khan, A., Baharudin, B., Lee, L. H., & Khan, K. (2010). A review of machine learning algorithms for text-documents classification. *Journal of Advances in Information Technology, 1*(1), 4–20.
Kima, M., & Kangb, D. (2010). Ensemble with neural networks for bankruptcy prediction. *Expert Systems with Applications., 37*(4), 3373–3379.

Kupiec, P. H., & Ramirez, C. D. (2013). Bank failures and the cost of systemic risk: Evidence from 1900 to 1930. *Journal of Financial Intermediation., 22*(3), 285–307.
Lagasio, V., Pampurini, F., Pezzola, A., & Quaranta, A. G. (2022). Assessing bank default determinants via machine learning. *Information Sciences, 618*, 87–97.
Le Cessie, S., & Van Houwelingen, J. C. (1992). Ridge estimators in logistic regression. *Journal of the Royal Statistical Society: Series C (applied Statistics), 41*(1), 191–201.
Lee, H., & Viviani, J. (2017). Predicting bank failure: an improvement by implementing Machine learning approach on classical financial ratios. *Research in International Business and Finance., 44*, 16–25. https://doi.org/10.1016/j.ribaf.2017.07.104
Ling, C. X., & Li, C. (1998). Data mining for direct marketing: Problems and solutions. In *Kdd* (Vol. 98, pp. 73–79).
Lomax, S., & Vadera, S. (2013). A survey of cost-sensitive decision tree induction algorithms. *ACM Computing Surveys (CSUR), 45*(2), 1–35.
Lu, W., & Whidbee, D. A. (2013). Bank structure and failure during the financial crisis. *Journal of Financial Economic Policy, 5*(3), 281–299.
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*(2), 786–801.
Martin, D. (1977). Early warning of bank failure: A logit regression approach. *Journal of Banking and Finance, 1*, 249–276.
Meyer, P. A., & Pifer, H. W. (1970). Prediction of bank failures. *The Journal of Finance, 25*(4), 853–858.
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. https://doi.org/10.1080/1331677X.2019.1689838
Olmeda, I., & Fernandez, E. (1997). Hybrid classifiers for financial multicriteria decision making: The case of bankruptcy prediction. *Computational Economics., 10*(4), 317–335.
Paramjeet, R. V., & Nekuri, N. (2012). Privacy preserving data mining using particle swarm optimisation trained auto-associative neural network: an application to bankruptcy prediction in banks. *International Journal of Data Mining, Modelling and Management., 4*(1), 39–56.
Petropoulos, A., Siakoulis, V., Stavroulakis, E., & Vlachogiannakis, N. E. (2020). Predicting bank insolvencies using machine learning techniques. *International Journal of Forecasting, 36*(3), 1092–1113.
Ramirez, C. D., & Shively, P. A. (2012). The effect of bank failures on economic activity: Evidence from U.S. States in the Early 20th Century. *Journal of Money, Credit and Banking., 44*(2–3), 433–455.
Ramu, K., & Ravi, V. (2009). Privacy preservation in data mining using hybrid perturbation methods: An application to bankruptcy prediction in banks. *International Journal Data Analysis Techniques and Strategies, 1*(4), 313–331.
Ravi, V., Kurniawan, H., Thai, P. N. K., & Kumar, R. (2008). Soft computing system for bank performance prediction. *Applied Soft Computing., 8*(1), 305–315.
Ravi, V., & Pramodh, C. (2010). Non-linear principal component analysis-based hybrid classifiers: An application to bankruptcy prediction in banks. *International Journal of Information and Decision Sciences, 2*(1), 50–67.
Rockoff, Hugh T., The Meaning of Money in the Great Depression (December 1993). NBER Working Paper No. h0052, Available at SSRN: https://ssrn.com/abstract=559222
Shrivastava, S., Jeyanthi, P. M., & Singh, S. (2020). Failure prediction of Indian Banks using SMOTE, Lasso regression, bagging and boosting. *Cogent Economics & Finance, 8*(1), 1729569.
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. https://doi.org/10.1016/j.is.2015.02.006
Sinkey, J. F. (1975). A multivariate statistical analysis of the characteristics of problem banks. *Journal of Finance, 30*(1), 21–36.
Srivastava, R., Kumar, S., & Kumar, B. (2023). Classification model of machine learning for medical data analysis. *Statistical Modeling in Machine Learning* (pp. 111–132). Academic Press.
Swicegood, P., & Clark, J. A. (2001). Off-site monitoring for predicting, bank under performance: A comparison of neural networks, discriminant analysis and professional human judgment. *International Journal of Intelligent Systems in Accounting, Finance and Management., 10*, 169–186.
Tam, K. Y. (1991). Neural network models and the prediction of bank bankruptcy. *Omega, 19*(5), 429–445.

Tam, K. Y., & Kiang, M. (1992). Predicting bank failures: A neural network approach. *Decision Sciences., 23*, 926–947.

Tamouridou, A. A., Pantazi, X. E., Alexandridis, T., Lagopodi, A., Kontouris, G., & Moshou, D. (2018). Spectral identification of disease in weeds using multilayer perceptron with automatic relevance determination. *Sensors, 18*(9), 2770.

Torna, G., & DeYoung, R. (2012). Nontraditional Banking Activities and Bank Failures During the Financial Crisis. *Journal of Financial Intermediation., 22*(3), 397–421. https://doi.org/10.2139/ssrn.2032246

Verikas, A., Kalsyte, Z., & Bacauskiene, M. (2010). Hybrid and ensemble-based soft computing techniques in bankruptcy prediction: A survey. *Soft Computing., 14*(9), 995.

Waibel, A. (1989). Modular construction of time-delay neural networks for speech recognition. *Neural computation*, *1*(1), 39–46.

West, R. C. (1985). A factor analytic approach to bank condition. *Journal of Banking and Finance., 9*, 253–266.

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Philip, S. Y., et al. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems, 14*(1), 1–37.

Yadav, A. K., & Chandel, S. S. (2015). Solar energy potential assessment of western Himalayan Indian state of Himachal Pradesh using J48 algorithm of WEKA in ANN based prediction model. *Renewable Energy, 75*, 675–693.

Zhou, Z. H., & Liu, X. Y. (2005). Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on Knowledge and Data Engineering, 18*(1), 63–77.