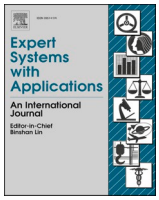
Expert Systems With Applications 223 (2023) 119882

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



Credit risk evaluation using clustering based fuzzy classification method

Furkan Baser [a,*], Oguz Koc [b], A. Sevtap Selcuk-Kestel [b]

[a] Department of Actuarial Sciences, Faculty of Applied Sciences, Ankara University, 06590 Ankara, Turkey
[b] Institute of Applied Mathematics, Middle East Technical University, 06800 Ankara, Turkey

ARTICLE INFO

Keywords:
Machine learning
Fuzzy classification
Clustering
Credit scoring
Home credit risk

ABSTRACT

Credit scoring is a crucial indicator for banks to determine the financial position and the eligibility of a client for credit. In order to assign statistical odds or probabilities to predict the risk of nonpayment in relation to many other factors, the scoring criterion becomes an important issue. The focus of this study is to propose a clustering based fuzzy classification (CBFC) method for credit risk assessment. We aim to illustrate the beneficial use of machine learning (ML) methods whose prediction power is increased by adopting fuzzy theory to calculate the default risk with a better selection of the features contributing to it. An important feature of the CBFC method is that membership values obtained as a result of the fuzzy k-means clustering algorithm are used for the purpose of better capturing the structure of an existing system.

An extensive comparison is performed to show how CBFC performs compared to the traditional ones for the datasets having different characteristics in terms of the variable types. Five different real-life datasets are studied to expose the contribution of fuzzy approach on improving the ML use. Our findings show that the proposed CBFC models can produce the promising classification results in credit risk evaluation which aid the practitioners and decision makers for issuance of credit purposes.

1. Introduction

The implementation of ML becomes very attractive, especially for processing large datasets (Alpaydin, 2016) and yields high performance in prediction, illustrating the strength in modeling. Due to their developments in the recent decade, ML methods allow us to benefit from diversity in the choice of methods and flexibility in handling the data and modeling, specifically on financial data whose structure is complex, multi-dimensional, nested with other factors and sensitive to exogenous effects. The banking sector whose financial risk is highly uncertain and difficult to predict due to its embedded structure with financial market, operational framework, economics, regulatory interference and global developments is prone to many risks. Among those the most commonly considered one is default risk, mainly due to credit transactions which may arise in respect to market risk, economic recessions and other specifications (Lessmann et al., 2015). A recent history (subprime crisis in 2008) has shown that, as a result of the unrestrained growth of banking sector and financial globalization, the financial crisis mainly arising from housing credit transactions in the USA, spread worldwide causing big financial disasters at every level in the world. Homeowners credit risk is another source of concern for the banks, especially after the subprime crisis in 2008 which was originally triggered due to house credits by home loans. Consequently, the lack of risk management in the banking sector before the subprime crisis results in unpredictable losses. The regulatory framework such as Basel III requires banks to implement risk management to determine their compatibility, competitiveness, resilience to financial crises and profitability (Gatzert & Wesker, 2012). Therefore, it gains importance for the banks to predict the amount of risk they are prone to by quantifying the credit score at institutional and individual-client bases.

Measuring the risk for credit issuance can be done either solely on the expert judgment (credit analyst) or by implementing a classification criterion based on the current and historical financial strength of the client (Crook, 1996). It gains importance in the risk management system to embody predictive methods to estimate a reliable risk measure. Credit scoring relies on a number of mainstream modeling approaches, among which discriminant analysis and logistic regression are quite popular due to their comprehensibility and practicality on implementation. Along with many predictive approaches such as probabilistic and econometric, using the most suitable ML classification algorithms assures a robust, reliable and modernized credit risk assessment (Anderson, 2007).

* Corresponding author.
E-mail addresses: furkan.baser@ankara.edu.tr (F. Baser), oguz.koc@metu.edu.tr (O. Koc), skestel@metu.edu.tr (A.S. Selcuk-Kestel).

https://doi.org/10.1016/j.eswa.2023.119882
Received 21 August 2022; Received in revised form 11 March 2023; Accepted 13 March 2023
Available online 17 March 2023

Literature offers vast amount of ML studies on credit scoring domain that suggest using artificial neural network (Zhao et al., 2015, Liang & Cai, 2020), support vector machine (Harris, 2015; Yu et al., 2020), decision tree (Teles et al., 2020, Golbayani et al., 2020), nearest neighbor (Hand & Henley, 1997), and Naïve Bayes classifier (Marqués et al., 2012). Except for the application of these single classifiers, complex credit scoring models have also been investigated by many researchers to adequately express the real contents of data. Due to its training efficiency and low computational requirements, ensemble learning, such as XGBoost, LightGBM, and CatBoost has been popularized for large-scale credit scoring (Ma et al., 2018; Xia et al., 2020a, 2021; Liu et al., 2022). Furthermore, ensemble methods are combined with different base classifiers to create heterogeneous ensembles that enable diversified predictions and improve their adaptability to different credit datasets (Lessmann et al., 2015; Xia et al., 2018, 2020b). In the other studies on ensemble classification for credit risk (e.g. Feng et al., 2019; Junior et al., 2020; Chen et al., 2021), the analyses are conducted from different perspectives and on different datasets. On the other hand, Gunnarsson et al. (2021) suggest deep learning methods to estimate credit risk in comparison with conventional methods, whose experimental results show that deep learning do not outperform ensemble algorithms in credit scoring applications.

In the implementation of ML methods, the success in prediction depends highly on the algorithm chosen, the feature selection, the data conversion, and the validation structure (Selcuk et al., 2022). The different combination of these creates countless trials, therefore, choosing the most relevant one requires not only the satisfactory performance measures, but also a good understanding of the variables and the technical properties of the ML algorithms (Koc, 2019). Credit rating analysis often requires consideration of many uncertain, difficult to define, and even interrelated and conflicting factors (Syau et al., 2001). This uncertainty that arises in risk assessment has long required a reliable and consistent system modeling approach to assist decision making processes. The fuzzy logic technique has reached a wide area of use in modeling different forms of uncertainties, ambiguities and the human thought process.

In recent years, credit risk assessment performed by combining fuzzy set analysis with ML techniques has been shown to be more effective in improving model performance (Bai et al., 2019). For instance, Malhotra and Malhotra (2002) propose the use of neuro-fuzzy modeling in consumer credit applications and demonstrate the advantages of neuro-fuzzy systems over traditional statistical techniques. Ramkumar (2016) builds a risk assessment model for third-party e-procurement systems by using fuzzy inference system and a modified analytical network process. Sohn et al. (2016) integrate fuzzy logic into the traditional logistic regression (LR) model to predict the probability of default. The proposed fuzzy LR model is determined to have higher model accuracy than the traditional LR. Shi and Xu (2016) propose a fuzzy support vector machine algorithm for credit risk analysis, in which the different contribution of each input point to the learning of the classification hyperplane is provided by the fuzzy membership values. Sun et al. (2022) introduce a multilayered modelling approach based on using fuzzy logic for qualitative indicators of credit risk. This fuzzy decision-making method is applied in predicting the credit ratings of small industrial enterprises.

Fuzzy-based ML approaches can be used as a remedy for unwanted over fitting caused by treating every data sample equally, however, the effect of the outliers when training fuzzy models is an interesting issue in the context of credit scoring modeling (Shieh & Yang, 2008). Credit data is usually considered to be unbalanced in terms of class distribution as the number of non-default cases is often much higher than the default ones. In classification models such unbalanced form results in biased predictions in favor of the majority class (no default), so that classification accuracy remains too low for the minority class (default). However, banks or financial institutes place more emphasis on detecting minority class cases (default) in practice, due to its relatively high risk leading to severe financial losses (Galar et al., 2011; Chang et al., 2020). The literature offers mostly a single credit risk model which is developed based on the historical information on the former borrowers to predict the probability of default or to decide whether to approve a loan for a new application. However, a single classification rule may not be sufficient to develop the most accurate behavioral model due to the heterogeneity of system inputs that depend on borrowers (Lim and Sohn, 2007). This obstacle can be handled using a fuzzy clustering approach assigning relative weights to the occurrences between binary outcomes (Çelikyılmaz and Türkşen, 2007, 2008a, 2008b). In line with this frame, Çelikyılmaz and Türkşen (2009) propose clustering based fuzzy modeling (CBFM) to identify the fuzzy structure of the system for regression and classification tasks. An important characteristic of this method is to assign fuzzy membership values shared among the input variables in order to better express the structure of an existing system. Fuzzy k-means (FKM) clustering algorithm, which derives memberships of an input to the clusters, forms the basis of CBFM. In order to estimate fuzzy regression function parameters for each cluster, Logistic Regression (LR) and Support Vector Machines (SVM) using different kernel functions are implemented.

Developing different classification rules for each cluster may contribute to increase the accuracy in risk classification since borrowers in the same clusters show similar behavior pattern (Correa et al., 2012; Scitovski & Šarlija, 2014; Ghanbari et al., 2014). The effectiveness of segmented modeling on credit risk analysis has been demonstrated in several studies, where borrowers are clustered according to their similarities. Harris (2015) investigates some of the limitations associated with traditional nonlinear support vector machines, and propose the use of clustering based support vector machine approach for credit scorecard evaluation. Zhang et al. (2018) integrate fuzzy assignments to the normal hard clustering in the training and testing phase of the classification algorithm. They determine the best classification model for credit scoring by using a genetic algorithm. Bao et al. (2019) propose an algorithm based on the combined use of clustering and classification methods to evaluate the credit risk of individuals. In this context, the authors discuss seven supervised classification models along with two different clustering models, k-means and self-organizing map. Boughaci et al. (2021) also use k-means clustering prior to the estimation of random forest ensemble model to improve financial distress prediction performance.

