Received: 11 November 2019 | Accepted: 18 June 2020

DOI: 10.1002/ijfe.2019

RESEARCH ARTICLE

A heterogeneous ensemble credit scoring model based on adaptive classifier selection: An application on imbalanced data

Tong Zhang | Guotai Chi

School of Economics and Management, Dalian University of Technology, Dalian, China

Correspondence
Tong Zhang, School of Economics and Management, Dalian University of Technology, Dalian 116024, China.
Email: zhangtong19900227@126.com

Funding information
The Key Programs of National Natural Science Foundation of China, Grant/Award Number: 71731003; The General Programs of National Natural Science Foundation of China, Grant/Award Numbers: 71873103, 71971034, 71971051; The Youth Programs of National Natural Science Foundation of China , Grant/Award Numbers: 71901055, 71903019; The Major Projects of National Social Science Foundation of China, Grant/Award Number: 18ZDA095

Abstract

In the domain of credit scoring, the number of bad clients is far less than that of good ones. So imbalanced data classification is a realisitc and critical issue in the credit scoring process. In this study, a novel heterogeneous ensemble credit scoring model is proposed for the problem of imbalanced data classification. This proposed model is on basis of five standard classifiers, namely LSVM, KNN, MDA, DT, LR, and adaptively selects the base classifiers with highest AUC according to the data distribution, then integrates all base classifiers to obtain a prediction. Finally, by using five comprehensive performance measures and four classical credit datasets, we find that the proposed model is better than other baseline models. This novel model can be applied to actual credit scoring and assist financial institutions in credit risk management.

KEYWORDS

binary classification, credit scoring, ensemble model, heterogeneous ensemble, imbalanced data

1 | INTRODUCTION

The great credit crisis of 2008 caused enormous losses all over the world, which is generally regarded as the most severe recession since the Great Depression beginning in 1929. This great credit crisis arose from large-scale defaulted sub-prime mortgage (Almunia, Benetrix, Eichengreen, O'Rourke, & Rua, 2010; Longstaff, 2010). Consequently, financial institutions are fully aware that it is of extreme necessity to introduce precise credit risk evaluation models in their risk management systems when approving loans to clients, which makes the credit risk evaluation become more and more important and crucial than ever before (Abelian & Castellano, 2017; Ala'raj & Abbod, 2016).

Credit scoring is the evaluation of the risk related to granting credit to a firm or a private person (Sadatrasoul, Gholamian, Siami, & Hajimohammadi, 2013), and it plays a critical part in recent financial affairs namely credit application screening, risk identification, post-lending monitoring and portfolio management (He, Zhang, & Zhang, 2018). Essentially, credit scoring is usually regarded as a widespread popular classification task aiming at dividing clients into "the good" and "the bad" with reference to their features such as gender, age, profession and wealth (Yang, 2007). For this classification task, many researchers have developed various methodologies, which can be roughly parted into two major classes: statistical techniques and artificial intelligence (AI) techniques (Wang, Ma, Huang, & Xu, 2012).

Among statistical techniques, Multivariate discriminant analysis (MDA) (Altman, 1968; Fisher, 1936) and logistic regression (LR) (Baesens et al., 2003; Wiginton, 1980) are two most popular methods, because they are easy to implement and interpret (Xiao, Xiao, & Wang, 2016). However, these methods will be limited in applications, because the data should be linear separable and strictly follow certain distribution. Nowadays, AI techniques, such as k-nearest neighbours (KNN) (Henley & Hand, 1996), support vector machines (SVM) (Huang, Chen, & Wang, 2007), artificial neural network (ANN) (Lee & Chen, 2005), genetic algorithm (GA) (Ong, Huang, & Tzeng, 2005), decision trees (DT) (Wang et al., 2012) and random forest (RF) (Malekipirbazari & Aksakalli, 2015), are widely used in credit domain as alternatives to the statistical techniques. In addition to these individual classifiers, ensemble models are introduced to the field of credit scoring, which perform better than individual classifiers in accuracy and stability, and these ensemble models are now considered to be the main stream.

In the real world of economics, the number of bad credit applicants turns out much lower than the good ones, that is, real datasets are imbalanced. The researchers should consider about the different costs of misclassifying "bad" objects as "good" and misclassifying "good" objects as "bad" when constructing a binary classification model, which brings a tremendous challenge (He et al., 2018). Recently, researchers have made their efforts to boost the capability of classification model in skewed data and propose several approaches, which can be further divided into two main categories: data-level methods ("over-sampling," "under-sampling," etc.), algorithm-level methods ("cost-sensitive," "ensemble," etc.) (Krawczyk, 2016).

In this paper, we propose a heterogeneous ensemble credit scoring model based on adaptive classifier selection. We try to increase the diversity of classifier ensemble in two ways. Firstly, we apply an under-sampling technique to generate several training subsets and corresponding validation subset. Secondly, distinguishing from the previous studies generally using homogeneous ensemble, we use five kinds of classifiers, including SVM, KNN, DT, MDA, LR, as the base classifiers so that the diversity can be increased. Then we train these five classifiers respectively on each training subsets and choose the one with best performance on the corresponding validation subset as the individual classifier. After all individual classifiers are trained training different individual classifiers, we predict the label of a new sample by majority voting. In the experimental study, we validate the proposed model on four imbalanced credit datasets.

The remainder of this paper is structured as follows. In Section 2, we discussed the previous work on processing imbalanced data and related researches on credit scoring. The proposed heterogeneous ensemble model is elaborated in Section 3. We describe the datasets and experimental setup in Section 4, and then give some analyses on the experimental results in Section 5. In the end, we give the conclusions and talk about the future directions in Section 6.

2 | LITERATURE REVIEW

This paper involves ensemble method on imbalanced learning, hence, we review some literatures on imbalanced learning initially in this section, then describes the evolution of ensemble learning as one of the important imbalanced learning approaches in credit scoring.

2.1 | Imbalanced learning approaches

As is mentioned, it is really a tough job to handle of binary imbalanced classification problems, because the standard learning algorithms are inclined to overvalue majority cases and overlook minority cases. In the past decade, a variety of approaches have been developed to deal with imbalanced data classification problems in credit scoring. These approaches can be mainly divided into two classes: data-level methods and algorithm-level methods (Krawczyk, 2016).

2.1.1 | Data-level methods

Data-level methods transform the distribution or structure of training datasets to adapt to the standard learning algorithms (Krawczyk, 2016). Over-sampling and under-sampling are two main data-level methods used to balance distributions of skewed data.

Over-sampling means generating new samples for minority groups. The common approach randomly selects samples from the minority groups (Batista, Prati, & Monard, 2004). However, this easily leads to overfitting when training a model. Consequently, several advanced techniques (Sun, Lang, Fujita, & Li, 2018; Zięba, Tomczak, & Gonczarek, 2015) are proposed to generate new samples. Among them, Synthetic Minority Over-Sampling Technique (SMOTE) is an advanced over-sampling method that generates new synthetic minority samples by randomly interpolating pairs of closest neighbours in the minority class (Chawla, Bowyer, Hall, & Kegelmeyer, 2002; Sun et al., 2018).

