IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 4, 2019, accepted June 8, 2019, date of publication June 13, 2019, date of current version June 28, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2922676

A Multi-Stage Self-Adaptive Classifier Ensemble Model With Application in Credit Scoring

SHANSHAN GUO[1], HONGLIANG HE[2], AND XIAOLING HUANG[3]
[1]Library, Zhejiang University of Finance and Economics, Hangzhou 310018, China
[2]School of Information, Zhejiang University of Finance and Economics, Hangzhou 310018, China
[3]School of International Education, Zhejiang University of Finance and Economics, Hangzhou 310018, China

Corresponding author: Xiaoling Huang (huangxl@zufe.edu.cn)

This work was supported by the Humanities and Social Sciences Research Project of the Education Ministry of China under Grant 15YJC870008.

ABSTRACT In recent years, credit scoring has received wide attention from financial institutions with the rating accuracy influencing both risk control and profitability to a considerable extent. This paper presents a novel multi-stage self-adaptive classifier ensemble model based on the statistical techniques and the machine learning techniques to improve the prediction performance. First, the multi-step data preprocessing is employed to process the original data into the standardized data and generate more representative features. Second, base classifiers can be self-adaptively selected from the candidate classifier repository according to their performance in datasets and their parameters are optimized by the Bayesian optimization algorithm. Third, the ensemble model is integrated through these optimized base classifiers, and it can generate new features through multi-layer stacking and obtain the classifier weights in the ensemble model through the particle swarm optimization. The proposed model is applied to credit scoring to test its prediction performance. In the experimental study, three real-world credit datasets and four evaluation indicators are adopted for the performance evaluation. The results show that compared to single classifier and other ensemble classification methods, the proposed model has better performance and better data adaptability. It proves the reliability and practicability of the proposed model and provides effective decision support for the relevant financial institutions.

INDEX TERMS Credit scoring, multi-stage, self-adaptive, classifier ensemble.

I. INTRODUCTION

In the past decades, statistical techniques and machine learning techniques are widely used in various fields. Numerous classifiers have been applied in binary classification, typically including Logistic Regression (LR), Linear Discriminant Analysis (LDA), K nearest neighbor (KNN), Naive Bayes (NB), support vector machine (SVM), decision tree (DT), and multilayer perceptron neural network (MLP). Among the classifiers mentioned above, LR is the most widely used statistical method. DT is often used as a base classifier, while SVM and MLP are also extensively found in the research. Previous research has found that each of these classifiers has strengths and weaknesses. For example, LR is an accessible and flexible statistical method with the advantages of fast execution speed and low computational cost, but with the disadvantage of being easy to underfit. DT is a simple and easily-interpretable algorithm with a strong expansibility which is insensitive to intermediate values and suitable for dealing with missing attributes. SVM is an algorithm with a solid theoretical basis and a low error rate, which can be used to deal with high dimensional data. However, its sensitivity to parameter adjustment and function selection is a noticeable weakness that must be considered.

To address the shortcomings of a single model, the study of machine learning technology is gradually moving to the ensemble model, which can both adopt the advantages of the base models and reduce their disadvantages. The two most common forms of ensemble model are the hybrid method and the ensemble method (i.e. hybrid classifier and classifier ensembles). The hybrid method refers to the combination of feature selection or parameter optimization before the classification. The ensemble method refers to the combination of multiple classifiers run in parallel [1]. Many existing ensemble models are variations or refinements on both methods [2], [3]. Nevertheless, despite the fact that considerable research

The associate editor coordinating the review of this manuscript and approving it for publication was Zhan Bu.

has been devoted to application of the hybrid classifier and classifier ensembles, little attention has been paid to the selection of base classifiers for different datasets and the ensemble of the hybrid classifiers, not to mention the use of a multi-stage ensemble strategy to improve the prediction performance of ensemble models.

This study proposed a novel multi-stage self-adaptive ensemble model to combine multiple effective components and select appropriate base classifiers for different credit scoring data, so as to improve the prediction performance. With the employment of several serial and parallel techniques, the proposed model focuses on three stages: 1) multi-step data preprocessing is applied; 2) base classifiers are self-adaptively selected for different datasets; and 3) the ensemble model is constructed to generate new features through multi-layer stacking. Particle swarm optimization (PSO) algorithm [4] is used to optimize the weight of weighted majority voting and obtain the final prediction result. In general, the proposed model can adaptively select the appropriate base classifiers for different datasets and use them to construct a self-adaptive classifier ensemble model, so that the prediction performance of the model is improved.

Credit scoring has always been critical for banks and financial institutions to assess whether to approve loan application from a client. Even 1% of improvement on the prediction performance of recognizing applicants would greatly increase the profit of banks and financial institutions [5]. In recent years, with the continuous enrichment of financial products and the rapid development of internet finance, a growing number of scholars are paying attention to the prediction performance of credit scoring model. Based on previous research [6], [7], it can be concluded that these three stages (i.e. data preprocessing, classifier selection, and classifier ensemble) can effectively improve the robustness and the prediction performance.

The remainder of this paper is organized as follows: In Section 2, the development of classification techniques in credit scoring is reviewed. In Section 3, a novel multi-stage self-adaptive classifier ensemble model is proposed, and the main stages are described. In Section 4, the experimental setup of the study is introduced, and in Section 5, the experimental results and analysis are presented. Finally, a conclusion and future works are given in Section 6.

II. RELATED WORK

To improve prediction performance, a number of classifiers are developed and improved. In this section, the development of classification techniques and their applications in credit scoring is reviewed.

A. BASE (SINGLE) CLASSIFIERS

The application of the base classifier is common, as discussed by Hoffmann *et al.* [8], [9], Ong *et al.* [10], and Wang and Huang [11]. Three common classifiers, which are the base classifiers in this study, are described in this sub-section.

LR is a probabilistic nonlinear model, which refers to a multivariate analysis method to study the relationship between the result y and the influencing factors $(x_1, x_2, x_3, \ldots, x_n)$. It has been widely applied to credit scoring in various disciplines, and has become one of the most important statistical techniques used for this purpose. Hosmer and Lemeshow pointed out that LR can be regarded as a special case of linear regression [12]. The main difference between LR and linear regression, is that LR is usually used for classification problems, while linear regression is generally used to fit the data when its values are normally distributed. Tang and Chi applied the LR to trade credit risk forecasting, and showed that LR has better classification accuracy and robustness when studying multiple sets of data [13]. They also concluded that the closer the analysis is to the occurrence of credit crisis, the higher the classification accuracy and greater the improvement in prediction accuracy. The advantages of LR are fewer required parameters, and that the model is easy to understand and implement. In many cases, LR is used as a base classifier to construct more complex models. For example, Sohn et al. proposed fuzzy logistic regression to predict the credit scoring data with fuzzy number attributes [14]. He et al. and Zhang et al. used logistic regression as stacked model in stacking ensemble model, so as to construct a complex ensemble model for predicting credit scoring data [6], [7].