With respect to aforementioned characteristics of credit risk data sets, the motivations of this study are to (i) propose a clustering based fuzzy classification (CBFC) approach, in which the observations are distributed to clusters according to FKM clustering to capture default risk much better by improving the prediction capacity of ML algorithms, and (ii) develop an algorithm for calculating a single probability of default (PD) for each input by weighting the model outputs from each cluster with fuzzy membership values. Contributions of this study can be summarized as (i) development of CBFC algorithm with the ML methods, such as k-Nearest Neighbor (kNN), Decision Tree (DT), Random Forest (RF), Gaussian Naïve Bayes (GNB), Artificial Neural Network (ANN), Extreme Gradient Boosting (XGB), Categorical Boosting (CatBoost), Light Gradient Boosting Machine (LGBM) based on the approach of Çelikyılmaz and Türkşen (2009) utilizing LR and SVM, (ii) an extensive comparison of the accuracies of traditional ML models and the proposed CBFCs with different benchmark datasets which have diversity in some aspects such as data volume, number of variables and imbalance ratios. To achieve these contributions, first time in literature we employ kNN, DT, RF, GNB, ANN, XGB, CatBoost, LGBM within the CBFM framework, and we illustrate the efficiency of the proposed method in predicting bank credit risk and home credit risk based on five real datasets. The application of five well-known datasets in the proposed CBFC approach provides important results which are expected to guide the researchers in: (i) activating the important characteristics in credit scoring, (ii) implementing the factors properly to have the maximum accuracy in ML applications using CBFC. We expect the

proposed approach captures default risk much better by improving the prediction capacity of ML algorithms compared to the ones in literature.

This paper is organized as follows: Section 2 is devoted to a brief explanation of CBFC algorithm in structure identification and its inference. Section 3 introduces the implementation of the proposed CBFC methodology. The application to credit datasets are presented thoroughly in Section 3 with the broad discussion on their results. Section 4 finalizes the paper with concluding comments.

2. Clustering based fuzzy classification models

Classification algorithms which have numerous applications in various data mining problems attempt to learn the functional relation between a set of feature variables and target variables. Due to its wide range of applicability, such algorithms allow many variations in the problem to be defined in different settings. More details in solving classification problems can be found in Mitchell and Mitchell (1997); Stork et al. (2001); Hastie et al. (2009). Moreover, classification models are used categorizing the unseen test samples into groups defined by class labels. While the segmentation of observations into groups is done by both clustering and classification, there is a fundamental difference between these two. In clustering problems, segmentation is performed based on similarities between observations. Accordingly, clustering and classification problems in statistical learning algorithms are defined as unsupervised and supervised learning, respectively. In supervised learning, the modeling process often requires application-specific supervision, as class labels represent important features of interest.

The membership values play a crucial role for fuzzy system modeling approaches in which similarities of the objects are described depending on the distance between vectors (Türkşen & Celikyilmaz, 2006). The CBFM is found to give better results in minimizing the error between the system and model outputs compared to the conventional fuzzy rule-based approaches (Baser & Demirhan, 2017; Chakravarty et al., 2020). To identify the structure of the system, CBFM employs fuzzy k-means (FKM) clustering algorithm proposed by Bezdek (1981). The CBFM is useful as the additional membership values are obtained as a result of the fuzzy clustering algorithm (i.e. FKM clustering) which is applied to the original data matrix to describe the functional relation between input (I) and output (O) variables (Çelikyılmaz & Türkşen, 2009).

We propose a CBFC system which is composed of training and inference processes. The system model in training part is analyzed with randomly selected training data from the entire dataset. Afterwards, the estimation accuracy of the model is evaluated by using test or validation data at inference part. A schematic overview of the CBFC algorithm is pictured in Fig. 1 which is designed and illustrated for credit data scheme. The detailed explanation of the steps in this flowchart is presented in the algorithm. The crucial steps of CBFC implementation are clustering I/O data (clustering step) and the estimation of cluster parameters (estimation step). Any observation in the training dataset will be influential at the clustering step, and the estimation of classification models for each cluster determines the full CBFC method.

In this paper, we consider ten ML methods (LR, SVM, kNN, DT, RF, GNB, ANN, XGB, CatBoost, LGBM) to create CBFC–LR, CBFC–SVM, CBFC–kNN, CBFC–DT, CBFC–RF, CBFC–GNB, CBFC–ANN, CBFC–XGB, CBFC–CatBoost and CBFC–LGBM algorithms. Based on the approach made for CBFC–LR and CBFC–SVM (Çelikyılmaz and Türkşen, 2009), we develop fuzzy framework for the other algorithms, such as CBFC–kNN, CBFC–DT, CBFC–RF, CBFC–GNB, CBFC–ANN, CBFC–XGB,

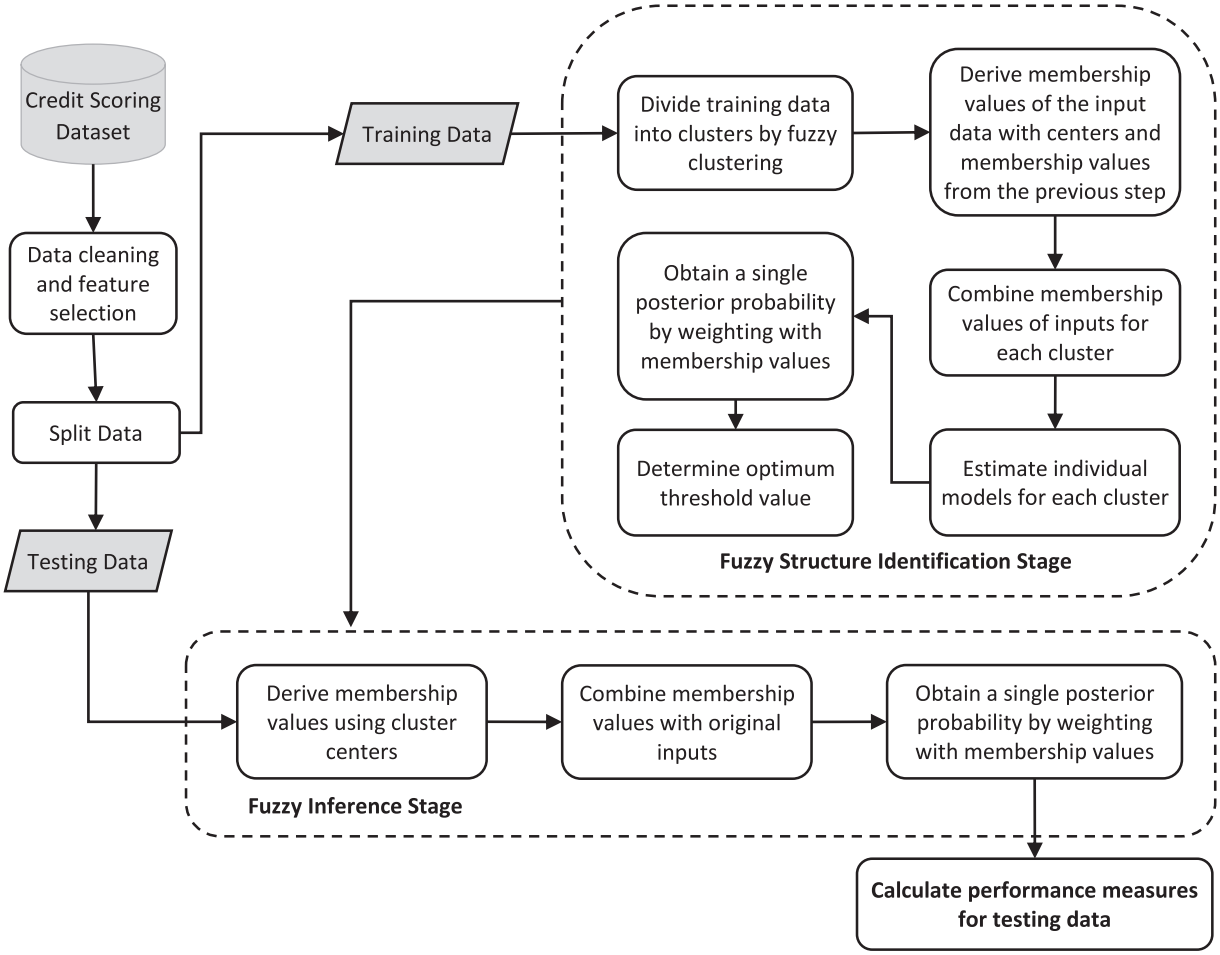


Fig. 1. The flow chart of the proposed CBFC approach for credit data set.

CBFC–CatBoost and CBFC–LGBM which is introduced in this study. Appendix A presents preliminaries on traditional ML algorithms: LR, SVM, kNN, DT, RF, GNB, ANN, XGB, CatBoost, and LGBM.

The process of the proposed CBFC approach utilizing the standard FKM clustering method is explained in two stages as the fuzzy structure identification (training) and the fuzzy inference (testing) in Fig. 1. The dashed boxes are related to these stages. As it can be seen the first part has seven steps to be completed whose results is the input of the second part completed in five steps. The implementation of the steps explained in this flowchart is aimed to be done on credit dataset to predict the default probability of each input.