On the other hand, under-sampling removes samples from majority groups. Standard under-sampling approach randomly removes samples, which may result in the removal of important ones (Krawczyk, 2016). Aside from the basic under-sampling methods, there are more complex methods (Yen & Lee, 2009; Zhou, 2013) trying to only remove overlapping and noisy samples in order to find a representative subset of majority class samples.

2.1.2 | Algorithm-level methods

Algorithm-level methods mainly make changes on standard learning algorithms so that they can cause less bias on majority cases and be suitable for skewed data (Krawczyk, 2016).

Cost-sensitive approaches are the most popular branch. Distinguishing from the traditional approaches weight equally on different classes, cost-sensitive methods (Kim, Baik, & Cho, 2016; Sahin, Bulkan, & Duman, 2013) assign a different cost to each class. In order to deal with asymmetric misclassification costs, Kim et al. (2016) undertook cost-learning using Meta Cost and proposed three multi-class classifiers to detect accounting fraud. Similarly, a new cost-sensitive DT approach was developed by Sahin et al. (2013), which optimized the splitting attribute in order to get the minimal misclassification costs. They found that this approach outperforms other traditional credit scoring models.

One-class learning is another branch of algorithm-level methods, which focuses only on a single set of objects. Kamaruddin and Ravi (2016) developed a hybrid architecture involving Particle Swarm Optimization (PSO) and Auto-Associative Neural Network (AANN) for one-class learning in credit card fraud detection. Singh and Prasad (2013) discussed the influence of different kernel functions on one-class SVM; they show the superiority of one-class SVM for problem of credit card fraud detection.

2.2 | Evolution of ensemble methods

Although several approaches have been developed to improve the performance of single classifiers on the classification of imbalance credit data in the previous researches, single classifiers cannot solve all problems, especially when the structure and characteristics of credit data change dramatically. To remedy the limitation of single classifiers, ensemble methods (Abelian & Castellano, 2017; Baesens et al., 2003; He et al., 2018; Sun et al., 2018) have been employed to deal with the imbalance data and proved to make better performance than single classifiers.

Ensemble methods can be regarded as a process of combining multiple classifiers and their decisions to form a consensus decision by majority voting, weighted averaging, ranking or other integration strategies. There are two common methods for ensemble learning: bagging and boosting. A bagging classifier trains base classifiers parallelly, and then integrates the predictions of all individual classifiers. Many researchers (Abelian & Castellano, 2017; Ala'raj & Abbod, 2016; Baesens et al., 2003; He et al., 2018) employed bagging algorithms in the field of credit scoring. On the other hand, the same with bagging algorithms, boosting also train different classifiers through different data sets, however, boosting combines base classifiers sequentially rather than parallelly. Sun, Jia, and Li (2011) introduced AdaBoost algorithm for financial distress prediction of Chinese listed companies, they found that the choice of base learner is important to improve the performance of AdaBoost. Xia, Liu, Li, and Liu (2017) combined Bayesian hyper-parameter optimization with extreme gradient boosting (XGBoost), the proposed model outperformed baseline models. Above all, both bagging and boosting are better than single classifiers. Considering about the fact that bagging is easier to implement than boosting, we employ bagging algorithm in this study.

From another perspective, according to the chosen of base classifiers, ensemble modelling can be achieved in two ways (Lessmann, Baesens, Seow, & Thomas, 2015): homogeneous ensemble and heterogeneous ensemble. Homogenous ensemble means using the same classifiers as base classifiers, while heterogeneous ensemble denotes integrating different kinds of base classifiers. Many researchers (Abelian & Castellano, 2017; Kim, Kang, & Kim, 2015; Sun, Fujita, Chen, & Li, 2017; Sun et al., 2011; Wang et al., 2012) exploited the field of homogeneous ensembles, but rare studies involved heterogeneous ensembles. Xia, Liu, Da, and Xie (2018) developed a novel heterogeneous ensemble credit model called “bstacking” and it performs better than individual classifiers and homogeneous ensemble model.

In our study, we propose a heterogeneous bagging ensemble model, which includes five popular basic classifiers (LSVM, MDA, KNN, DT and LR) and selects base classifiers adaptively depending on the structure of dataset. The specific details are illustrated in Section 3.

3 | METHODOLOGY

In this section, we propose an adaptive classifier ensemble model applied in imbalanced credit datasets. First, we divide the original data into two parts: training data and testing data. Then we train the ensemble model on the training data. The Figure 1 show the framework of the proposed model which includes three stages: (1) In the first stage, bagging method is used to bootstrap data from the training dataset, in each bootstrap process, the bootstrapped part is used as training subset and the remaining part is used as validation subset, so that, every training subset matches with one validation subset; (2) In the second stage, base classifiers are trained respectively

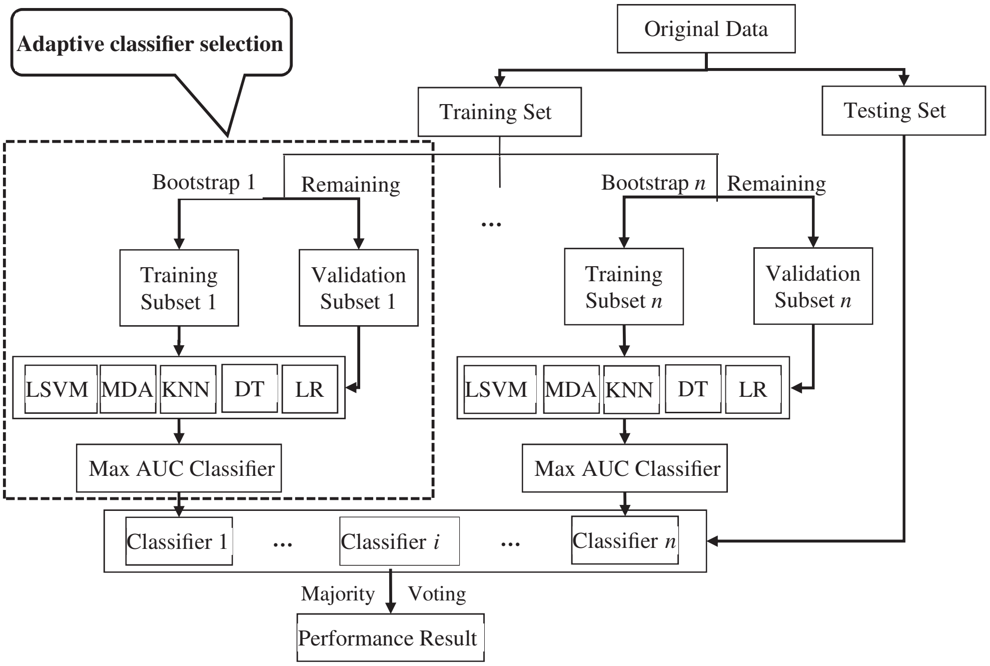

FIGURE 1 Framework of the proposed model

in each subset and validated in the corresponding validation subset, then the base classifier with highest AUC is chosen as the single classifier in each training subset; (3) In the third stage, we combine the multiple single classifiers as an ensemble classifier and integrate them by majority voting. Finally, the prediction of testing dataset is obtained through the ensemble model.