Decision tree is a prediction model that indicates the reflexive relationship between the properties of the objects and their attributes. Every node represents an entity, with the divergent path defined as the probable attributes of the properties, and the leaf node representing the attributes of objects which track the path from leaf node to root node. There are three standard tree algorithms, namely: (1) the chi-square automatic interaction detector, which uses the chi-square test as the splitting criterion; (2) the classification and regression tree (CART), which uses gini as the splitting criterion; and (3) C5, which uses entropy as the splitting criterion. Boyle et al. were some of the first scholars to advocate the application of a decision tree method to credit scoring [15]. Lee et al. studied CART in the application of the score, and compared it with other methods [16]; from this they found that the performance of CART in experimental datasets was superior to other traditional methods. The explanatory and expansibility capabilities of DT are excellent, and more complex models can be built on the basis of DT. Siami et al. experimented the locally linear model tree algorithm to evaluate the superiority of DT's performance and achieved a good result [17]. However, DT also has a drawback in that it is easy to overfit and often ignores the correlation between data features. Therefore, in recent research, the basic DT model is generally not used alone, but often used as a base classifier for ensemble models.

SVM was first proposed by Cortes and Vapnik [18]. This model tries to find a hyperplane to separate two types of training sample, to ensure the smallest classification error rate. Huang et al. applied SVM to credit rating prediction using two datasets relating to Taiwan financial institutes and

United States commercial banks [19]; they found that SVM performed satisfactorily in their experiments. Lee applied SVM to corporate credit rating problem, and found that SVM (with the radial basis function kernel) had the best performance when compared to alternative baseline algorithms [20]. SVM has strong generalization ability and can efficiently handle high dimensional datasets, but is sensitive to parameter adjustment and the choice of function. Zhou et al. use direct search method to optimize the SVM-based credit scoring model, and the experimental results show that the performance and robustness of SVM is improved after this method [21]. Sebastián et al. proposed a profit-driven approach for classifier construction and simultaneous variable selection based on linear SVM in credit scoring [22].

B. SOFT CLASSIFICATION TECHNIQUES

While the application of single classification technology to credit scoring is common, it is undeniable that there are some limitations to it, such as low prediction performance and robustness. To address these limitations and combine the advantages of single classifiers, hybrid and ensemble-based soft classification techniques are introduced in this sub-section.

Hybrid classifier serially combine two or more heterogeneous machine learning techniques as different components. According to the analysis of Lin *et al.* [23], there are three ways to build hybrid classifier. The first approach is called cascaded hybrid classifier, meaning that each single classifier is connected serially to form a new classifier. More specifically, the output of the former classifier is used as the input to the next classifier, and so on. The second approach is the integration of clustering and classifiers, the first step of which is to either pre-classify the original data or to distinguish the main categories of the dataset by clustering; the clustering results are then used as the classifier input to generate the terminal prediction results. For example, Hsieh proposed a hybrid approach to the credit scoring problem, based on K-means clustering and neural network techniques [24]. He first applied the K-means clustering algorithm to generate new clusters and remove non-representative samples, and then used new classes of samples for further design of the credit scoring model. Huang et al. proposed a hybrid GA-SVM strategy which can simultaneously complement feature selection tasks, and optimize model parameters [25]. AghaeiRad et al. used the unsupervised learning based on self-organizing map (SOM) to improve the discriminant capability of feed-forward neural network, and it obtain better performance than the stand-alone FNN [26]. Hsu et al. tried to combine the artificial bee colony approach and SVM to enhance prediction performance of the credit ratings [27]. Zhang et al. also proposed a multi-stage hybrid model to enhance the prediction performance of credit scoring [7].

Classifier ensembles are developed by combining a number of classifiers in parallel, and are widely used in credit scoring. The idea of classifier ensembles can be expressed as a probabilistic framework as (1).

$$p(t\,|x) = \sum_{i=1}^{n} A_i p(t\,|(x, m_i)), \tag{1}$$

where t represents the value of prediction; $p(t\,|x)$ denotes the conditional distribution of t given an input variable x; m_i $(i = 1, 2, 3, \ldots n)$ indexes a set of possible models; A_i represents the probability of applying each model; $p(t\,|(x, m_i))$ denotes the conditional distribution of t given an input variable by applying a model m_i.

In the opinion of Wang *et al.* [28], ensemble learning is based on the machine learning approach, where several learning algorithms can be applied to one problem; this avoids the drawbacks of a single classifier and combines the advantages of multiple classifiers. Classifier ensembles can be divided into three main approaches: Bagging, Boosting, Stacking.

Bagging: This algorithm was developed by Breiman [29]. Bagging is one of the earliest ensemble learning algorithms based on the majority voting concept, where different training data subsets are randomly selected from the entire datasets and used to train the different base learners of the same type [28]. Random forest is one of the typical bagging algorithms.

Boosting: Boosting was proposed by Schapire [30]. Each training sample is given the same probability, and the datasets are then implemented with T iterations. After each iteration, the weight (resampling) of the samples with the wrong classification is increased, so that they are more focused in the next iteration. AdaBoost, firstly proposed by Schapire [30], Freund and Schapire [31], and Gradient Booting by Friedman [32], have been widely used as boosting algorithms in machine learning.

Stacking: Stacking was identified by Wolpert [33], and is another ensemble method whereby a model is essentially trained on top of models. This model combines the results of the individually trained models in different ways to produce a result. Since this combination model is general, stacking can effectively act in the same way as any other ensemble models.