The structure identification of proposed CBFC algorithm is composed of two parts whose first stage requires the following seven steps:

Step 1. Execute FKM clustering algorithm for $m \geq 1.1$ and $c > 1$ on the training data pairs, $\{(\mathbf{x}_i, y_i); i = 1, \cdots, n\}$. Here, c and m indicate the number of clusters and degree of fuzziness, respectively.

Let $Z = (X, \mathbf{y})$ represent the $(n \times (d+1))$ input–output dataset (matrix) such that $X = \left[x_{j,i} | i = 1, 2, \cdots, n; j = 1, 2, \cdots, d\right]$ where d is the number of selected inputs and n is the total number of data vectors. Thus, $\mathbf{z}_i = (\mathbf{x}_i, y_i) \subset \Re^{d+1}, i = 1, 2, \cdots n$ denote any data point (vector) from the training set, and every data point is composed of $(d+1)$ dimensions of input vectors, $\mathbf{x}_i = (x_{1,i}, x_{2,i}, \cdots, x_{d,i}) \in \Re^d$, and an output, $y_i \in \Re$. Let $\mu_{k,i}(\mathbf{x}, y) \in [0, 1]$ represent the membership value of the i^{th} datum in cluster k.

Accordingly, calculate iteratively the optimum cluster centers, $\mathbf{v}_k(\mathbf{x}, y) = \left(x^c_{1,k}, x^c_{2,k}, \cdots, x^c_{d,k}, y^c_k\right) \in \Re^{d+1}$ and interactive (input–output) membership values, $\mu_{k,i}(\mathbf{x}, y)$ by implementing FKM (Bezdek et al., 1984), as follows:

$$\mathbf{v}_k(\mathbf{x}, y) = \frac{\sum_{i=1}^{n} \left(\mu_{k,i}(\mathbf{x}, y)\right)^m \mathbf{z}_i}{\sum_{i=1}^{n} \left(\mu_{k,i}(\mathbf{x}, y)\right)^m} \tag{1}$$

$$\mu_{k,i}(\mathbf{x}, y) = \left(\sum_{l=1}^{c} \left(d_{k,i}(\mathbf{x}, y) / d_{l,i}(\mathbf{x}, y)\right)^{2/m-1} \right)^{-1} \tag{2}$$

$$d_{k,i}(\mathbf{x}, y) = \|(\mathbf{x}_i, y_i) - \mathbf{v}_k(\mathbf{x}, y)\|; i = 1, 2, \cdots, n; k = 1, 2, \cdots, c \tag{3}$$

In order to determine the optimum values of m and c, cluster validity index (CVI) analysis can be used (Pal and Bezdek, 1995; Kim and Ramakrishna, 2005). We take $m = 2$ to have reasonable clustering results in system modeling analysis based on Türkşen (1999).

Step 2. Identify the cluster centers of the given input space from previous step as $\mathbf{v}_k(\mathbf{x}) = \left(x^c_{1,k}, x^c_{2,k}, \cdots, x^c_{d,k}\right) \in \Re^d$ and obtain the membership values for the input space using

$$\mu_{k,i}(\mathbf{x}) = \left(\sum_{l=1}^{c} \left(d_{k,i}(\mathbf{x}) / d_{l,i}(\mathbf{x})\right)^{2/m-1} \right)^{-1} \tag{4}$$

$$d_{k,i}(\mathbf{x}) = \|\mathbf{x}_i - \mathbf{v}_k(\mathbf{x})\|; i = 1, 2, \cdots, n; k = 1, 2, \cdots, c \tag{5}$$

Step 3. Combine the membership values of inputs, $\mu_{k,i}$ and their user-defined mathematical transformations such as $\mu^2_{k,i}$, $\exp\mu_{k,i}$, $\ln\mu_{k,i}$ with the original inputs for each cluster, k ($k = 1, 2, \cdots, c$). Here, for each k, a different subset can also be obtained with different selection of α–cut by the following constraint:

$$\mu_{k,i} > \alpha - \text{cut}, k = 1, 2, \cdots, c; i = 1, 2, \cdots, l; l < n \tag{6}$$

With the use of α–cut > 0, the vectors having higher deviations from the cluster centers are neglected, and therefore, sub-datasets for each cluster are formed. According to the experimental results, if the number of observations in a cluster remains under n/c, we usually take α-cut $= 0$ (Çelikyılmaz and Türkşen, 2009).

Step 4. Estimate the models using a ML method (LR, SVM, kNN, DT, RF, GNB, ANN, XGB, CatBoost, LGBM) to constitute the fuzzy classifier for each cluster.

Step 5. Transform the output values obtained from each fuzzy classifier into posterior probabilities which can be measured using the approximation of model output labels with a sigmoid function (Kuhn, 2008).

Step 6. Obtain the estimated fuzzy outputs for each cluster as the weighted values with respect to the corresponding membership values yielding a single output value such as

$$\widehat{P}_i = \frac{\sum_{k=1}^{c} \widehat{P}_{k,i} \mu_{k,i}}{\sum_{k=1}^{c} \mu_{k,i}}, k = 1, 2, \cdots, c; i = 1, 2, \cdots, n \tag{7}$$

Step 7. Determine the optimum threshold value of a specific CBFC model based on the estimated posterior probabilities. The optimum threshold is captured whilst subsampling cross validation and is used to measure recognition performance of the data vectors. The recognition performance, i.e., the accuracy, is a probabilistic measure that captures the correctly classified samples as a ratio of the total samples in the dataset.

The test algorithm proposed for the CBFC approach includes an inference process over testing dataset to evaluate the generalization capabilities of the models determined during the structure identification whose five steps are presented as follows:

Step 1. Calculate membership values for each input by using cluster centers.
Step 2. Combine membership values and their transformations of inputs from the previous step with the original inputs of the inference data.
Step 3. Obtain class probabilities of the new inputs as a result of estimated fuzzy classifiers.
Step 4. Calculate single outputs by weighting the inferred class probabilities of each cluster with the corresponding membership values.
Step 5. Use the optimum threshold values obtained during the structure identification to predict a class label, $\widehat{y}_i = 0$ or $\widehat{y}_i = 1$, for each data object which represent either one of the two dichotomous outputs. So, the model performances in terms of its accuracy is measured based on the estimated class labels.

3. Application of Fuzzy-ML method

The main characteristic in credit risk is the diversity in the type of the variables collected for each credit accredited to the customer (either company or individual) whose quality and frame may change over the years. The two data set processed in this paper also show different structure in terms of the variables, their types, sample size and data curation. The comprehensive analysis targeted to understand each entry in a dataset to determine default classes using expert evaluation methods which is followed by eliminating the instances resulting in "null" (empty cells) cases and detecting the features containing a dominating number of missing values. The algorithm explained thoroughly as the proposed CBFC method is applied, and comparative analyses and concluding remarks are discussed in this section. The ML algorithms and proposed method are coded in R, and it is also used for statistical analyses.

3.1. Datasets and features

Due to the nature of their business, the access to credit or default data is uneasy and financial institutions are extremely reluctant to release such information. However, some web-sourced data sets on credit default make it possible to assess credit risk and constitutes a platform to make comparison on the studies applied to these data sets in literature. In this experiment, five datasets are used to test the proposed model to validate its performance.

German Credit data is one of those at which many researchers refer to. In this dataset, each entry represents a person who takes a credit by a bank. Among 1000 credit cases, each person is classified as good or bad credit risks according to the set of features. It contains 20 variables 13 of which are categorical. German credit dataset can be retrieved from https://www.kaggle.com/datasets/uciml/german-credit.

Home Credit (Homeowners) data which is also an accessible data source, contains loan records with basic personal information such as credit amount, income, gender and other more characteristics which enables us to examine the impact of many contributors on credit default. It contains 50 categorical and 70 continuous variables constituting 307,511 lines each of those belonging to home credit owner. Home credit dataset is available at https://www.kaggle.com/datasets/julianocosta/home-credit.

Bank Fear data is published by Indessa Bank. With the messy data collected over all the years, the bank needs a credit scoring model to identify those likely to default. The dataset consists of 532,428 sample, and includes 32 features, 8 of which are categorical. The Bank Fear dataset is available at https://www.kaggle.com/datasets/gauravduttakiit/bank-fears-loanliness.

The PPDai data, provided by He et al. (2018) is derived from the transaction records of an advanced P2P lending platform in China. The dataset consists of 55,596 sample, and includes 29 features, 22 of which are continuous, and 7 are categorical.

Give Me Some Credit (GMSC) data is provided by a ML competition in the Kaggle community. In this competition the research problem is defined as the estimation of the classification model to evaluate whether a consumer can get financing and what conditions can influence or undermine investment decisions. The dataset includes 150,000 samples and 10 features defined as continuous variables. The Give dataset is provided on the Kaggle platform at https://www.kaggle.com/c/GiveMeSomeCredit.

3.2. Data cleaning and feature selection

It is well known that banks are very selective in issuing credits and they make an intensive credit history to accredit the customer and they record many information on the applicant to avoid adverse selection. The data sets on credit risk are commonly digitalized at which some critical information is even masked. This may require a thorough data curation. In five credit datasets selected for this paper, we conduct a data cleaning process due to many missing and inconsistent lines. Data curation focuses on detecting and fixing the missing cells and outliers. To cure the missing data, commonly implemented methods are: deletion, imputation, model-based procedures and machine learning methods (García-Laencina et al., 2010) based on the structure of the dataset.