3.1 | Bagging

Bagging is a kind of ensemble model which can reduce the variance and avoid overfitting. Bagging can be implemented by Bootstrap technique. When ensemble models are constructed, diversity is very important criterion. To enhance diversity of model we create different training subsets having the same size with original training set. In this paper, we generate these training subsets by drawing samples randomly from the training data set with replacement. These training subsets are called replicated training sets, because there may be some replicated instances in these subsets. Afterwards, we build single classifiers with each subset respectively so that we can obtain several classifiers. The result is obtained through a majority voting of all single classifiers.

3.2 | Five base classifiers

In this subsection, we introduce the base classifiers which are chosen to be part of the heterogeneous ensemble model. Five base classifiers are employed in this study because of their wide application in credit scoring. They are linear SVM (LSVM), Multivariate discriminant analysis (MDA), K-nearest neighbourhood (KNN), DT, Logistic regression (LR).

3.2.1 | Linear support vector machine

SVM, known as a popular AI technique, is widely used in the field of credit scoring. The main idea of SVM is mapping input space to a more linearly separable space (Huang et al., 2007). In the new feature space, a special kind of linear model, the so-called optimal separating hyperplane, is found to categorize the data into two classes and minimizes the expected generalization error. To achieve the mapping process, a kernel function is employed to modify the data. There are mainly four kinds of kernel functions, namely, linear kernel function, polynomial kernel function, radial basis kernel function and sigmoid kernel function. In our study, only linear kernel function is used in the SVM, so the SVM is called linear SVM (LSVM).

3.2.2 | Multivariate discriminant analysis

MDA is a classical statistical model widely used to predict company distress or failure on the base of available data information. MDA is first applied in credit domain by Altman (1968); he constructed a function combining several features linearly in order to separate two kinds of clients. There are several types of MDA models; we introduce the MDA model developed by Fisher (1936) in this study. This MDA model called fisher discriminant maximizes the ratio of intra-class variance and inter-class variance. The expression of MDA model is shown as Equation (1).

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots \beta_i x_i + \cdots + \beta_n x_n, \tag{1}$$

where x_i is the value of ith feature, β_i is the coefficient of ith feature, β_0 is the intercept term, and y is the dependent variable. Classification cut-off point is the midpoint of the two group's mean values.

3.2.3 | K-Nearest neighbourhood

KNN is an algorithm that predicts the label of a new instance depending on its multiple closest neighbours' labels, usually the number of closest neighbours k is set as an odd number, then the label of new instance should be the same as the majority of k closest neighbours in training set (Henley & Hand, 1996). KNN as a non-parametric classifier has an advantage over other parameterized algorithms on robustness; nevertheless, it takes high computational cost to calculate the distance of all instances in training set. In our study, the k is set as 5.

3.2.4 | Decision tree

DT is also a non-parametric algorithm in machine learning. There are two kinds of DT, one kind is classification tree, another is regression tree, and we use the classification tree in our study. Classification tree is a tree-like model that is constructed based on training data set and used to predict the label of new instance. Classification tree can be illustrated as a tree diagram composed by nodes and oriented-edges, usually there is only one root nodes in a tree, others are internal nodes and leaf nodes. A classification tree starts with a root node, and then the training data set is split into two subsets according to the certain rules of the chosen feature. The DT algorithm searches for all possible splits of features to find the optimal one aiming at the lowest overall error rate. The advantage of DT is its good interpretability and easy implementation.

3.2.5 | Logistic regression

LR, probably the most common used classic statistical technique for credit scoring, is applied to calculate the default probability of clients. Similar with other regression models, the LR model utilizes several features which can be categorical or continuous, however, different from other linear regression models, the LR model is a nonlinear model by introducing a logit function and the dependent variable of LR model is a logit variable (Baesens et al., 2003; Wiginton, 1980). All these characters make the LR own the advantages that it can obtain the default probability of an instance and it does not require normally distributed input data. Then the class of an instance can be predicted depending on a pre-set cut-off point and the probability. For example, when the cut-off point is set as 0.5, if the probability is larger than 0.5, the instance would be classified into "bad credit," otherwise "good credit." The LR model can be expressed as Equation (2).

$$\pi(x_1, x_2, \ldots, x_n) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_i x_i + \cdots + \beta_n x_n)}}, \tag{2}$$

where x_i is value of ith feature, β_i is the coefficient of ith independent variables and $\pi(x_1, x_2, \ldots, x_n,)$ represents the default probability of a client.

3.3 | The adaptive selection of classifiers

The original bagging approach is achieved by homogeneous classifiers. In other words, each bagging subset is trained by the same kind of classifier. We usually name this kind of ensemble model homogeneous ensemble. There several kinds of homogeneous ensembles, such as, bagging-DT, bagging-SVM, bagging-LR and so on.

As is known to us, diversity is very important when constructing an ensemble model. In bagging method, diversity is increased by bootstrapping new datasets from the original dataset, thus several datasets with varieties of distributions are generated and different models are trained by different datasets. However, except for DT and ANN which can achieve an excellent improvement by using bagging method, other single classifiers do not get obvious promotion on performance when constructing an ensemble model because of their insensitiveness to data sets of various distributions. Consequently, some single classifiers like SVM and LR are not very suitable when using bagging method. That is a limitation of homogeneous ensemble methods.

To surmount the limitation, we introduce heterogeneous classifiers into the bagging method in this study. In the proposed model, we generate new datasets by bootstrapping just as bagging method does. Contrast with the traditional bagging method which use only one kind of classifiers to fit each generated dataset, we use five classical classifiers to train each dataset and choose the highest AUC one as the final classifier of the corresponding dataset. In this way, different new generated datasets may train different kind of classifiers. The varieties of both classifiers and datasets can increase the diversity of ensemble model.

Because of the employ of heterogeneous classifier, the selection of classifiers depends on the new generated dataset, that is, the selection of classifiers is adaptive. The pseudo-code of adaptive classifier selection is shown in Figure 2. The algorithm in the Figure 2 only illustrates the process of adaptive classifier selection which is also shown with the dotted box in Figure 1. During the bagging iteration, a classifier can be obtained every time after the process of adaptive classifier selection. Then several classifiers are ensembled to make decision.