According to Verikas *et al.* [34] and Lin *et al.* [23], although some recent research has focused on the development of a single classifier, soft classification techniques are currently the trends for credit scoring. However, existing research into hybrid classifier and classifier ensembles has mainly focused on the improvement of a specific step; for example, Min et al. examined the optimization of the classifier parameters [35], Chen and Li looked at feature selection [36], and Wang et al. focused on the integration of the models [28]. Lessmann et al. proved that the performance of heterogeneous ensembles is better than single models through a large-scale empirical analysis [2]. Xia et al. proposed a heterogeneous ensemble credit model that integrates the bagging algorithm with the stacking method to improve the prediction performance [37]. In our previous study [6], an imbalanced learning approach is combined with tree-based classifiers to construct an ensemble model, but the adaptive selection issues of the base classifiers have not been

addressed. A further extension by Abellán and Castellano is implied concerning the selection of best classifiers forming ensemble model in credit datasets which makes a step forward but still exits some limitations [38].

In order to compensate for the shortcomings of the above studies and to improve the self-adaptive adjustment ability of the existing method for different datasets, a novel multi-stage self-adaptive classifier ensemble model is proposed to enhance prediction performance and improve the self-adaptive adjustment ability.

III. MULTI-STAGE SELF-ADAPTIVE CLASSIFIER ENSEMBLE MODEL

In this section, the novel multi-stage self-adaptive classifier ensemble model is formulated. The proposed model can be divided into three main stages. In the first stage, the multi-step data preprocessing is employed to process the original data into standardized data and generate more representative features. In the second stage, the base classifier will be adaptively selected according to the prediction performance in Validation Set, and the parameters of the selected base classifiers will be optimized by Bayesian optimization algorithm [39]. In the third stage, the self-adaptive classifiers are integrated to obtain the final prediction results through multi-layer stacking and the optimization of PSO algorithm. The proposed model has not only the ability to self-adaptively adjust the base classifiers according to different datasets, but also to achieve better performance while integrating different classifiers. The framework of the proposed model is illustrated in Fig. 1. Here, the standardized data was obtained after the first stage and divided into three parts, namely, Training Set, Validation Set and Testing Set. clf_1 to clf_9 represent candidate classifiers in candidate classifier repository (CCR), clf_{t1} to clf_{t5} indicate the base classifiers selected for ensemble model, P1 to P5 represent the prediction result of the Training Set using top 5 classifiers, i.e., clf_{t1} to clf_{t5}. VP1 to VP5 represent the mean prediction result of the Validation Set using clf_{t1} to clf_{t5}. TP1 to TP5 represent the mean prediction result of the Testing Set using clf_{t1} to clf_{t5}. The proposed model is described in detail as follows.

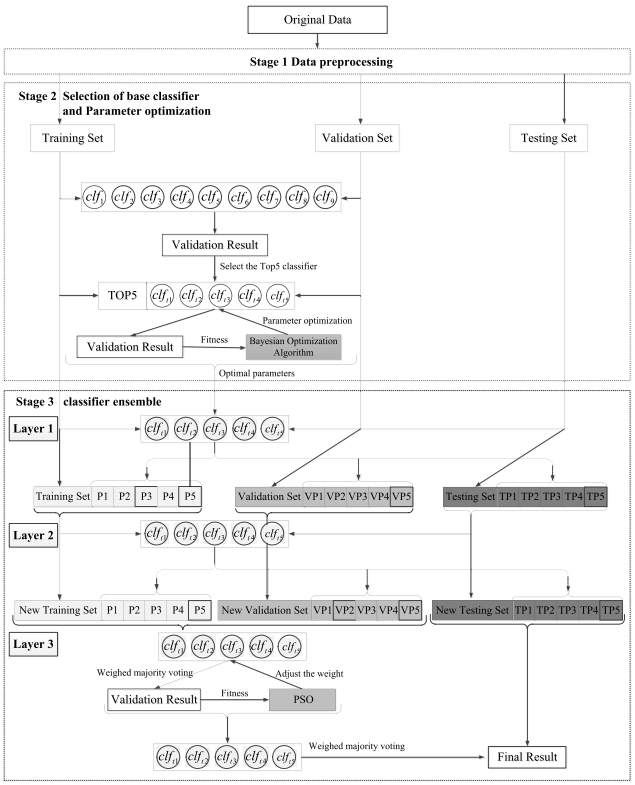


FIGURE 1. Framework of the proposed model.

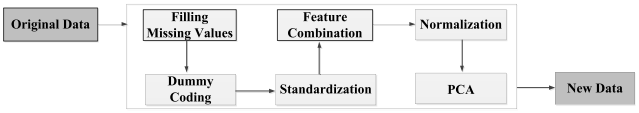


FIGURE 2. Multi-step data preprocessing process of the proposed model.

A. MULTI-STEP DATA PREPROCESSING

In practice, original data are sometimes irregularly awash with missing values and abnormal values which are unfavorable for further commutation and prediction of the proposed model. Original data also have some more representative features that have yet to be discovered. Therefore, data preprocessing plays a significant role in experimental process by making the data more representative and standardized. In this study, multi-step data preprocessing is constructed to generate more representative features and standardized data, as shown in Fig. 2. The process of multi-step data preprocessing includes six steps, namely, filling missing values, dummy coding, standardization, feature combination, normalization and Principal Component Analysis (PCA) [40]. After data preprocessing of the original data through these steps, new data (i.e. standardized data) is obtained. The details are as follows.

In the multi-step data preprocessing, the first step is filling missing values. The missing data in original dataset are firstly filled according to the type of features. That is, a new category is created to replace the missing values in categorical feature, and the mean value of numerical feature is used to replace the missing data in numerical feature. Then, the second step (dummy coding) is performed. By considering that the unordered and multi-categorical eigenvalues are incomparable, the dummy coding for dummy variable is used to quantify the non-quantitative variables. The feature with n categories can be transferred into n features which can be described as (2). In addition, the third step (standardization) is performed to eliminate numerical differences between features.

$$\begin{bmatrix} 1 \\ 2 \\ 3 \\ \dots \\ n \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}. \quad (2)$$

In data processing, many scholars [1], [28], [41]–[43] ignored the correlation between features, but it's useful to add such complexity to the proposed model by considering features correlation and nonlinear features of the input data. Therefore, the interrelationships of the features are taken into account through the fourth step (feature combination), which is achieved by constructing the feature polynomial to represent the association of features. For example, the general linear model is shown as (3). While the expression that considers the association between features is shown as (4).

$$y = w_0 + \sum_{i=1}^{n} w_i x_i, \tag{3}$$

$$y = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij} x_i x_j, \tag{4}$$

where w_0 is a constant term, n represents the number of features, x_i represents the ith feature, w_i represents a coefficient, and $x_i x_j$ represents a combination of x_i and x_j.