In this study, in terms of having incomplete cells, the feature is excluded if the rate of missing data is above 95%. In case if it remains below a pseudo-feature is re-defined which codes the empty cells as 1, the full cells as 0. Among the remaining columns, we remove the missing observations. For categorical columns with a finite set of values, we create factors to express categorical data. Here, the factors constitute important class for statistical analysis and plots. After this step, the data set becomes a collection of factors and numerical data.

To improve the prediction performance of the classifiers and provide faster and more cost-effective classifiers, feature selection is also applied for these datasets by using a filter method and Recursive Feature Elimination method successively. This step is also important as it gives an insight into what could be the most important ratios used to construct a risk evaluation model which could be applied in real process.

Relying on the characteristics of data, filter models evaluate features without utilizing any classification algorithms (Liu & Motoda, 2007). Fisher Score (Gu et al., 2011) is a supervised feature selection algorithm and has been widely applied to many applications. It is designed to select features whose values are more uniformly distributed for samples in the same class but more dissimilar for samples in different classes. In this work, we obtain the Fisher score and use a threshold to determine the valuable or meaningful variables. Then, models are trained using full set of features based on Recursive Feature Elimination (RFE) method. The process continues recursively until desired number of features are selected. RFE is based on the idea of eliminating recursively features that less important in terms of contribution to the prediction of target variable. The resulting analysis related to the data cleaning and feature selection is presented in Tables B1–B5 of Appendix B, and accordingly the certain characteristics of the datasets are summarized in Table 1.

Even both datasets are related to credit defaults, they have diversity in some aspects such as sample size, number of variables and imbalance ratio. Home credit dataset surpasses the other datasets in terms of (i) number of (categorical) variables which enables decision maker to comprehend the influence of wide range of factors, (ii) the size of the loans which are considerably large amounts and extended to longer time durations. Bank Fear dataset surpasses the other datasets in terms of sample size which is taken as an advantage to attain the law of large numbers. On the other hand, GMSC dataset differs in regard to the higher proportion of non-default cases with an imbalance ratio of 15.56. Therefore, it is expected to show how CBFC method contributes to the improvement on the accuracy of prediction on credit risks for the loans accredited for different causes.

3.3. Model performances

To assess how good or how accurate a classifier is at predicting the class labels of instances, the Accuracy Ratio (AR), sensitivity and specificity rates, Area Under Curve (AUC) ratio, H measure and Brier score are commonly used. The AR expresses the ratio of the number of truly predicted observations to the total number of observations. The other indicators, specificity and sensitivity explain the ratio of correctly predictions. The sensitivity indicates the ratio of correctly classified observations by total number of observations in Class 1 (observations with default). On the other hand, the specificity is defined as the proportion of number of truly detected Class 0 (observations without default) to the total number of cases in Class 0. Additional to those, for a given model the trade-off between the true positive rate (TPR) and the false positive rate (FPR) is determined by ROC curve and the area under the ROC curve which measures the accuracy of the model (AUC ratio) (Han et al., 2011).

The H measure is a threshold-varying evaluation metric based on expected minimum misclassification loss. It is proposed by Hand (2009) to overcome some deficiencies of AUC. The H measure assumes that the distribution of misclassification costs should depend on the classification problem (Hand & Anagnostopoulos, 2013). It uses a pre-specified Beta distribution to represent the misclassification cost distributions functions. In this study, we adopt the Beta distribution with parameters $\alpha = \beta = 2$ to indicate the relative severities of classification errors to be consistent across classifiers.

The Brier score is an indicator used to evaluate performance of the classifier based on the prediction of the posterior probability. It is obtained by calculating the mean squared loss between the true label and the probability prediction. The Brier score is expressed as follows

Table 1
Brief description of the data sets (number of data items, features and classes).

Name	#Samples	#Features	#Classes [Distribution]	Imbalance Ratio
German credit data	1000	8	2 [700/300]	2.33
Home credit data	70,000	30	2 [56000/14000]	4.00
Bank Fear data	454,783	26	2 [365888/88895]	4.12
PPDai data	55,596	16	2 [48413/7183]	6.74
GMSC data	110,437	9	2 [103767/6670]	15.56

$$Brierscore = \frac{1}{n}\sum_{i=1}^{n}(\widehat{P}_i - y_i)^2 \quad (7)$$

where n denotes the number of samples. y_i and $\widehat{P}_i$ represent the true label and the prediction of the posterior probability of i-th sample, respectively (Xia et al., 2020b).

ML algorithms are designed and applied to two sets of data partitioned randomly as train and test. The selection criteria and distribution of data set within these groups is important as the model obtained using the data in train part is the candidate model to employ for prediction if its performance on the test data is satisfactory enough. The training and testing partition of the data are randomly assigned to 80% and 20%, respectively, based on the commonly recommended partition in similar studies. We generate five different testing and training data partitions of each credit dataset, and calculate the average performances for each benchmark model accordingly.

3.4. Model settings

The computational parts of the analyses are performed using R programing language. The implementation of FKM algorithm is carried out with the package "fclust" (Ferraro et al., 2019). For FKM method, number of clusters is set to 5 for the German credit dataset, 10 for the other datasets. Additionally, the default value of 2 is given to degree of fuzziness.

For the supervised classification models RF, DT, GNB, and SVM, we use the packages "randomForest" (Breiman, 2001), "rpart" (Breiman, 1984), naivebayes" (Majka, 2020), "e1071" (Meyer et al., 2019), respectively. Also, for the model fitting and the algorithms ANN, kNN, and LR, the "caret" (Kuhn, 2008) package is employed.

The hyper-parameters are found based on randomized grid search in caret package for ML algorithms except for ensemble algorithms, and fixed to the random set combination to 20 for algorithms DT and kNN, and 2 for the GNB. Different values for a hyper-parameter of the algorithm are searched and the best combination is searched in this process. The optimum parameter values are determined with 5-fold cross-validation of 10-repeats using grid-search and the validation metric is chosen to be ROC value. For kNN, the number of neighbors is the tuning parameter, and its search space is {2, 5, …, 43} (20 values). The minimum vote for definite decision is kept as default value of 1. The number of trees is the hyper-parameter in RF, and the best value for this parameter is found via a "for loop" based on ROC ranging between the values {5,10,…,40}. We use the square root of the number of features for number of variables sampled at every split, and the minimum sample in each node is kept as 1. In GNB, the kernel parameter is used for the tuning, {False and True}. In the algorithm, the value is let 0 for the Laplace smoothing. To find the best model, the function randomly tries 20 different values within the range [3.21e-4, 6.76e-3] for the complexity parameter related to the pruning mechanism in DT. The other parameters, such as minimum split, maximum competitor split, and maximum depth, are set as values 20, 4, and 15, respectively. For ANN, we set the number of neurons in the hidden layer as 10, and 0.5 for the regularization parameter. The maximum iteration number is let as 100 same as the default setting. For SVM, we allow the epsilon value as to be 0.1 for the intensive loss function same as in the default model. Only the linear and radial bases kernels are employed. For the radial based kernel SVM, we keep the gamma parameter to control how much the decision boundaries fit the data as the default value of 1.

For gradient boosting models XGB, Light GBM, and CatBoost, the libraries "xgboost" (Chen, and Guestrin, 2016), "lightgbm" (Ke et al., 2017) and "catboost" (Veronika Dorogush et al., 2018), respectively, are utilized. We optimize the number of boosted trees between the range [5,40] in these algorithms by using a for loop with ROC validation metric same as the process in RF. Other parameters related to the algorithms are demonstrated in Table 2.

Table 2
An expression of hyper-parameters and searching space.

Algorithm	Hyper-parameter	Description	Searching space
CatBoost	No. of iterations	The number of trees in CatBoost	[5,40]
	Maximum depth	Maximum depth of a single tree	3
LightGBM	No. of estimators	The number of trees in LightGBM	[5,40]
	Maximum depth	Maximum depth of a single tree	6
	Learning rate	It shrinks the contribution of each tree	0.3
RF	No. of estimators	The number of trees in the forest.	[5,40]
	Maximum depth	Maximum depth of a single tree	6
XGB	No. of estimators	The number of trees in XGBoost	[5,40]
	Maximum depth	Maximum depth of a single tree	6
	Subsampling rate	The fraction of samples used for training a single tree	1
	Learning rate	It shrinks the contribution of each tree	0.3
	Column sampling rate	The fraction of features used for training a single tree	1
	Gamma	Minimum loss reduction for further split	1
DT	Complexity parameter	Any split that does not decrease the overall lack of fit by a factor of cp is not attempted	[3.21e-4, 6.76e-3]
	Maximum depth	Maximum depth of the tree	15
ANN	No. of hidden units	The number of neurons in hidden layer	10
	Decay	Regularization parameter	0.5
SVM	kernel function	Function for transforming the data to higher number of dimension spaces	{Radial, Linear}
	gamma	Controls the influence of a single training sample on hyperplane	1
kNN	k	Number of neighbors	[2,43]
GNB	Kernel	Usage of conditional densities for class prediction	{TRUE, FALSE}

3.5. Discussion

In this subsection, the predictive results of the 22 classifiers across the five credit scoring datasets are presented in terms of the six

Table 3
Performance comparison of different models for German credit dataset.