3.4 | Majority voting

Majority voting is a popular ensemble strategy to combine all single classifiers together. Many single classifiers are trained during the process of adaptive classifier selection. Afterwards, all single classifiers are used to predict the labels of testing data respectively. In the majority voting, for one testing sample, when more than half of the classifiers predict it a “bad client,” then the final prediction is “bad,” else it is “good.”

4 | EXPERIMENTAL SET-UP

We design a comparative experiment to validate the performance and robustness of the proposed model. In this section, we describe the experimental set-up, which includes five parts: credit datasets, performance

```
Algorithm 1 Adaptive classifier selection

Require: Training dataset $S$={($x_i$, $y_i$), $i$=1, 2, …, $n$}, $x_i$∈R$^d$ and their corresponding class labels $y_i$=0 or 1.
    1: Randomly draw $n$ samples with replacement, called new training data set $S^N$={($x^N_i$, $y^N_i$), $i$=1, 2, …, $n$}, from original training dataset $S$, the remaining samples which are not drawn are set as validation dataset $S^V$={($x^V_i$, $y^V_i$), $i$=1, 2, …, $n^V$}. The number of samples in $S^V$ depends on the number of unique instances in $S^N$.
    2: Training five models respectively, namely LSVM, MDA, KNN, DT, LR, with $S^N$ and then predict the labels of validation dataset $S^V$ by using five models respectively:
    $\mathrm{Mdl}_{LSVM}$=train ($LSVM$, $x^N$, $y^N$), $yp^V_{LSVM}$=test ($\mathrm{Mdl}_{LSVM}$, $x^V$)
    $\mathrm{Mdl}_{MDA}$=train ($MDA$, $x^N$, $y^N$), $yp^V_{MDA}$=test ($\mathrm{Mdl}_{MDA}$, $x^V$)
    $\mathrm{Mdl}_{KNN}$=train ($KNN$, $x^N$, $y^N$), $yp^V_{KNN}$=test ($\mathrm{Mdl}_{KNN}$, $x^V$)
    $\mathrm{Mdl}_{DT}$=train ($DT$, $x^N$, $y^N$), $yp^V_{DT}$=test ($\mathrm{Mdl}_{DT}$, $x^V$)
    $\mathrm{Mdl}_{LR}$=train ($LR$, $x^N$, $y^N$), $yp^V_{LR}$=test ($\mathrm{Mdl}_{LR}$, $x^V$)
    3: Compare the predicted labels and real labels of validation dataset, calculate the AUC of each model on $S^V$.
    $\mathrm{AUC}_{LSVM}$=AUC ($yp^V_{LSVM}$, $y^V$)
    $\mathrm{AUC}_{\mathrm{MDA}}$=AUC ($yp^V_{MDA}$, $y^V$)
    $\mathrm{AUC}_{KNN}$=AUC ($yp^V_{KNN}$, $y^V$)
    $\mathrm{AUC}_{DT}$=AUC ($yp^V_{DT}$, $y^V$)
    $\mathrm{AUC}_{LR}$=AUC ($yp^V_{LR}$, $y^V$)
    4: Select the model with maximal AUC as the optimal model of the $S^N$.
    [$\mathrm{Mdl}_{\mathrm{optimal}}$, $\mathrm{AUC}_{\mathrm{max}}$]=Max($\mathrm{AUC}_{LSVM}$, $\mathrm{AUC}_{\mathrm{MDA}}$, $\mathrm{AUC}_{KNN}$, $\mathrm{AUC}_{DT}$, $\mathrm{AUC}_{LR}$)
    Output: Optimal model $\mathrm{Mdl}_{\mathrm{optimal}}$ and $\mathrm{AUC}_{\mathrm{max}}$.
```

FIGURE 2 Pseudo-code of adaptive classifier selection

TABLE 1 The dataset descriptions in the experiment

Dataset	Number of instances	Good/bad	Imbalance ratio	Total features	Selected feature
German	1,000	700/300	2.33	20	11
DefaultData	30,000	23,364/6,636	3.52	23	17
Chilean	6,569	5,227/1,342	3.89	20	20
GMSC	120,269	111,912/8,357	13.39	10	7

measures, data pre-processing, parameters setting and statistical significance testing.

4.1 | Credit dataset

In the experiment, four different credit datasets are employed to validate the proposed model. These four datasets are all imbalance datasets; it means that the proportions of default clients and healthy clients are different. The data that support the findings of this study are available from the corresponding author upon reasonable request. Within these datasets, Default credit card client (DefaultData) dataset (http://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients) and German credit dataset (http://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29) are draw from the UCI Machine Learning Repository (He et al., 2018), Chilean credit dataset is provided by the R package "Smbinning" (https://cran.r-project.org/web/packages/smbinning/index.html), and Give Me Some Credit (GMSC) dataset (https://www.kaggle.com/c/GiveMeSomeCredit/data) is obtained from the platform "Kaggle." We describe the basic information of these four datasets in Table 1.

In this study, "good credit" samples are defined as negative samples while "bad credit" samples as positive ones. German data contains 1,000 samples and 14 initial features, in which 700 are "good credit" and others are "bad credit," it means that the imbalance ratio of German data is 2.33. Similar with German data, DefaultData has totally 30,000 samples with 23 initial features and its imbalance ratio is 3.52. Chilean dataset has 6,569 samples with 5,227 negative samples and 1,342 positive samples; the initial feature of Chilean is 20. GMSC dataset has the highest imbalance ratio of 13.39; there are totally 111,912 negative samples and 8,357 positive samples.

4.2 | Performance measures

It is crucial to select appropriate performance measures, which can reflect the model's overall performance, when confirming the validation of the proposed model and comparing it with the benchmarks. In credit scoring, clients are usually divided into "good" and "bad." According to the real label and predicted label, the predicted clients have four statuses, which is illustrated by the confusion matrix. These four statuses are basic elements constituting various performance measures. Shown in Table 2, True positive (TP) denote bad clients predicted as bad ones; false negative (FN) indicate that the bad credit samples are predicted as good credit samples; false positives (FP) are samples with good label but predicted as the opposite label; true negative (TN) denote good clients with a bad predicted label. Accuracy is one of the popular performance measures used by many researchers in the credit scoring, but it tends to generate deceptive high accuracy and turns invalid when the sample is imbalanced (He et al., 2018). To deal with this deficiency, we employ five performance measures in this study, namely, G-mean (GM), F-measure (FM), Mathew correlation coefficient (MCC), Bookmaker informedness (BM), area under the receiver operating characteristic curve (AUC), these measures are mostly constructed based on the four basic elements illustrated in confusion matrix.