Further, the fifth step (normalization) is carried out so that the data of each dimension (feature) can be unified. At this time, the features of dataset become large with a number of ineffective or counterproductive features. In order to avoid curse of dimensionality and assure the mutual independence of variables, the sixth step - PCA is used to reduce the dimension, which can select several representative features in high dimensional space. PCA is a relatively effective method to reduce the dimension of the dataset, and its core is to transfer the original data into a set of linear independent eigenvectors through orthogonal transformation. After dimensionality reduction by PCA algorithms, the features whose dimensions are raised by polynomial become more representative for further experiment.

B. BASE CLASSIFIER SELECTION AND OPTIMIZATION

A popular approach to calculate similarity between scientific documents based on the traditional co-citation network is shown in (2).

The selection of the base classifiers is an important component of the ensemble model, and in general, it can be artificially regulated. For example, Hsieh and Hung selected neural network (NN), SVM, and Bayesian network as the base classifiers [44], while Wang et al. chose LR, DT, ANN, and SVM as base classifiers [28]. Ala'raj and Abbod selected RF, DT, NB, NN, and SVM [3]. However, it is assumed that the classifier is blunt, which leads to a model incapable of self-adaptively adjusting to a more suitable base classifier for different datasets, so that the result may not be optimal. To make a model classifier possess a self-adaptive selection capability for different datasets, the process of selecting suitable base classifier is constructed, as shown in Fig. 3. A CCR with nine classifiers ($clf_1 \sim clf_9$ in Fig. 3) is designed to select the optimal base classifiers for different data. By inputting the standardized data into CCR, the performance of each classifier in Validation Set is evaluated with the k evaluation

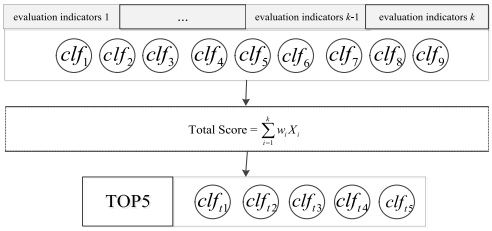


FIGURE 3. The process of selecting suitable base classifiers.

indicators. These evaluation indicators will be introduced in Section 4. Then the *Total Score* of each classifier is calculated and expressed in (5), reflecting the overall performance of the classifier. The top five classifiers with the highest total score will be selected as the base classifiers for the proposed model.

$$Total\ Score = \sum_{i=1}^{k} w_i X_i, \tag{5}$$

where k represents the number of evaluation indicators, X_i represents the performance of the ith evaluation indicator, and w_i is the ith weight of evaluation indicators.

Corresponding to different dataset, top five classifiers are self-adaptively selected to form the base classifiers for further ensemble modeling. Simultaneously, Bayesian Optimization Algorithm is adopted to determine the value of parameters for base classifiers to find out their optimum performance. Using the observed values, a regression model with Gaussian process is built, and the model is used to predict the mean value $\mu_{t-1}(x)$ and standard deviation $\sigma_{t-1}(x)$ on unknown input position. The position of the maximal sum of mean and the standard deviation is set to be the next point of sample. The acquisition function can be depicted as (6).

$$x_t = \underset{x \in D}{argmax}\ \mu_{t-1}(x) + \beta_t^{1/2} \sigma_{t-1}(x), \tag{6}$$

where $\beta_t^{1/2}$ is the parameter of the weight, and the parameters are set referring to [45].

In order to balance the exploitation and exploration, Upper Confidence Bound is used as the Acquisition Function in this study. The optimal parameters of base classifiers are calculated through the iteration of Bayesian optimization algorithm. Therefore, the parameters-optimized base classifiers are used for further ensemble modeling.

IV. CONSTRUCTING CLASSIFIER ENSEMBLES

Based on the optimized base classifiers in the above procedures, multi-classifiers stacking (MCS) method based on original stacking is proposed. It is described in detail as follows:

Firstly, selected base classifiers perform the first layer ensemble operation. The process details are shown in Fig. 4. In the process, the Training Set is divided into N folds for cross validation (e.g. $N = 5$ in Fig. 4). In each iteration, $N - 1$ folds (e.g. T in Fig. 4) are used to train selected base classifiers (e.g. clf_{t1} in Fig. 4), and the remaining one fold is

used for prediction (e.g. P in Fig. 4). Meanwhile, the trained base classifiers predict both Validation Set and Testing Set in each iteration. After N iterations, the prediction result for the whole training folds can be obtained. The mean prediction values in Validation Set and Testing Set are recognized as the global prediction results of the base classifiers in Validation Set and Testing Set, respectively (e.g. VP1 to VP5 and TP1 to TP5 in Fig. 4). Subsequently, these prediction results are added to the input dataset as new features, That is, the prediction results of base classifiers are integrated with the input dataset to form the new Training Set, Validation Set, and Testing Set in the first layer. Prediction result is presented as the right side of Fig. 4.

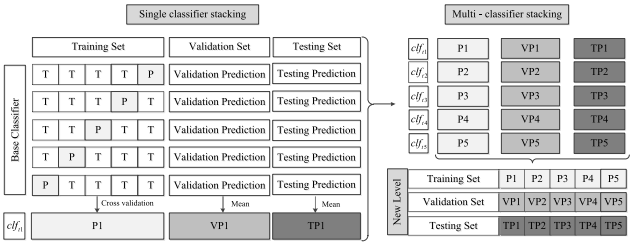


FIGURE 4. Process of first-layer classifier ensembles.

The new training dataset obtained in the first layer is used for the training input of the MCS procedure in the second layer to train base classifiers. The more layers the MCS produces, the higher dimensional data and more comprehensive features are obtained. In this study, two layers are used to verify the efficiency of the MCS.

With the aforementioned procedures, new Training Set, new Validation Set and new Testing Set are obtained. New Training Set is used to training the base classifiers and new Validation Set is used to validate the performance of the base classifiers, meanwhile, PSO algorithm is used to adjust the weight of the base classifiers, and the objective function of PSO algorithm is the comprehensive prediction performance (i.e. *Total Score* in (5)) in the Validation Set. It is initialized to a bunch of random particles (random solution), and searches for an optimized solution through iterations. In each iteration, the particles track two "extremum" (personal optimum p*best* and global optimum *gbest*) to update. In this procedure, particle i can be denoted as a vector with D dimensions with its position expressed as $X_i = (x_{i1}, x_{i2}, \ldots, x_{iD})^T$, velocity as $V_i = (v_{i1}, v_{i2}, \ldots, v_{iD})^T$. With the two optimums obtained, the particles can update the position and velocity as the following (7) and (8).

$$v_{id}^{k+1} = v_{id}^{k} + c_1 r_1^k (pbest_{id}^k - x_{id}^k) + c_2 r_2 (gbest_d^k - x_{id}^k), \quad (7)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1}, \quad (8)$$

where v_{id}^k is the velocity of the dth dimension of i in the kth iteration; c_1,c_2 is denoted as acceleration coefficient, usually assigned $c_1 = c_2 = 2$, and r_1,r_2 is random number belonging to [0, 1]; $pbest_{id}$ is the best previous position of the dth dimension of particle i; $gbest_d$ is the best previous position of the dth dimension for all the particles; x_{id}^k denotes the current position of the dth dimension of i in the iteration k.