Models	Accuracy	Brier score	H measure	AUC	Sensitivity	Specificity	AvgR
LR	**0.7440**	0.2449	0.2530	0.7534	0.4400	0.8814	8.00
CBFC-LR	**0.7490**	0.1810	0.2692	**0.7580**	**0.5200**	0.8400	**5.00**
SVM_Linear	0.7390	0.2330	**0.2842**	**0.7725**	0.3300	0.9143	7.75
CBFC-SVM_Linear	0.7360	**0.1744**	**0.2705**	0.7539	**0.5600**	0.8114	**6.17**
SVM_RBF	**0.7430**	0.2384	**0.2909**	**0.7738**	0.3767	0.9000	6.83
CBFC-SVM_RBF	0.7390	**0.1756**	0.2451	0.7438	**0.5367**	0.8257	7.92
ANN	0.7410	0.2332	0.2311	0.7193	0.3533	0.9071	11.50
CBFC-ANN	0.7400	**0.1768**	0.2579	0.7476	0.4533	0.8629	**6.33**
NB	0.6960	0.2784	0.1966	0.7209	0.4100	0.8186	17.42
CBFC-NB	0.7210	0.2166	0.1900	0.7060	0.4100	0.8543	14.25
RF	0.7030	0.2672	0.2245	0.7345	0.1167	**0.9543**	15.17
CBFC-RF	0.7280	0.1988	0.2045	0.7209	0.4600	0.8429	12.00
DT	0.7230	0.2539	0.1803	0.7143	0.3667	0.8757	15.83
CBFC-DT	0.7260	0.1973	0.1587	0.6753	0.4367	0.8500	13.83
kNN	0.6270	0.2483	0.0224	0.5593	0.2867	0.7729	20.50
CBFC-kNN	0.6230	0.2453	0.0238	0.5715	0.3500	0.7400	20.00
XGB	0.7250	0.2222	0.2460	0.7463	0.2567	0.9257	10.67
CBFC-XGB	0.7300	0.1815	0.2595	0.7411	0.4767	0.8386	8.67
CatBoost	0.7160	0.2241	0.2365	0.7461	0.2100	**0.9329**	12.00
CBFC-CatBoost	0.7270	0.1809	0.2412	0.7346	0.4800	0.8329	9.83
LGBM	0.7110	0.2188	0.2294	0.7445	0.1267	**0.9614**	12.17
CBFC-LGBM	0.7290	0.1844	0.2145	0.7298	0.5000	0.8271	11.17

Table 4
Performance comparison of different models for Home credit dataset.

Models	Accuracy	Brier score	H measure	AUC	Sensitivity	Specificity	AvgR
LR	**0.8108**	0.1790	0.2103	0.7508	0.1640	0.9726	7.67
CBFC-LR	**0.8103**	**0.1373**	**0.2123**	**0.7520**	0.1750	0.9691	**4.33**
SVM_Linear	0.7997	0.1619	0.0296	0.5877	0.0004	**0.9996**	15.00
CBFC-SVM_Linear	0.7999	0.1574	0.0332	0.5921	0.0001	**0.9999**	14.33
SVM_RBF	0.8081	0.1751	0.1542	0.6801	0.1323	0.9770	10.83
CBFC-SVM_RBF	0.7844	0.1463	0.1452	0.6898	**0.3339**	0.8970	11.83
ANN	0.8006	0.1677	0.1031	0.6660	0.0049	0.9995	13.33
CBFC-ANN	0.8021	0.1491	0.1180	0.6812	0.0419	0.9922	11.50
NB	0.8000	0.1996	0.1832	0.7314	0.0000	**1.0000**	12.67
CBFC-NB	0.8023	0.1986	0.1098	0.6646	0.0538	0.9895	14.00
RF	0.8008	0.1989	0.0794	0.5819	0.0256	0.9946	15.67
CBFC-RF	0.7691	0.1456	0.1554	0.7090	**0.3902**	0.8638	11.50
DT	0.8068	0.1700	0.1080	0.6341	0.1118	0.9806	12.50
CBFC-DT	0.8052	0.1478	0.1219	0.6708	0.1314	0.9736	11.00
kNN	0.7767	0.2438	0.0044	0.5290	0.0619	0.9554	19.50
CBFC-kNN	0.7634	0.1797	0.0113	0.5531	0.1041	0.9282	18.83
XGB	0.8086	0.1729	0.2028	0.7455	0.1034	0.9848	9.00
CBFC-XGB	0.8078	0.1376	0.2101	0.7494	**0.2508**	0.9470	7.17
CatBoost	0.8005	0.1907	0.1665	0.7202	0.2111	0.9479	12.50
CBFC-CatBoost	**0.8094**	**0.1375**	**0.2112**	**0.7515**	0.2189	0.9571	**5.00**
LGBM	0.8085	0.1720	0.2052	0.7466	0.0938	0.9872	8.67
CBFC-LGBM	0.8075	**0.1375**	**0.2111**	**0.7509**	0.2442	0.9484	**6.17**

performance measures. Tables 3–7 provide the average performances of each benchmark model over five different testing and training data partitions of each credit dataset. The three best classifiers referring to the order of ranking for each evaluation metric are typed in bold characters. All the experiment is accomplished with R 4.1.2 on a PC with 2.10 GHz Intel Xeon E5-2620 CPU, 32 GB RAM, and a Nvidia GTX 1060 3 GB GPU.

The outcomes of the proposed approach expose promising results which can be taken as improvement in the traditional ML models. Five main conclusions can be listed as follows: (i) The CBFC generally results in recognizable improvement on the considered metrics compared to traditional ML alternatives. (ii) The sensitivity of the CBFC model is superior to other classifiers for both datasets. The CBFC-SVM Linear in the German and PPDai datasets; CBFC-RF in the Home credit and GMSC datasets, and CBFC-CatBoost in the Bank Fear dataset yield high sensitivity results. (iii) The Brier score of CBFC perform better than other classifiers generally, approving the advantages of CBFC method over others. According to the Brier scores, CBFC-SVM Linear for the German and PPDai; CBFC-LR for the Home credit and Bank Fear, CBFC-ANN for the GMSC rank as the first. (iv) The H measure of the proposed method also produce the promising classification results, so SVM-RBF, CBFC-LR, CBFC-CatBoost, CBFC-LGBM, and CBFC-ANN take the first place in the German, Home credit, Bank Fear, PPDai and GMSC data sets, respectively. (v) Specificity comes out to be generally much better than sensitivity whose values are marked in bold face. The main reason is that it is more difficult to classify bad customers from all credit applicants due to the complexity of credit risk.

To determine the benchmarks, classifier performances across

Table 5
Performance comparison of different models for Bank Fear dataset.

Models	Accuracy	Brier score	H measure	AUC	Sensitivity	Specificity	AvgR
LR	0.8041	0.1424	0.1711	0.7245	0.1085	0.9775	10.00
CBFC-LR	**0.8138**	**0.1121**	0.1839	0.7359	0.1209	0.9789	**6.00**
SVM_Linear	0.7945	0.1745	0.0023	0.5212	0.0120	0.9901	17.83
CBFC-SVM_Linear	0.7848	0.1543	0.0198	0.5625	0.0300	0.9735	17.33
SVM_RBF	0.7990	0.1596	0.0621	0.6212	0.0183	0.9942	14.50
CBFC-SVM_RBF	0.7995	0.1407	0.1752	0.7164	**0.3267**	0.9177	11.17
ANN	0.8001	0.1600	0.0034	0.5008	0.0007	**0.9999**	16.17
CBFC-ANN	0.7998	**0.1312**	0.0111	0.5194	0.0397	0.9899	13.50
NB	0.7997	0.2002	0.0481	0.6182	0.0000	**0.9997**	16.17
CBFC-NB	0.8097	0.1901	0.0746	0.6456	0.0125	0.9994	12.00
RF	0.8002	0.1997	0.0023	0.5023	0.0017	**0.9998**	16.67
CBFC-RF	**0.8177**	0.1473	0.1318	0.6965	**0.3277**	0.9177	9.00
DT	0.8047	0.1534	0.0678	0.6146	0.0875	0.9840	13.00
CBFC-DT	0.8099	0.1484	0.1076	0.6560	0.1243	0.9814	9.33
kNN	0.8055	0.1547	0.0571	0.5993	0.0720	0.9888	13.17
CBFC-kNN	0.8072	0.1490	0.0813	0.6351	0.0903	0.9857	11.00
XGB	0.8023	0.1656	0.1575	0.7275	0.1033	0.9591	12.33
CBFC-XGB	0.8058	0.1389	**0.2162**	**0.7567**	0.2510	0.9445	6.67
CatBoost	0.8074	0.1383	**0.2109**	**0.7528**	0.0911	0.9865	**6.00**
CBFC-CatBoost	**0.8113**	**0.1328**	**0.2474**	**0.7719**	**0.3489**	0.9269	**4.83**
LGBM	0.8036	0.1409	0.1956	0.7444	0.0441	0.9934	8.50
CBFC-LGBM	0.8054	0.1393	0.2077	0.7516	0.2085	0.9546	7.83

Table 6
Performance comparison of different models for PPDai dataset.