G-Mean (*GM*): A comprehensive performance measure based on TP rate (TPR) and true negative rate (TNR). The definition of GM is illustrated by Equation (3); TPR and TNR are shown in Equations (4) and (5). A GM value means that the model has a good performance on credit scoring.

$$GM = \sqrt{TPR \times TNR}, \tag{3}$$

$$TPR = \frac{TP}{TP + FN}, \tag{4}$$

TABLE 2 Confusion matrix

		Predicted	
		Positive	Negative
Real	Positive	True positive (TP)	False negative (FN)
	Negative	False positive (FP)	True negative (TN)

Algorithm 1

Adaptive classifier selection

```
Require: Training dataset $S = \{(x_i, y_i), i = 1, 2, ..., n\}$, $x_i \in R^d$ and their corresponding class labels $y_i = 0$ or 1.
  1: Randomly draw $n$ samples with replacement, called new training data set $S^N = \{(x^N{}_i, y^N{}_i), i = 1, 2, ..., n\}$, from original training dataset $S$, the remaining samples which are not drawn are set as validation dataset $S^V = \{(x^V{}_i, y^V{}_i), i = 1, 2, ..., n^V\}$. The number of samples in $S^V$ depends on the number of unique instances in $S^N$.
  2: Training five models respectively, namely LSVM, $MDA$, $KNN$, $DT$, $LR$, with $S^N$ and then predict the labels of validation dataset $S^V$ by using five models respectively:
     $\mathrm{Mdl}_{LSVM}$ = train ($LSVM$, $x^N$, $y^N$), $yp^V{}_{LSVM}$ = test ($\mathrm{Mdl}_{LSVM}$, $x^V$).
     $\mathrm{Mdl}_{MDA}$ = train ($MDA$, $x^N$, $y^N$), $yp^V{}_{MDA}$ = test ($\mathrm{Mdl}_{MDA}$, $x^V$).
     $\mathrm{Mdl}_{KNN}$ = train ($KNN$, $x^N$, $y^N$), $yp^V{}_{KNN}$ = test ($\mathrm{Mdl}_{KNN}$, $x^V$).
     $\mathrm{Mdl}_{DT}$ = train ($DT$, $x^N$, $y^N$), $yp^V{}_{DT}$ = test ($\mathrm{Mdl}_{DT}$, $x^V$).
     $\mathrm{Mdl}_{LR}$ = train ($LR$, $x^N$, $y^N$), $yp^V{}_{LR}$ = test ($\mathrm{Mdl}_{LR}$, $x^V$).
  3: Compare the predicted labels and real labels of validation dataset, calculate the AUC of each model on $S^V$
     $\mathrm{AUC}_{LSVM}$ = AUC ($yp^V{}_{LSVM}$, $y^V$).
     $\mathrm{AUC}_{\mathrm{MDA}}$ = AUC ($yp^V{}_{MDA}$, $y^V$).
     $\mathrm{AUC}_{KNN}$ = AUC ($yp^V{}_{KNN}$, $y^V$).
     $\mathrm{AUC}_{DT}$ = AUC ($yp^V{}_{DT}$, $y^V$).
     $\mathrm{AUC}_{LR}$ = AUC ($yp^V{}_{LR}$, $y^V$).
  4: Select the model with maximal AUC as the optimal model of the $S^N$
     [$\mathrm{Mdl}_{\mathrm{optimal}}$, $\mathrm{AUC}_{\max}$] = Max($\mathrm{AUC}_{LSVM}$, $\mathrm{AUC}_{\mathrm{MDA}}$, $\mathrm{AUC}_{KNN}$, $\mathrm{AUC}_{DT}$, $\mathrm{AUC}_{LR}$).
Output: Optimal model $\mathrm{Mdl}_{\mathrm{optimal}}$ and $\mathrm{AUC}_{\max}$.
```

$$TNR = \frac{TN}{TN + FP}. \tag{5}$$

F-measure (*FM*): It is also a comprehensive performance measure, which is the harmonic mean of precision and recall. FM is defined as Equation (6), while the definitions of precision and recall are shown respectively as Equations (7) and (8).

$$FM = \frac{2 \times recall \times precision}{recall + precision}, \tag{6}$$

$$precision = \frac{TP}{TP + FP}, \tag{7}$$

$$recall = \frac{TP}{TP + FN}. \tag{8}$$

Mathew correlation coefficient (*MCC*): A comprehensive performance measure constructed by four basic elements in confusion matrix, Equation (9) shows the definition of *MCC*.

$$MCC = \frac{(TP \times TN - FP \times FN)}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}}. \tag{9}$$

Bookmaker Informedness (BM): A comprehensive performance measure constructed by TPR and TNR, Equation (10) shows the definition of *BM*.

$$BM = TPR + TNR - 1. \tag{10}$$

AUC: It represents the area under the Receiver Operating Characteristic (ROC) curve, of which the x-axis is the false positive rate (FPR) and the y-axis is TPR. AUC can be comprehensive performance measure in comparative experiment.

4.3 | Data processing

It is of great importance for us to pre-process the data before training the models. There are two kinds of features: categorical features and continuous features. For the categorical features, the missing values are replaced by the most frequently appearing category. For the continuous features, we use the mean value of samples to fill the missing values.

Because of the sensitivities of some classifiers to feature scaling, we standardize the features by Equation (11) after filling the missing value.

$$x' = \frac{x - \mu}{\sigma}. \tag{11}$$

x' denotes the feature value after standardizing, x denotes a given feature, μ denotes the mean value of feature x, and σ denotes the variance of feature x.

After the pre-processing step above, we select the features by the method of *t*-test in a 1% significant level. As