Finally, the weights of the base classifiers through PSO algorithm are used as the weights of majority voting in ensemble. The proposed model uses the weighted majority voting method to predict the new Testing Set to obtain the final prediction results.

TABLE 1. Three credit datasets from the UCI repository.

Dataset	Number of instances	Good cases	Bad cases	categorical features	Numeric features	Total features
German	1000	700	300	13	7	20
Australian	690	307	383	6	8	14
Japanese	690	307	383	11	4	15

V. EXPERIMENTAL STUDY

A. CREDIT DATASETS

In the experiments, three real-world credit datasets obtained from the UCI machine learning repository [46], namely, German, Australian and Japanese, are employed to evaluate the performance of the proposed model. The details for the three datasets are shown in Table 1.

The German dataset consists of 1000 instances, of which 700 are positive (good cases), and 300 are negative (bad cases). Each instance contains 13 categorical features, seven numerical features, and a target attribute (accepted or rejected). Each feature represents a different meaning, such as credit history, repayment behavior, or employment stability.

The Australian dataset comprises 690 instances, of which 307 are positive (good cases), and 383 are negative (bad cases). Each instance contains six categorical attributes, eight numeric attributes, and a target attribute (accepted or rejected).

The Japanese dataset consists of 690 instances, of which 307 are positive (good cases), and 383 are negative (bad cases). Each instance contains 11 categorical features, four numerical features, and a target attribute (accepted or rejected). The composition of the Japanese dataset is similar to the Australian dataset, but has some missing values in its data.

B. EVALUATION INDICATORS

To estimate the performance of the proposed model, four evaluation indicators are used, namely, *Accuracy*, F_{score}, *AUC*, and *LogLoss*, and they are based on the confusion matrix shown in Table 2. According to Table 2, the true positives (TP) are positive instances predicted as positive, and the false negatives (FN) are positive instances predicted as negative; similarly, the false positives (FP) are negative instances predicted as positive, and the true negatives (TN) are negative instances predicted as negative. Based on these

TABLE 2. Confusion matrix.

		Predicted	
		Positive	Negative
Real	Positive	True Positives(TP)	False Negatives(FN)
	Negative	False Positives(FP)	True Negatives(TN)

factors, *Accuracy* and F_{score} are expressed as the following (9) and (10) [3], [47].

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}, \tag{9}$$

$$F_{score} = \frac{2 \times Precision \times Recall}{Precision + Recall}, \tag{10}$$

where $Precision = \frac{TP}{TP+FP}$ represents the proportion of the actual positive instances in all positive instances of the prediction, and $Recall = \frac{TP}{TP+FN}$ corresponds to the proportion of the actual positive instances in the positive class relative to the predicted positive class.

As described by Wang *et al.* [48], an ROC is a two-dimensional graph in which a true positive rate is plotted on the Y-axis and a false positive rate is plotted on the X-axis; the AUC is the area under the ROC curve, which can perfectly reflect the performance of the proposed model. In general, a model that has a larger AUC indicates better performance.

The first three indicators express the performance of prediction metrics positively, while *LogLoss* evaluates whether the classifier prediction is reliable, by measuring the loss. *LogLoss* is also known as cross-entropy loss, which is defined as the negative log-likelihood of the true labels, given a probabilistic classifier's predictions [49]; this is often used to evaluate the probability output of a classifier. For binary classification with true label $y \in \{0, 1\}$ and a probability estimation $p = \Pr(y = 1)$, the *LogLoss* per instance is the negative log-likelihood of the classifier given the true label. The formula is described as (11).

$$\begin{aligned} LogLoss(y, p) &= -\log \Pr(y \,|p\,) \\ &= -(y \log(p) + (1 - y) \log(1 - p)). \end{aligned} \tag{11}$$

VI. DATA PREPARATION AND PREPROCESSING

Before constructing a model, standardized dataset should be prepared through standardization and normalization. For the dataset (e.g., Japanese dataset) that contains missing values, the approach mentioned in Section 3 is adopted to fill the missing values of categorical features by new categories and fill the missing values of numerical features by mean value of corresponding features. After that, dummy coding, polynomial transformation method, and PCA will be applied on the dataset. The polynomial conversion formula with three characteristics and two degrees is depicted as (12).

$$\begin{aligned} &(x_1^{'}, x_2^{'}, x_3^{'}, x_4^{'}, x_5^{'}, x_6^{'}, x_7^{'}, x_8^{'}, x_9^{'}, x_{10}^{'}) \\ &= (1, x_1, x_2, x_3, x_1^2, x_1 * x_2, x_1 * x_3, x_2^2, x_2 * x_3, x_3^2). \end{aligned} \tag{12}$$

According to the preprocessing described in Fig. 2, a process of standardization and normalization is conducted according to the condition of the variance and characteristic for each feature. In brief, standardization applies z-score to transfer eigenvalue into same unit according to the columns of the feature matrix, which can be described as (13). As for normalization, it aims at setting unified standard when calculating the similarity by dot multiplication or other kernel functions, that is to say, transferring the value of each feature into a unit vector. L2 norm is adopted for normalization, and its formula is portrayed as following (14).

$$x^{'} = \frac{x - X''}{S}, \tag{13}$$

$$x^{'} = \frac{x}{\sqrt{\sum_{j=1}^{m} x[j]^2}}, \tag{14}$$

where $x^{'}$ represents the result value after processing, x represents the original value, X'' represents the mean of the column features, S represents the standard deviation of the column features, and m represents the dimensions of the feature.