Models	Accuracy	Brier score	H measure	AUC	Sensitivity	Specificity	AvgR
LR	**0.8709**	0.1084	0.0782	0.6215	0.0127	0.9983	7.50
CBFC-LR	0.8705	0.1090	0.0753	0.6176	0.0175	0.9971	9.42
SVM_Linear	0.8708	0.1217	0.0172	0.5431	0.0000	**1.0000**	12.25
CBFC-SVM_Linear	0.7817	**0.0528**	0.0036	0.5032	**0.1141**	0.8807	14.67
SVM_RBF	0.8707	0.1229	0.0034	0.5061	0.0001	0.9999	15.08
CBFC-SVM_RBF	**0.8716**	0.1179	0.0366	0.5745	0.0167	0.9985	9.50
ANN	0.8708	0.1121	0.0122	0.5292	0.0000	**1.0000**	12.42
CBFC-ANN	0.8706	0.1119	0.0175	0.5540	0.0004	0.9998	11.50
NB	0.8705	0.1289	0.0161	0.5536	0.0019	0.9994	13.42
CBFC-NB	0.8699	0.1293	0.0158	0.5524	0.0043	0.9984	14.83
RF	0.8708	0.1291	0.0034	0.5044	0.0000	**1.0000**	15.25
CBFC-RF	0.8696	0.1286	0.0071	0.5117	0.0060	0.9978	16.00
DT	0.8700	0.1103	0.0468	0.5757	0.0225	0.9958	11.08
CBFC-DT	0.8696	0.1108	0.0524	0.5876	0.0376	0.9931	11.33
kNN	0.8603	0.1636	0.0039	0.5174	0.0316	0.9833	17.67
CBFC-kNN	0.8402	0.1460	0.0054	0.5209	**0.0685**	0.9547	16.67
XGB	0.8708	0.1084	0.0854	**0.6396**	0.0014	0.9998	**6.17**
CBFC-XGB	0.8688	0.1079	**0.0873**	**0.6382**	**0.0649**	0.9881	8.50
CatBoost	0.8708	0.1093	0.0703	0.6247	0.0001	**1.0000**	7.83
CBFC-CatBoost	0.8704	**0.1078**	**0.0879**	**0.6404**	0.0415	0.9935	**6.67**
LGBM	**0.8710**	0.1135	0.0785	0.6244	0.0032	0.9998	8.00
CBFC-LGBM	0.8700	**0.1076**	**0.0883**	0.6343	0.0527	0.9913	**7.25**

datasets and accuracy indicators are ranked. Based on the approach by Lessmann et al. (2015), each classifier is labelled by ranks at which the best (marked as one) to the poorest (marked as twenty-two) to identify the performance of each classifier with respect to various evaluation measures. The average (AvgR) of each evaluation measure for each classifier is taken into account. Six evaluation measures constitute the base for the rank statistics approach to reflect the comprehensive performance of the models. The best classifiers (lowest average rank) for each measure are also presented in boldface in Tables 3–7. Considering the top three methods according to the ranking results into consideration, it is noteworthy to state that CBFC-LR, CBFC-CatBoost and CBFC-LGBM yield the best comprehensive AvgR ranking values three times out of the five datasets, which is taken as an indication of its excellent performance.

To depict more elaborately, the performance ranking with respect to ML methods with and without CBFC approach over five datasets are summarized in Fig. 2. It can be seen that the mean AvgR values support the adequacy of CBFC implementation to improve the performance of ML methods. LR under CBFC is ranked as the best performing model followed closely by CBFC-XGB, CBFC-CatBoost, CBFC-LGBM. On the other hand, majority of the ML models which remain in the last rankings are mostly dominated by the ML methods without CBFC approach. Additionally, kNN and CBFC-kNN do perform insufficiently in five datasets.

Table 7
Performance comparison of different models for GMSC dataset.

Models	Accuracy	Brier score	H measure	AUC	Sensitivity	Specificity	AvgR
LR	0.9403	0.0510	0.2659	0.7855	0.0393	**0.9978**	11.58
CBFC-LR	0.9394	0.0509	0.2779	0.7938	0.1036	0.9927	11.67
SVM_Linear	0.9400	0.0576	0.2441	0.6470	0.0071	**0.9995**	14.33
CBFC-SVM_Linear	**0.9424**	0.0564	0.3342	0.7788	0.0869	0.9970	10.00
SVM_RBF	0.9419	0.0546	0.2506	0.6135	0.0452	**0.9991**	11.67
CBFC-SVM_RBF	0.9416	0.0528	0.2558	0.6537	0.1512	0.9921	11.00
ANN	0.9412	**0.0468**	**0.3824**	**0.8423**	0.0883	0.9957	6.67
CBFC-ANN	**0.9423**	**0.0464**	**0.3958**	**0.8459**	0.1289	0.9942	**4.00**
NB	0.9401	0.0588	0.2317	0.7360	0.0405	0.9975	14.33
CBFC-NB	0.9403	0.0543	0.3710	0.8285	0.1238	0.9924	9.92
RF	0.9395	0.0552	0.2018	0.6367	0.1357	0.9908	15.33
CBFC-RF	0.9348	0.0480	0.3357	0.8056	**0.2952**	0.9756	11.17
DT	0.9407	0.0599	0.2096	0.6466	0.1683	0.9900	14.67
CBFC-DT	0.9417	0.0584	0.2297	0.6711	**0.2144**	0.9881	13.00
kNN	0.9308	0.0808	0.0287	0.5603	0.0087	0.9896	21.00
CBFC-kNN	0.9311	0.0704	0.0329	0.5808	0.0423	0.9878	20.17
XGB	0.9411	0.0470	0.3738	0.8285	0.0988	0.9949	8.00
CBFC-XGB	**0.9423**	0.0469	0.3746	0.8383	0.1726	0.9914	**5.50**
CatBoost	0.9365	0.0674	0.2421	0.8242	0.1038	0.9897	15.17
CBFC-CatBoost	0.9227	0.0493	0.3425	0.8342	**0.2348**	0.9666	11.17
LGBM	0.9418	0.0473	0.3618	0.8289	0.0893	0.9962	7.33
CBFC-LGBM	0.9415	**0.0465**	**0.3927**	**0.8503**	0.1524	0.9919	**5.33**

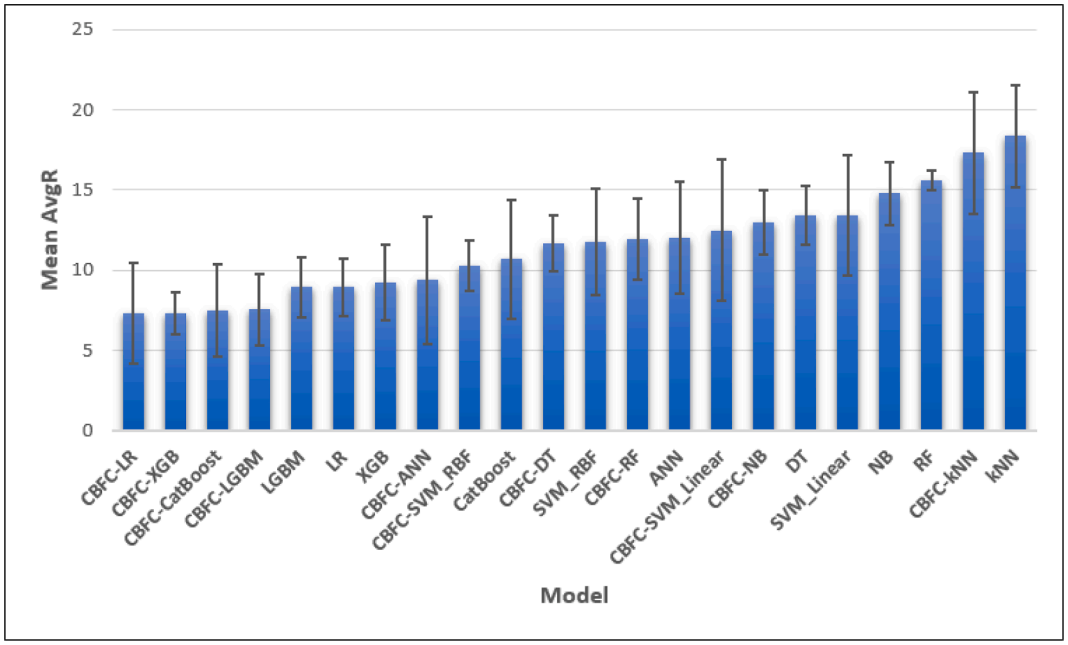


Fig. 2. The performance rankings over five credit datasets.

Table 8
The average ranks of the CBFC and traditional ML methods over all datasets and significance tests.

Method	Accuracy	Brier score	H measure	AUC	Sensitivity	Specificity
CBFC	**11.34**	**8.40**	**10.24**	**10.56**	**7.81**	14.78
Traditional ML	11.65	14.60	12.76	12.44	15.19	**8.22**
Wilcoxon Signed Rank Test (p value)	0.635	0.003*	0.003*	0.008*	0.003*	0.004*

* $p < 0.05$.

Even though proposed CBFC models show superiority as stated, their statistical significances compared to benchmark models are done using Wilcoxon matched-pairs signed rank test. This nonparametric significance test determines if the population median ranks of two related samples differ. To achieve this, the performance, which is assessed by the average value of different testing dataset partitions for a certain evaluation measure, is used to rank different models. Then, for each of five datasets, the average ranks of different models based on a specific evaluation metric are determined to detect if there exists difference between the average ranks of CBFC model and traditional ML alternatives differ significantly.

The average ranks of the CBFC and traditional ML methods over all datasets are shown in Table 8, which are based on six evaluation measures. Table 8 presents some important findings. First, the superiority of

the proposed CBFC method is demonstrated since it achieves the lowest average rank (highlighted in bold type) over the traditional ML method for the accuracy, Brier score, H measure, AUC, and sensitivity. Second, according to the results of Wilcoxon matched-pairs signed rank test there exist statistically significant differences between the CBFC and traditional ML methods based on the Brier score, H measure, AUC and sensitivity (p value < 0.05), so CBFC ranks first indicating its superior performance. Third, there is also statistically significant difference in the group of model performances based on the specificity (p value < 0.05) which indicates traditional ML methods have higher specificity values.