TABLE 3 Results of different models

		Individual classifier					Homogeneous ensemble					Heterogeneous ensemble	
Dataset	Performance measure	LSVM	MDA	KNN	DT	LR	Bag-LSVM	Bag-MDA	Bag-KNN	Bag-DT	Bag-LR	Traditional	Proposed
German	GM	0.620	0.629	0.601	0.614	0.627	0.627	0.642	0.597	0.658	0.625	0.642	0.658
	FM	0.512	0.520	0.470	0.485	0.520	0.517	0.538	0.465	0.549	0.514	0.538	0.552
	MCC	0.365	0.365	0.251	0.267	0.371	0.363	0.391	0.249	0.381	0.358	0.390	0.396
	BM	0.322	0.330	0.247	0.266	0.331	0.327	0.352	0.244	0.363	0.324	0.352	0.369
	AUC	0.661	0.665	0.623	0.633	0.666	0.664	0.676	0.622	0.681	0.662	0.676	0.684
	AvgR	8.4	6.0	11.0	10.0	5.2	7.2	3.2	12.0	2.2	8.2	3.4	1.2
DefaultData	GM	0.482	0.498	0.572	0.576	0.477	0.483	0.494	0.573	0.599	0.477	0.496	0.605
	FM	0.357	0.376	0.395	0.403	0.352	0.357	0.371	0.396	0.474	0.352	0.373	0.478
	MCC	0.333	0.345	0.222	0.238	0.330	0.331	0.341	0.224	0.382	0.331	0.343	0.381
	BM	0.211	0.226	0.222	0.234	0.207	0.211	0.222	0.224	0.317	0.207	0.223	0.322
	AUC	0.605	0.613	0.611	0.617	0.603	0.605	0.611	0.612	0.658	0.604	0.612	0.661
	AvgR	9.0	4.6	7.2	4.4	11.4	8.8	7.4	5.8	1.8	10.4	6.0	1.2
Chilean	GM	0.638	0.651	0.625	0.616	0.627	0.643	0.654	0.630	0.667	0.629	0.635	0.670
	FM	0.520	0.531	0.452	0.448	0.506	0.526	0.535	0.457	0.536	0.510	0.514	0.544
	MCC	0.439	0.445	0.309	0.310	0.424	0.446	0.450	0.316	0.436	0.430	0.431	0.450
	BM	0.373	0.388	0.312	0.305	0.358	0.381	0.393	0.320	0.402	0.362	0.368	0.410
	AUC	0.687	0.694	0.656	0.652	0.679	0.690	0.696	0.660	0.701	0.681	0.684	0.705
	AvgR	5.8	4.0	11.2	11.8	9.2	4.6	2.6	9.6	2.8	8.2	7.0	1.2
GMSC	GM	0.098	0.304	0.461	0.463	0.196	0.100	0.306	0.459	0.432	0.193	0.223	0.437
	FM	0.019	0.158	0.229	0.255	0.072	0.020	0.159	0.231	0.271	0.070	0.091	0.277
	MCC	0.066	0.199	0.173	0.210	0.136	0.068	0.198	0.176	0.270	0.132	0.153	0.276
	BM	0.009	0.086	0.170	0.184	0.036	0.009	0.088	0.170	0.174	0.035	0.047	0.179
	AUC	0.505	0.543	0.585	0.592	0.518	0.505	0.544	0.585	0.587	0.518	0.524	0.589
	AvgR	12.0	6.4	4.6	1.8	9.0	11.0	5.8	4.4	3.0	10.0	8.0	2.0

Note: The best performance for each evaluation measure is highlighted in bold.

is shown in the last column of Table 1, finally, the German dataset retains 11 features, the DefaultData dataset has 17 features, the Chilean dataset has 20 features, and the GMSC dataset contains seven features.

4.4 | Parameters setting

To carry out the experiment, we set 70% of total data as training set for each dataset, while the remaining 30% is used as the testing set. We randomly divide the dataset for 20 times so that we can get 20 pairs of training and testing data. For each pair of datasets, we train and test the model on the training data and testing data respectively. Finally, we get the average value of performance measures.

Five popular models are chosen as base classifiers, they are LSVM, MDA, KNN, DT and LR. According to Sun et al. (2018), we set the number of bags to 30.

4.5 | Statistical significance test

We introduce a non-parametric test to validate the significant difference of model performance. The non-parametric test is called Friedman test which is based on the ranks of different performance measures. The Equation (12) shows the calculation of the Friedman test statistics (Zhang, He, & Zhang, 2019):

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_{j=1}^{k} AvR_j{}^2 - \frac{k(k+1)^2}{4}\right], \tag{12}$$

$$AvR_j = \frac{1}{N}\sum_{i=1}^{N} r_i^j, \tag{13}$$

Within Equation (12), N denotes the number of datasets used in the comparative experiment, k is the number of classifiers, AvR_j is the average performance ranks of jth classifier, and it can be obtained by Equation (13). In Equation (13), r_i^j denotes the average rank of classifier j on dataset i. Particularly, we can calculate the average rank for each classifier.

5 | EXPERIMENTAL ANALYSIS

We totally set 11 benchmark models for purpose of comprehensively confirming the effectiveness of the proposed model. These benchmark models can be divided into three classes including five kinds of individual learners, five homogeneous ensemble models and a heterogeneous ensemble model with traditional strategy. Furthermore, four imbalanced credit databases and five comprehensive performance measures are employed to validate the models. We run the experiment in MATLAB r2018a, basic configuration of the desktop is as follows: Intel Core i7-3,470 3.2GHz CPU, 8 GB RAM and Microsoft Windows 10 operating system.

5.1 | Classification results

In this subsection, experimental results are presented to validate the performance of the proposed ensemble model.

TABLE 4 The average rank of classifiers in different datasets

Classifier family	Classifiers	German	DefaultData	Chilean	GMSC	Avg rank
Individual classifier	LSVM	8.4	9.0	5.8	12.0	8.8
	MDA	6.0	4.6	4.0	6.4	5.3
	KNN	11.0	7.2	11.2	4.6	8.5
	DT	10.0	4.4	11.8	**1.8**	7.0
	LR	5.2	11.4	9.2	9.0	8.7
Homogeneous ensemble	LSVM	7.2	8.8	4.6	11.0	7.9
	MDA	3.2	7.4	2.6	5.8	4.8
	KNN	12.0	5.8	9.6	4.4	8.0
	DT	2.2	1.8	2.8	3.0	2.5
	LR	8.2	10.4	8.2	10.0	9.2
Heterogeneous ensemble	Traditional	3.4	6.0	7.0	8.0	6.1
	Proposed	**1.2**	**1.2**	**1.2**	2.0	**1.4**

TABLE 5 The average ranks across datasets for different performance measures

Classifier family	Classifiers	GM		FM		MCC		BM		AUC		Avg rank		High score
Individual classifier	LSVM	9.3	(.000)	9.3	(.000)	7.3	(.001)	9.3	(.000)	9.0	(.000)	8.8	(0.000)	11
	MDA	5.5	(.008)	5.8	(.003)	4.5	(.011)	5.3	(.005)	5.3	(.005)	5.3	(0.006)	4
	KNN	7.3	(.002)	8.0	(.000)	10.5	(.000)	8.5	(.000)	8.3	(.000)	8.5	(0.000)	9
	DT	6.5	(.004)	7.0	(.001)	8.5	(.000)	6.5	(.002)	6.5	(.002)	7.0	(0.001)	6
	LR	9.3	(.000)	8.8	(.000)	8.0	(.000)	8.8	(.000)	8.8	(.000)	8.7	(0.000)	10
Homogeneous ensemble	Bag-LSVM	8.0	(.001)	8.0	(.000)	7.3	(.001)	8.0	(.000)	8.3	(.000)	7.9	(0.000)	7
	Bag-MDA	5.3	(.009)	5.3	(.004)	3.3	(.022)	5.0	(.006)	5.0	(.006)	4.8	(0.009)	3
	Bag-KNN	6.8	(.003)	7.5	(.000)	9.8	(.000)	7.8	(.000)	8.0	(.000)	8.0	(0.000)	8
	Bag-DT	2.5	(.038)	2.0	(.032)	3.3	(.022)	2.3	(.032)	2.3	(.032)	2.5	(0.031)	2
	Bag-LR	9.5	(.000)	9.3	(.000)	8.8	(.000)	9.3	(.000)	9.3	(.000)	9.2	(0.000)	12
Heterogeneous ensemble	Traditional	6.3	(.004)	6.3	(.002)	5.5	(.005)	6.3	(.002)	6.3	(.002)	6.1	(0.003)	5
	Proposed	**2.0**	(/)	**1.0**	(/)	**1.5**	(/)	**1.3**	(/)	**1.3**	(/)	**1.4**	(/)	**1**
Friedman χ^2_{12}	23.07	(.004)	21.63	(.009)	25.57	(.001)	28.95	(.000)	25.39	(.001)	25.14	(.001)		

Note: Bold face indicates the best classifier (lower average rank) per performance measure.