In this study, experiments were repeated 30 times on each dataset to reduce the influence of contingency. The candidate classifiers in the CCR are KNN, LDA, LR, DT, SVM, MLP, RF, AdaBoost, and GBDT. Based on the characteristic of dataset and the practical experience, the weight ratio is set to: $w_{acc} : w_{auc} : w_{f1} : w_{loss} = 0.2 : 0.5 : 0.2 : -0.1$. A slight adjustment of the weight ratio has no effect on the final result. Dimensionality reduction with PCA is one of a key step of data preprocessing, and to minimize losses of information, 99.9% is set as a threshold to get dimensions by PCA. The dataset is separated into total Training Set and Testing Set with the proportion of 8:2. Specifically, in the experiment, the distribution of training data and test data for German is set to 800:200, and Australian and Japanese are the same, set to 552:138. To further observe the performance of the proposed model, the total Training Set is divided into Training Set and Validation Set with the same proportion. So the training data for German is further divided to be 640:160, and Australian and Japanese to be 442:110.

A. EVALUATION INDICATORS

Base classifiers strangled the throat of the performance of the ensemble model, so selection and optimization of the base classifiers is significant. In order to make full use of the dataset and minimize the impact of classic data limitation, cross-verification methods are applied to divide the training data into 5 folds, therein, 4 folds are used as the Training Set in the experiment and remaining one fold is used as the Validation Set. The evaluation metric is organized as (5), the mean value of total scores in five iterations is taken as the performance of classifier. As mentioned in Section 3, the parameters of selected classifiers will be optimized by Bayesian optimization algorithm with the iteration number set to 100. The classifiers used in the experiment are implemented

through the Scikit-learn[1] package. The parameters are optimized as follows: LR is optimized for the parameters C and *max_iter*, where C is the inverse of regularization strength, and the *max_iter* is the maximum number of iterations. SVM is optimized for the parameters C and γ, where C is the penalty parameter of the error term, and γ determines the distribution of the data to the new feature space. MLP is optimized for the parameters n, α, r, t, *max_iter*, m, $\beta_1\beta_1$, $\beta_2\beta_2$, where n is the hidden layer sizes, α is the penalty parameter, r is the learning rate, t is the exponent for the inverse scaling learning rate, *max_iter* is the maximum number of iterations, and m is the momentum for the gradient descent update. β_1 and β_2 are the exponential decay rates for estimates of the first and second moment vectors in Adam. RF is optimized for the parameter n, which is the number of trees in the forest. GBDT is optimized for the parameters m, r, n, and s, where m is the maximum depth of the individual regression estimators, r is the learning rate, n are the number of boosting stages to perform, and the fraction of samples s is to be used for fitting the individual base learners.

B. ENSEMBLE OF CLASSIFIERS

With adequate preparation of dataset and base classifiers, ensemble method is applied to obtain a better performance result. As aforementioned in Section 3, new Training Set, Validation Set and Testing Set are obtained after two-layer MCS method. After that, weighed majority voting is employed to work out a comprehensive result in Validation Set. Moreover, PSO algorithm is used to determine the weights of majority voting in ensemble. The dimension of particles in PSO algorithm is the number of base classifiers. The maximum number of iterations of the PSO algorithm is set to 100. When the PSO algorithm is performed more than 100 generations, the iteration ends and the weight of majority voting is output. The weighted majority voting is applied on new Testing Set to obtain the final result of the proposed model.

VII. EXPERIMENTS RESULTS AND ANALYSIS

In this section, experimental results are shown to illustrate the advantages of the proposed model. The proposed model was validated by four evaluation indicators, implemented on three datasets. All of the experiments used Python Version 3.6 on a PC with a 3.2 GHz Intel CORE i7 processor.

A. BENCHMARKING RESULTS

In the sub-section, Training Set and Validation Set are used to verify the performance of candidate classifiers; by observing their performance, top five classifiers are selected as the base classifiers for each dataset. The performance of each classifier on the different dataset is shown in Table 3, and changing tendency is intuitively lined in Fig. 5 - 7. These results are used as a benchmark for comparison in subsequent experiments

[1]http://scikit-learn.org

TABLE 3. Performance of each classifier for each dataset.

Datasets	Classifier	Performance measure metric				
		Accuracy	AUC	Fscore	LogLoss	Total Score
Australian	KNN	0.838	0.889	0.818	2.252	0.551
	LDA	0.866	0.922	0.841	0.369	**0.766**
	LR	0.865	0.936	0.852	0.330	**0.779**
	DT	0.830	0.829	0.811	5.857	0.157
	SVM	0.858	0.929	0.847	0.343	**0.771**
	MLP	0.867	0.939	0.851	0.324	**0.781**
	RF	0.862	0.935	0.861	0.783	0.734
	AdaBoost	0.843	0.908	0.828	0.651	0.723
	GBDT	0.858	0.930	0.843	0.345	**0.771**
German	KNN	0.706	0.633	0.804	3.908	0.228
	LDA	0.686	0.653	0.777	2.381	0.381
	LR	0.760	0.782	0.836	0.505	**0.660**
	DT	0.704	0.649	0.790	10.224	-0.399
	SVM	0.753	0.785	0.842	0.499	**0.662**
	MLP	0.743	0.750	0.823	0.842	**0.604**
	RF	0.740	0.753	0.821	0.963	0.593
	AdaBoost	0.716	0.746	0.822	0.664	**0.614**
	GBDT	0.762	0.781	0.843	0.504	**0.661**
Japanese	KNN	0.852	0.897	0.862	2.106	0.581
	LDA	0.858	0.911	0.859	0.553	0.744
	LR	0.854	0.931	0.858	0.347	**0.773**
	DT	0.812	0.812	0.825	6.507	0.083
	SVM	0.854	0.926	0.861	0.352	**0.771**
	MLP	0.867	0.929	0.872	0.658	**0.746**
	RF	0.834	0.920	0.867	0.507	**0.750**
	AdaBoost	0.859	0.912	0.845	0.638	0.733
	GBDT	0.861	0.931	0.869	0.350	**0.776**

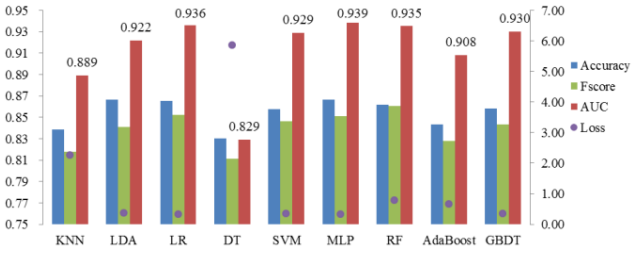


FIGURE 5. Performance comparisons of various classifiers in the Australian dataset.