To sum up, it can be stated that, the performance comparisons illustrate that the proposed CBFC method yields high performance as CBFC can find optimal model parameters efficiently. Besides, CBFC implements an efficient parameter search algorithm that can effectively handle the computational complexity for the instances in large datasets. In order to investigate the computational cost of CBFC method for training, the computation times required to implement each dataset are recorded and presented in Table C1 of Appendix C. The execution times of CBFC methods with some learning algorithms seem to be higher compared to the execution times with traditional ML. However, when the dataset is large in terms of the number of observations, use of CBFC is generally preferable over traditional ML due to the computational time requirements.

4. Concluding comments

Credit scoring is crucial for financial industry, and more sophisticated techniques and tools for predicting credit worthiness of loan applicants are always needed. The common practice in the literature is to develop a single credit score model from the historical available information to predict the likelihood of default of new applicants for making loan decisions. Using traditional ML algorithms on credit data may lead misleading conclusions as the data composition and the decision criteria of "default" and "non-default" cases. Therefore, in this paper, we propose the strategy of integrating fuzzy clustering with supervised learning to construct credit scoring models. In this context, borrower segmentation is performed with the help of FKM clustering, and a separate fuzzy credit scoring model is developed for each cluster. The process for the CBFC approach with different training algorithms is investigated to identify the heterogeneity of system inputs that depend on borrowers. The main advantage of CBFC method is that fuzzy membership is adopted to indicate different contribution of each input point to the learning of classification model.

In the experiment, five well-known open-sourced datasets are chosen. These datasets show different characteristics in accordance with sample size, rate of "default", number of categorical features, total number of features which allow us to detect the power of ML methods and proposed fuzzy modification on those algorithms. So, the predictive results of benchmark models across these datasets are presented.

CBFC in comparison to traditional ML models shows superiority in terms of its influence on the performance improvement. It is shown that CBFC-LR, CBFC-CatBoost and CBFC-LGBM are found to be the best regardless of the characteristics of the datasets utilized in this paper. In credit scoring, CBFC method improves the performance in comparison with traditional ML alternatives. It is also noticed that the accuracy value provides similar results, yet the performance on sensitivity, Brier score, H measure and AUC values exhibit highlighting patterns. This elaborates the necessity of evaluating models also from different aspects, such as label, probability, and discriminatory ability. Additionally, among the traditional ML methods, LR and XGB, CatBoost, LGBM present convincing results which also explains their role as benchmark in the existing literature. kNN and RF are found to be the least preferable among traditional ML due to its unfitting structure to credit data.

Due to its superiority with respect to performance measures, CBFC approach can be reliably implemented in practice to determine credit risk, as the fuzzy modification proposed in this paper can be used to predict "default" cases with high accuracy performance. Even though it is possible to obtain different results with different sets of features, the contribution is evident enough to demonstrate the usefulness of CBFC method in the assessment of credit risk.

CRediT authorship contribution statement

Furkan Baser: Conceptualization, Methodology, Investigation, Validation, Visualization, Writing – original draft, Writing – review & editing. **Oguz Koc:** Conceptualization, Methodology, Software, Visualization, Writing – original draft. **A. Sevtap Selcuk-Kestel:** Conceptualization, Methodology, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to my data.

Appendix A. Machine learning methods

Preliminaries on ML methods are presented in the electronic supplementary material related to this article can be found online at https://doi.org/10.1016/j.eswa.2023.119882.

Appendix B. The results of analysis related to the feature selection

Selected features and their characteristics for each dataset are presented in the electronic supplementary material related to this article can be found online at https://doi.org/10.1016/j.eswa.2023.119882.

Appendix C. The computational cost of CBFC method for training

Computation times to train different models for datasets are presented in the electronic supplementary material related to this article can be found online at https://doi.org/10.1016/j.eswa.2023.119882.

References

Alpaydin, E. (2016). *Machine learning: The new AI*. MIT Press.

Anderson, R. (2007). *The credit scoring toolkit: Theory and practice for retail credit risk management and decision automation*. Oxford University Press.

Bai, C. G., Shi, B. F., Liu, F., & Sarkis, J. (2019). Banking credit worthiness: Evaluating the complex relationships. *Omega, 83*, 26–38. https://doi.org/10.1016/j.omega.2018.02.001

Bao, W., Lianju, N., & Yue, K. (2019). Integration of unsupervised and supervised machine learning algorithms for credit risk assessment. *Expert Systems with Applications, 128*, 301–315. https://doi.org/10.1016/j.eswa.2019.02.033

Baser, F., & Demirhan, H. (2017). A fuzzy regression with support vector machine approach to the estimation of horizontal global solar radiation. *Energy, 123*, 229–240. https://doi.org/10.1016/j.energy.2017.02.008

Bezdek, J. C. (1981). Objective function clustering. In *Pattern recognition with fuzzy objective function algorithms* (pp. 43-93). Springer, Boston, MA.

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences, 10*(2–3), 191–203.

Boughaci, D., Alkhawaldeh, A. A., Jaber, J. J., & Hamadneh, N. (2021). Classification with segmentation for credit scoring and bankruptcy prediction. *Empirical Economics, 61*(3), 1281–1309. https://doi.org/10.1007/s00181-020-01901-8

Breiman, L. (1984). *Classification and regression trees*. Belmont, California, USA: Wadsworth International Group.

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32.

Çelikyılmaz, A., & Türkşen, I. B. (2007). Fuzzy functions with support vector machines. *Information Sciences, 177*(23), 5163–5177. https://doi.org/10.1016/j.ins.2007.06.022

Celikyilmaz, A., & Turksen, I. B. (2008a). Enhanced fuzzy system models with improved fuzzy clustering algorithm. *IEEE Transactions on Fuzzy Systems, 16*(3), 779–794. https://doi.org/10.1109/TFUZZ.2007.905919

Celikyilmaz, A., & Turksen, I. B. (2008b). Uncertainty modeling with evolutionary improved fuzzy functions approach. *IEEE Systems, Man, and Cybernetics-Part B, 38*(4), 1098–1110. https://doi.org/10.1109/TSMCB.2008.924587

Çelikyılmaz, A., & Türkşen, İ. B. (2009). Modeling uncertainty with fuzzy logic with recent theory and applications introduction. *Modeling Uncertainty With Fuzzy Logic: With Recent Theory And Applications*. Springer-Verlag Berlin.

Chakravarty, S., Demirhan, H., & Baser, F. (2020). Fuzzy regression functions with a noise cluster and the impact of outliers on mainstream machine learning methods in the regression setting. *Applied Soft Computing, 96*, Article 106535. https://doi.org/10.1016/j.asoc.2020.106535

Chang, Y. C., Chang, K. H., & Huang, Y. H. (2020). A novel fuzzy credit risk assessment decision support system based on the python web framework. *Journal of Industrial and Production Engineering, 37*(5), 229–244. https://doi.org/10.1080/21681015.2020.1772385

Chen, S., Guo, Z., & Zhao, X. (2021). Predicting mortgage early delinquency with machine learning methods. *European Journal of Operational Research, 290*(1), 358–372. https://doi.org/10.1016/j.ejor.2020.07.058

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*, KDD '16, ACM, New York, NY, USA, ISBN: 978-1-4503-4232-2, pp. 785–794, https://doi.org/10.1145/2939672.2939785.

Correa, A., Gonzalez, A., Nieto, C., & Amezquita, D. (2012). Constructing a credit risk scorecard using predictive clusters. *SAS Global Forum, 128*.

Crook, J. N. (1996). Credit scoring: An overview (Working paper series No. 96/13). *British Association, Festival of Science, University of Birmingham and the University of Edinburgh*.

Feng, X., Xiao, Z., Zhong, B., Dong, Y., & Qiu, J. (2019). Dynamic weighted ensemble classification for credit scoring using Markov Chain. *Applied Intelligence, 49*(2), 555–568. https://doi.org/10.1007/s10489-018-1253-8

Ferraro, M. B., Giordani, P., & Serafini, A. (2019). fclust: An R package for fuzzy clustering. *The R Journal, 11*(1).

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2011). A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42*(4), 463–484. https://doi.org/10.1109/TSMCC.2011.2161285

García-Laencina, P. J., Sancho-Gómez, J. L., & Figueiras-Vidal, A. R. (2010). Pattern classification with missing data: A review. *Neural Computing and Applications, 19*(2), 263–282. https://doi.org/10.1007/s00521-009-0295-6

Gatzert, N., & Wesker, H. (2012). A comparative assessment of Basel II/III and Solvency II. *The Geneva Papers on Risk and Insurance-Issues and Practice, 37*(3), 539–570.

Ghanbari, S., Pashazadeh, S., & Bevrani, H. (2014). Credit risk prediction using clustered classification. *International Journal of Artificial Intelligence and Mechatronics, 3*(5), 247–253.

Golbayani, P., Florescu, I., & Chatterjee, R. (2020). A comparative study of forecasting corporate credit ratings using neural networks, support vector machines, and decision trees. *The North American Journal of Economics and Finance, 54*, Article 101251. https://doi.org/10.1016/j.najef.2020.101251

Gu, Q., Li, Z., & Han, J. (2011). Generalized fisher score for feature selection. *Proceedings of the International Conference on Uncertainty in Artificial Intelligence*.