We respectively discuss different classifier structures including individual classifier, homogeneous ensemble and heterogeneous ensemble. We assume that the proposed ensemble further enhances the diversity in an ensemble system compared with homogeneous ensemble and traditional heterogeneous ensemble, which means the proposed heterogeneous ensemble outperforms the other models in credit classification. To confirm this assumption, we design a comprehensive comparative experiment and utilize several baseline models including individual classifiers, homogeneous ensemble classifiers and traditional heterogeneous ensemble classifier. We employ LSVM, MDA, KNN, DT and LR to build the individual classifiers. Then these base models are also used to build homogeneous ensemble classifiers, including bagging+LSVM (bag-LSVM), bagging+LDA (bag-MDA), bagging+KNN (bag-KNN), bagging+DT (bag-DT) and bagging+LR (bag-LR). Meanwhile, we employ two heterogeneous ensemble models, a traditional one and the proposed one, which combine LSVM, MDA, KNN, DT and LR with majority voting.

Table 3 exhibits the performance results of all models, including five performance measures and the average ranks of classifiers across performance measures. Table 4 reports the average ranks of different classifiers across different data sets. We can obtain some key findings and draw some conclusion from the experimental results.

As for the individual classifiers, MDA turned out the best one among all; it obtained an average rank of 5.3. DT was the second-best classifier over most of performance measures; it ranked seven among all the classifiers. KNN, LR, LSVM were the worst-performing individual classifier.

With respect to the homogeneous ensemble, four out of five base classifiers except for LR showed some obvious improvements in most datasets compared with individual classifiers. Among all, DT benefited the most from ensemble learning. The average rank of DT rose from 7 to 2.5. The average rank of LR dropped from 8.7 to 9.2.

With regard to the heterogeneous ensemble, the traditional heterogeneous ensemble was better than most of the individual classifier except for MDA, while the proposed heterogeneous ensemble model based on adaptive classifier selection was better than all the other models, including all homogeneous ensemble. The proposed heterogeneous ensemble model performed the best rank of 1.4 of all the classifiers.

5.2 | Comprehensive analysis and significance tests results

In this subsection, we compare 12 different classifiers across four different credit datasets. To demonstrate the performance difference of these classifiers, we employ the Friedman test. Then a post-hoc Bonferroni–Dunn procedure (Garcia, Fernandez, Luengo, & Herrera, 2010) is applied to testify the superiority of the proposed model.

Table 5 displays the average ranks of five performance measures across datasets. The value of best rank is marked bold for each performance measure. For example, the proposed model gets an average rank of 2.0 across four datasets for GM. We can easily know that the proposed model can get best average ranks for all performance measures. According to Zhang et al. (2019), an overall average (Avg Rank) shown in penultimate column of Table 5 is calculated to represent the comprehensive abilities of classifiers. These Avg Ranks are also converted to high scores in the last column. The proposed model gets an Avg Rank of 1.4 and the corresponding high score is 1, which means that the proposed model is the best model from an overall perspective. Furthermore, we also conduct a pairwise comparison experiment between the baseline model and the proposed model. The *p*-values adjusted by Bonferroni–Dunn procedure (Garcia et al., 2010) are shown as the values in brackets of Table 5, which illustrate the significant ranking different between the proposed model and baseline model. Finally, the statistical values and *p*-values of Friedman test are listed in the last row of Table 5.

There are some conclusions drawn from Table 5. First, in the pairwise comparison, all *p*-values are less than .05, and the *p*-values of Friedman test are also less than .05 (the null-hypothesis that all models are equal is rejected); these two results prove that the proposed model is superior to other models. Second, the homogeneous ensemble and heterogeneous ensemble perform better than individual classifiers. Third, the ensemble models show different performance depending on the chosen base classifiers. In this experiment, bag-DT and bag-MDA rank higher than other three classifiers. What is more, the proposed heterogeneous ensemble based on adaptive classifier selection shows obvious superiority over the traditional heterogeneous ensemble with constant classifiers.

6 | CONCLUSION AND FUTURE WORK

Credit scoring is always a hot research topic in the field of financial risk management. The imbalance classification problem as a critical issue has attracted much attentions from scholars. In this study, for dealing with the imbalance classification problem in credit scoring, we propose a novel heterogeneous ensemble which can select the base classifiers adaptively depending on different data sets. The study extends the traditional heterogeneous ensemble model to an adaptive classifier selection heterogeneous ensemble by introducing bagging method. In each training subset, varieties of base classifiers are trained and the best base classifier is chosen according to AUC of the corresponding validation subset. Finally, all base classifiers combine their results to a final one by majority voting. In the experiment, we apply the proposed model and benchmark models in four credit datasets with different imbalance ratios and use five comprehensive performance measures to compare the models, so that we can confirm that the proposed model is valid and superior in credit scoring. The experimental results show that the proposed model significantly outperforms the benchmark individual, homogeneous ensemble models and traditional heterogeneous ensemble models.

In addition, there are some limitations in this study. For example, in this study, we use a simple *t*-test method to screen features; we can improve the feature selection procession by applying other more complex filter, wrapper or embedded methods in the future. Further, in this study, we only try five kinds of base classifiers, more kinds of base classifiers such as deep learning algorithms should be employed in further exploration which may lead to better performance. Another direction of our future is adopting other integration strategies. Finally, we can take the time cost of the model into consideration in the future work.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Tong Zhang https://orcid.org/0000-0003-3116-1272

REFERENCES

Abelian, J., & Castellano, J. G. (2017). A comparative study on base classifiers in ensemble methods for credit scoring. *Expert Systems with Applications*, *73*, 1–10.

Ala'raj, M., & Abbod, M. F. (2016). Classifiers consensus system approach for credit scoring. *Knowledge-Based Systems*, *104*, 89–105.

Almunia, M., Benetrix, A., Eichengreen, B., O'Rourke, K. H., & Rua, G. (2010). From great depression to great credit crisis: Similarities, differences and lessons. *Economic Policy*, *25*(62), 219–265.