Table 3 shows the results of each classifier operating on three datasets, in terms of the four evaluation indicators previously defined. Top five classifiers are highlighted in bold by comparing their total scores. Each dataset possess its own suitable base classifiers due to the different performance of these classifiers in variant dataset. In Australian dataset, top five classifiers are: MLP, LR, SVM, GBDT, LDA; in German dataset, top five classifiers are: LR, SVM, GBDT, AdaBoost, MLP; and in Japanese dataset, GBDT, LR, SVM, RF, MLP are ranked top five.

From Figs. 5 - 7, the Accuracy, AUC, and Fscore F_{score} of each classifier show the same trend. When one of the classifier performance measurement metrics has a larger value,

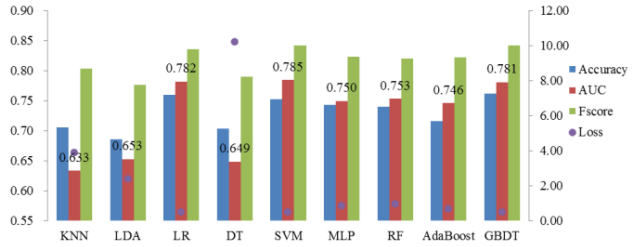


FIGURE 6. Performance comparisons of various classifiers in the German dataset.

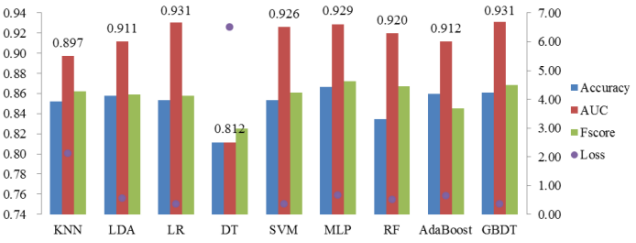


FIGURE 7. Performance comparisons of various classifiers in the Japanese dataset.

TABLE 4. Prediction result of optimized base classifiers in each dataset.

Dataset	Classifier	Accuracy	AUC	Fscore	LogLoss	Total Score
Australian	MLP	0.873	0.935	0.852	0.329	0.780
	LR	0.875	0.936	0.862	0.321	0.783
	SVM	0.875	0.932	0.862	0.328	0.781
	GBDT	0.864	0.928	0.853	0.518	0.756
	LDA	0.871	0.931	0.861	0.375	0.775
German	LR	0.765	0.787	0.835	0.504	0.663
	SVM	0.769	0.788	0.846	0.475	0.669
	GBDT	0.765	0.786	0.841	0.505	0.664
	AdaBoost	0.766	0.789	0.845	0.493	0.668
	MLP	0.771	0.793	0.844	0.503	0.669
Japanese	GBDT	0.862	0.934	0.870	0.346	0.779
	LR	0.858	0.932	0.863	0.339	0.776
	SVM	0.867	0.937	0.870	0.325	0.783
	RF	0.868	0.938	0.877	0.373	0.781
	MLP	0.862	0.932	0.869	0.362	0.776

the other two will also be larger and the LogLoss will be smaller. The parameters of each base classifier are optimized in accordance with Bayesian optimization algorithm mentioned in Section 3. Take the Australian dataset for an example: the value of parameter C for base classifier LR is set to 7.21 and the value of *max_iter* is set to 216.51; similarly, the value of parameter *learning_rate* for GBDT is set to 0.02 *max_depth* set to 9.96 and *n_estimators* set to 276.63; the parameters *alpha*, *beta_1*, *beta_2*, *hidden_layer_sizes*, *learning_rate_init*, *max_iter*, *momentum*, *power_t* for base classifier MLP are 0.54, 0.87, 0.95, 50.05, 0.02, 298.45, 0.87 and 0.32, respectively; the parameter C for SVM is set to 8.92 and *gamma* is to 0.006; the remaining classifier LAD is selected with its parameter *tol* set to 0.001.

TABLE 5. Performance comparisons of SVM before and after optimization.

Datasets	Classifier	Accuracy	AUC	Fscore	LogLoss	Total Score
Australian	SVM_ori	0.858	0.929	0.847	0.343	0.771
	SVM_opt	0.875	0.932	0.862	0.328	0.781
German	SVM_ori	0.753	0.785	0.842	0.499	0.662
	SVM_opt	0.769	0.788	0.846	0.475	0.669
Japanese	SVM_ori	0.854	0.926	0.861	0.352	0.771
	SVM_opt	0.867	0.937	0.870	0.325	0.783

TABLE 6. Final result of the proposed model with three datasets.

Dataset	Accuracy	AUC	Fscore	LogLoss	Total Score
Australian	0.874	0.940	0.868	0.320	0.786
German	0.783	0.806	0.850	0.474	0.682
Japanese	0.870	0.942	0.873	0.327	0.787

B. PREDICTION RESULTS OF OPTIMIZED BASE CLASSIFIERS

Table 4 shows the prediction results of the base classifiers after parameter optimization in three datasets. These prediction results of optimized base classifiers are effectively improved when compared to the original prediction results of the former classifier. This reflects that the application of Bayesian optimization algorithm in the proposed model is effective.

To more intuitively understand the parameter optimization utility, the results of SVM parameter optimization are compared for an example. According to Table 5, it can be found that the prediction results of the optimized SVM are improved under each evaluation indicator. Compared to the aforementioned benchmark results, in the Australian dataset, the prediction result of SVM improved 1.78% on Accuracy, 0.29% on AUC, 1.58% on Fscore, and 1.58% on LogLoss; in the German dataset, the prediction result of SVM improved 1.64% on Accuracy, 0.27% on AUC, 0.38% on Fscore, and 2.41% on LogLoss; in the Japanese dataset, the prediction result of SVM improved 1.31% on Accuracy, 1.06% on AUC, 0.91% on Fscore, and 2.67% on LogLoss.

C. THE FINAL PREDICTION RESULTS

With two-layer MCS and weighed majority voting, the final prediction result is worked out and listed in Table 6. According to the experimental results, final result of the proposed model is superior to those of the base classifiers, which demonstrates the importance and effectiveness of model ensemble processes. Although the multi-stage ensemble strategy increases the computational complexity to some extent in the proposed model, the costs are acceptable compared to the economic benefits due to the prediction performance improvements. The cost of time and effort required by the multi-stage ensemble strategy in the offline training stage is large but within the acceptable range, while the cost of time and effort in the online prediction stage is little. It costs

TABLE 7. Comparison of performance with other credit scoring models.