Gunnarsson, B. R., Vanden Broucke, S., Baesens, B., Óskarsdóttir, M., & Lemahieu, W. (2021). Deep learning for credit scoring: Do or don't? *European Journal of Operational Research, 295*(1), 292–305. https://doi.org/10.1016/j.ejor.2021.03.006

Han, J., Pei, J., & Tong, H. (2011). *Data mining: Concepts and techniques*. Morgan Kaufmann.

Hand, D. J., & Henley, W. E. (1997). Statistical classification methods in consumer credit scoring: A review. *Journal of the Royal Statistical Society: Series A (Statistics in Society), 160*(3), 523–541.

Hand, D. J. (2009). Measuring classifier performance: A coherent alternative to the area under the ROC curve. *Machine Learning, 77*, 103–123.

Hand, D. J., & Anagnostopoulos, C. (2013). When is the area under the receiver operating characteristic curve an appropriate measure of classifier performance? *Pattern Recognition Letters, 34*(5), 492–495. https://doi.org/10.1016/j.patrec.2012.12.004

Harris, T. (2015). Credit scoring using the clustered support vector machine. *Expert Systems with Applications, 42*(2), 741–750. https://doi.org/10.1016/j.eswa.2014.08.029

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction* (Vol. 2,, 1–758.

He, H., Zhang, W., & Zhang, S. (2018). A novel ensemble method for credit scoring: Adaption of different imbalance ratios. *Expert Systems with Applications, 98*, 105–117. https://doi.org/10.1016/j.eswa.2018.01.012

Junior, L. M., Nardini, F. M., Renso, C., Trani, R., & Macedo, J. A. (2020). A novel approach to define the local region of dynamic selection techniques in imbalanced credit scoring problems. *Expert Systems with Applications, 152*, Article 113351. https://doi.org/10.1016/j.eswa.2020.113351

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems 30* (pp. 3146–3154). Curran Associates Inc.

Kim, M., & Ramakrishna, R. S. (2005). New indices for cluster validity assessment. *Pattern Recognition Letters, 26*(15), 2353–2363. https://doi.org/10.1016/j.patrec.2005.04.007

Koc, O. (2019). *Comparison of machine learning algorithms on consumer credit classification*. Middle East Technical University). Master's thesis.

Kuhn, M. (2008). Building predictive models in R using the caret package. *Journal of Statistical Software, 28*, 1–26. https://doi.org/10.18637/jss.v028.i05.

Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research, 247*(1), 124–136. https://doi.org/10.1016/j.ejor.2015.05.030

Liang, L., & Cai, X. (2020). Forecasting peer-to-peer platform default rate with LSTM neural network. *Electronic Commerce Research and Applications, 43*, Article 100997. https://doi.org/10.1016/j.elerap.2020.100997

Lim, M. K., & Sohn, S. Y. (2007). Cluster-based dynamic scoring model. *Expert Systems with Applications, 32*(2), 427–431. https://doi.org/10.1016/j.eswa.2005.12.006

Liu, H., & Motoda, H. (Eds.). (2007). *Computational methods of feature selection*. CRC Press.

Liu, W., Fan, H., & Xia, M. (2022). Credit scoring based on tree-enhanced gradient boosting decision trees. *Expert Systems with Applications, 189*, Article 116034. https://doi.org/10.1016/j.eswa.2021.116034

Ma, X., Sha, J., Wang, D., Yu, Y., Yang, Q., & Niu, X. (2018). Study on a prediction of P2P network loan default based on the machine learning LightGBM and XGboost algorithms according to different high dimensional data cleaning. *Electronic Commerce Research and Applications, 31*, 24–39. https://doi.org/10.1016/j.elerap.2018.08.002

Majka, M. (2020). Package 'naivebayes'. Retrieved from https://cran.microsoft.com/web/packages/ naivebayes/naivebayes.pdf. Accessed August 15, 2022.

Malhotra, R., & Malhotra, D. K. (2002). Differentiating between good credits and bad cred-its using neuro-fuzzy systems. *European Journal of Operational Research, 136*(1), 190–211.

Marqués, A. I., García, V., & Sánchez, J. S. (2012). Exploring the behaviour of baseclassifiers in credit scoring ensembles. *Expert Systems with Applications, 39*(11), 10244–10250. https://doi.org/10.1016/j.eswa.2012.02.092

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C. C., ... & Meyer, M. D. (2019). Package 'e1071'. Retrieved from http://r.meteo.uni.wroc.pl/web/packages/e1071/e1071.pdf. Accessed August 15, 2022.

Mitchell, T. M., & Mitchell, T. M. (1997). *Machine learning* (Vol. 1, No. 9). New York: McGraw-hill.

Pal, N. R., & Bezdek, J. C. (1995). On cluster validity for the fuzzy c-means model. *IEEE Transactions on Fuzzy Systems, 3*(3), 370–379. https://doi.org/10.1109/91.413225

Ramkumar, M. (2016). A modified ANP and fuzzy inference system based approach for risk assessment of in-house and third party e-procurement systems. *Strategic Outsourcing: An International Journal, 9*(2), 159–188. https://doi.org/10.1108/SO-12-2015-0030

Scitovski, S., & Šarlija, N. (2014). Cluster analysis in retail segmentation for credit scoring. *Croatian Operational Research Review, 5*(2), 235–245.

Selcuk, M., Koc, O., & Kestel, A. S. (2022). The prediction power of machine learning on estimating the sepsis mortality in the intensive care unit. *Informatics in Medicine Unlocked, 28*, Article 100861. https://doi.org/10.1016/j.imu.2022.100861

Shi, J., & Xu, B. (2016). Credit scoring by fuzzy support vector machines with a novel membership function. *Journal of Risk and Financial Management, 9*(4), 13. https://doi.org/10.3390/jrfm9040013

Shieh, M. D., & Yang, C. C. (2008). Classification model for product form design using fuzzy support vector machines. *Computers & Industrial Engineering, 55*(1), 150–164. https://doi.org/10.1016/j.cie.2007.12.007

Sohn, S. Y., Kim, D. H., & Yoon, J. H. (2016). Technology credit scoring model with fuzzy logistic regression. *Applied Soft Computing, 43*, 150–158. https://doi.org/10.1016/j.asoc.2016.02.025

Stork, D. G., Duda, R. O., Hart, P. E., & Stork, D. (2001). *Pattern classification*. A Wiley-Interscience Publication.

Sun, Y., Chai, N., Dong, Y., & Shi, B. (2022). Assessing and predicting small industrial enterprises' credit ratings: A fuzzy decision-making approach. *International Journal of Forecasting, 38*(3), 1158–1172. https://doi.org/10.1016/j.ijforecast.2022.01.006

Syau, Y. R., Hsieh, H. T., & Lee, E. S. (2001). Fuzzy numbers in the credit rating of enterprise financial condition. *Review of Quantitative Finance and Accounting, 17*(4), 351–360.

Teles, G., Rodrigues, J. J., Saleem, K., Kozlov, S., & Rabêlo, R. A. (2020). Machine learning and decision support system on credit scoring. *Neural Computing and Applications, 32*(14), 9809–9826. https://doi.org/10.1007/s00521-019-04537-7

Türkşen, I. B. (1999). Type I and Type II fuzzy system modeling. *Fuzzy Sets and Systems, 106*(1), 11–34. https://doi.org/10.1016/S0165-0114(98)00354-6

Türkşen, I. B., & Celikyilmaz, A. (2006). Comparison of fuzzy functions with fuzzy rule base approaches. *International Journal of Fuzzy Systems, 8*(3), 137–149.

Veronika Dorogush, A., Ershov, V., & Gulin, A. (2018). CatBoost: gradient boosting with categorical features support. *arXiv e-prints*, arXiv-1810.

Xia, Y., Liu, C., Da, B., & Xie, F. (2018). A novel heterogeneous ensemble credit scoring model based on bstacking approach. *Expert Systems with Applications, 93*, 182–199. https://doi.org/10.1016/j.eswa.2017.10.022

Xia, Y., He, L., Li, Y., Liu, N., & Ding, Y. (2020a). Predicting loan default in peer-to-peer lending using narrative data. *Journal of Forecasting, 39*(2), 260–280. https://doi.org/10.1002/for.2625

Xia, Y., Zhao, J., He, L., Li, Y., & Niu, M. (2020b). A novel tree-based dynamic heterogeneous ensemble method for credit scoring. *Expert Systems with Applications, 159*, Article 113615. https://doi.org/10.1016/j.eswa.2020.113615

Xia, Y., He, L., Li, Y., Fu, Y., & Xu, Y. (2021). A dynamic credit scoring model based on survival gradient boosting decision tree approach. *Technological and Economic Development of Economy, 27*(1), 96–119. https://doi.org/10.3846/tede.2020.13997

Yu, L., Yao, X., Zhang, X., Yin, H., & Liu, J. (2020). A novel dual-weighted fuzzy proximal support vector machine with application to credit risk analysis. *International Review of Financial Analysis, 71*, Article 101577. https://doi.org/10.1016/j.irfa.2020.101577

Zhang, H., He, H., & Zhang, W. (2018). Classifier selection and clustering with fuzzy assignment in ensemble model for credit scoring. *Neurocomputing, 316*, 210–221. https://doi.org/10.1016/j.neucom.2018.07.070

Zhao, Z., Xu, S., Kang, B. H., Kabir, M. M. J., Liu, Y., & Wasinger, R. (2015). Investigation and improvement of multi-layer perceptron neural networks for credit scoring. *Expert Systems with Applications, 42*(7), 3508–3516. https://doi.org/10.1016/j.eswa.2014.12.006