Altman, E. I. (1968). Financial ratios, discriminant analysis and prediction of corporate bankruptcy. *Journal of Finance*, *23*(4), 589–609. https://doi.org/10.2307/2978933

Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society*, *54*(6), 627–635. https://doi.org/10.1057/palgrave.jors.2601545

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *Acm Sigkdd Explorations Newsletter*, *6*(1), 20–29.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, *7*, 179–188.

Garcia, S., Fernandez, A., Luengo, J., & Herrera, F. (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences*, *180*(10), 2044–2064. https://doi.org/10.1016/j.ins.2009.12.010

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

Henley, W. E., & Hand, D. J. (1996). A k-nearest-neighbour classifier for assessing consumer credit risk. *Journal of the Royal Statistical Society Series D-the Statistician*, *45*(1), 77–95. https://doi.org/10.2307/2348414

Huang, C. L., Chen, M. C., & Wang, C. J. (2007). Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications*, *33*(4), 847–856. https://doi.org/10.1016/j.eswa.2006.07.007

Kamaruddin, S., & Ravi, V. (2016). Credit Card Fraud Detection using Big Data Analytics: Use of PSOAANN based One-Class Classification. Paper presented at the ICIA-16: Proceedings of the International Conference on Informatics and Analytics. 33, 1–8. https://doi.org/10.1145/2980258.2980319.

Kim, M.-J., Kang, D.-K., & Kim, H. B. (2015). Geometric mean based boosting algorithm with over-sampling to resolve data imbalance problem for bankruptcy prediction. *Expert Systems with Applications*, *42*(3), 1074–1082. https://doi.org/10.1016/j.eswa.2014.08.025

Kim, Y. J., Baik, B., & Cho, S. (2016). Detecting financial misstatements with fraud intention using multi-class cost-sensitive learning. *Expert Systems with Applications*, *62*, 32–43.

Krawczyk, B. (2016). Learning from imbalanced data: Open challenges and future directions. *Progress in Artificial Intelligence*, *5*(4), 221–232. https://doi.org/10.1007/s13748-016-0094-0

Lee, T. S., & Chen, I. F. (2005). A two-stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines. *Expert Systems with Applications*, *28*(4), 743–752. https://doi.org/10.1016/j.eswa.2004.12.031

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

Longstaff, F. A. (2010). The subprime credit crisis and contagion in financial markets. *Journal of Financial Economics*, *97*(3), 436–450.

Malekipirbazari, M., & Aksakalli, V. (2015). Risk assessment in social lending via random forests. *Expert Systems with Applications*, *42*(10), 4621–4631. https://doi.org/10.1016/j.eswa.2015.02.001

Ong, C. S., Huang, J. J., & Tzeng, G. H. (2005). Building credit scoring models using genetic programming. *Expert Systems with Applications*, *29*(1), 41–47. https://doi.org/10.1016/j.eswa.2005.01.003

Sadatrasoul, S. M., Gholamian, M., Siami, M., & Hajimohammadi, Z. (2013). Credit scoring in banks and financial institutions via data mining techniques: A literature review. *Journal of AI and Data Mining*, *1*(2), 119–129. https://doi.org/10.22044/jadm.2013.124

Sahin, Y., Bulkan, S., & Duman, E. (2013). A cost-sensitive decision tree approach for fraud detection. *Expert Systems with Applications*, *40*(15), 5916–5923.

Singh, M. H., & Prasad, Y. (2013). One-class support vector machines approach to anomaly detection. *Applied Artificial Intelligence*, *27*(5), 351–366.

Sun, J., Fujita, H., Chen, P., & Li, H. (2017). Dynamic financial distress prediction with concept drift based on time weighting combined with Adaboost support vector machine ensemble. *Knowledge-Based Systems*, *120*, 4–14. https://doi.org/10.1016/j.knosys.2016.12.019

Sun, J., Jia, M. Y., & Li, H. (2011). AdaBoost ensemble for financial distress prediction: An empirical comparison with data from Chinese listed companies. *Expert Systems with Applications*, *38*(8), 9305–9312. https://doi.org/10.1016/j.eswa.2011.01.042

Sun, J., Lang, J., Fujita, H., & Li, H. (2018). Imbalanced enterprise credit evaluation with DTE-SBD: Decision tree ensemble based on SMOTE and bagging with differentiated sampling rates. *Information Sciences*, *425*, 76–91. https://doi.org/10.1016/j.ins.2017.10.017

Wang, G., Ma, J., Huang, L. H., & Xu, K. Q. (2012). Two credit scoring models based on dual strategy ensemble trees. *Knowledge-Based Systems*, *26*, 61–68. https://doi.org/10.1016/j.knosys.2011.06.020

Wiginton, J. C. (1980). A note on the comparison of logit and discriminant models of consumer-credit behavior. *Journal of Financial and Quantitative Analysis*, *15*(3), 757–770. https://doi.org/10.2307/2330408

Xia, Y., Liu, C., Li, Y., & Liu, N. (2017). A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications*, *78*, 225–241. https://doi.org/10.1016/j.eswa.2017.02.017

Xia, Y. F., Liu, C. Z., Da, B. W., & Xie, F. M. (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

Xiao, H. S., Xiao, Z., & Wang, Y. (2016). Ensemble classification based on supervised clustering for credit scoring. *Applied Soft Computing*, *43*, 73–86. https://doi.org/10.1016/j.asoc.2016.02.022

Yang, Y. X. (2007). Adaptive credit scoring with kernel learning methods. *European Journal of Operational Research*, *183*(3), 1521–1536. https://doi.org/10.1016/j.ejor.2006.10.066

Yen, S. J., & Lee, Y. S. (2009). Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, *36*(3), 5718–5727.

Zhang, W. Y., He, H. L., & Zhang, S. (2019). A novel multi-stage hybrid model with enhanced multi-population niche genetic algorithm: An application in credit scoring. *Expert Systems with Applications*, *121*, 221–232. https://doi.org/10.1016/j.eswa.2018.12.020

Zhou, L. (2013). Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods. *Knowledge-Based Systems*, *41*, 16–25. https://doi.org/10.1016/j.knosys.2012.12.007

Zięba, M., Tomczak, J. M., & Gonczarek, A. (2015). RBM-SMOTE: Restricted Boltzmann Machines for Synthetic Minority Oversampling Technique. Paper presented at the Asian Conference on Intelligent Information and Database Systems ACIIDS 2015: Intelligent Information and Database Systems. pp 377–386. https://doi.org/10.1007/978-3-319-15702-3_37

How to cite this article: Zhang T, Chi G. A heterogeneous ensemble credit scoring model based on adaptive classifier selection: An application on imbalanced data. *Int J Fin Econ*. 2020;1–14. https://doi.org/10.1002/ijfe.2019