Method	Evaluation indicators	Australian	German	Japanese
Our proposed model	Accuracy	0.874	0.783	0.870
	AUC	0.940	0.806	0.942
Teng et al.'s work [47]	Accuracy	0.870	0.732	
Abellán and Mantas 's work [41]	AUC	0.934	0.785	0.929
Sadatrasoul et al.'s work [50]	Accuracy	0.848	0.735	
Chen et al.'s work [51]	Accuracy	0.860	0.760	

less than 1 second to predict an instance in the experiment. At present, the costs of computer hardware resources have been reduced to some extent, and with the development of cloud computing, the cost of time and effort in this study will be reduced more rapidly.

In recent years, some scholars [41], [47], [50], [51] have also proposed new models to improve the performance of credit scoring in these datasets. In order to verify the validity of the proposed model, the prediction results of the proposed model are compared with these state of the art models under the same experimental conditions, and the results are shown in Table 7. The comparison results show that the prediction performance of the proposed model is more robust in different datasets, and can achieve good performance in different datasets. This also shows the effectiveness of the proposed model.

VIII. CONCLUSIONS AND FUTURE WORKS

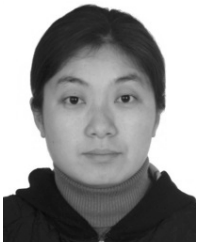
In recent years, machine learning technology has made rapid development, and ensemble learning also has been widely studied. Some researches have shown that ensemble methods have been recognized as prevalent modeling techniques [6], [36]. In this study, a multi-stage self-adaptive classifier ensemble model is proposed, which combines multiple effective components and selects appropriate base classifiers for different credit scoring data to improve the prediction performance. In the proposed model, base classifiers can be self-adaptively selected from CCR according to the characteristics of different datasets, and their parameters are optimized by Bayesian optimization algorithm. The proposed model can generate new features through multi-layer stacking and obtain the final weight of base classifiers through the PSO algorithm. In the experimental study, four evaluation indicators are used to determine the reliability of proposed model on three real-world datasets from UCI. The results show that compared to single classifier and other ensemble classifiers, the proposed model has better robustness and better prediction performance. It clearly illustrates the validity and usefulness of the proposed model.

Although the proposed model has better robustness and prediction performance compared to other models, it still has some deficiencies. In future works, several issues will be considered. First, the datasets used in this study are small, and some larger datasets will be used to verify the validity of the proposed model. Second, heuristic algorithms will be used for base classifier selection to make it more intelligent. In addition, dynamic prediction of the default lender risk is another possible further research direction.

REFERENCES

[1] C.-F. Tsai and C. Hung, "Modeling credit scoring using neural network ensembles," *Kybernetes*, vol. 43, no. 7, pp. 1114–1123, 2014.
[2] S. Lessmann, B. Baesens, L. C. Thomas, and H.-V. Seow, "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research," *Eur. J. Oper. Res.*, vol. 247, no. 1, pp. 124–136, Nov. 2015.
[3] M. Ala'raj and M. F. Abbod, "Classifiers consensus system approach for credit scoring," *Knowl.-Based Syst.*, vol. 104, pp. 89–105, Jul. 2016.
[4] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4. Nov. 1995, pp. 1942–1948.
[5] D. J. Hand and W. E. Henley, "Statistical classification methods in consumer credit scoring: A review," *J. Roy. Stat. Soc., A*, vol. 160, pp. 523–541, Sep. 1997.
[6] H. He, W. Zhang, and S. Zhang, "A novel ensemble method for credit scoring: Adaption of different imbalance ratios," *Expert Syst. Appl.*, vol. 98, pp. 105–117, May 2018.
[7] W. Zhang, H. He, and S. Zhang, "A novel multi-stage hybrid model with enhanced multi-population niche genetic algorithm: An application in credit scoring," *Expert Syst. Appl.*, vol. 121, pp. 221–232, May 2019.
[8] F. Hoffmann, B. Baesens, J. Martens, F. Put, and J. Vanthienen, "Comparing a genetic fuzzy and a neurofuzzy classifier for credit scoring," *Int. J. Intell. Syst.*, vol. 17, no. 11, pp. 1067–1083, 2002.
[9] F. Hoffmann, B. Baesens, C. Mues, T. Van Gestel, and J. Vanthienen, "Inferring descriptive and approximate fuzzy rules for credit scoring using evolutionary algorithms," *Eur. J. Oper. Res.*, vol. 177, no. 1, pp. 540–555, 2007.
[10] C.-S. Ong, J.-J. Huang, and G.-H. Tzeng, "Building credit scoring models using genetic programming," *Expert Syst. Appl.*, vol. 29, pp. 41–47, Jul. 2005.
[11] C.-M. Wang and Y.-F. Huang, "Evolutionary-based feature selection approaches with new criteria for data mining: A case study of credit approval data," *Expert Syst. Appl.*, vol. 36, pp. 5900–5908, Apr. 2009.
[12] D. W. Hosmer and S. Lemeshow, "Goodness of fit tests for the multiple logistic regression model," *Commun. Statist.*, vol. 9, no. 10, pp. 1043–1069, 1980.
[13] T.-C. Tang and L.-C. Chi, "Predicting multilateral trade credit risks: Comparisons of Logit and Fuzzy Logic models using ROC curve analysis," *Expert Syst. Appl.*, vol. 28, pp. 547–556, Apr. 2005.
[14] S. Y. Sohn, D. H. Kim, and J. H. Yoon, "Technology credit scoring model with fuzzy logistic regression," *Appl. Soft Comput.*, vol. 43, pp. 150–158, Jun. 2016.
[15] L. C. Thomas, J. N. Crook, and D. B. Edelman, Eds., "Credit scoring and credit control," in *Methods for Credit Scoring Applied to Slow Payers*. Oxford, U.K.: Oxford Univ. Press, 1989, pp. 75–90.
[16] T.-S. Lee, C.-C. Chiu, Y.-C. Chou, and C.-J. Lu, "Mining the customer credit using classification and regression tree and multivariate adaptive regression splines," *Comput. Statist. Data Anal.*, vol. 50, pp. 1113–1130, Feb. 2006.
[17] M. Siami, M. R. Gholamian, and J. Basiri, "An application of locally linear model tree algorithm with combination of feature selection in credit scoring," *Int. J. Syst. Sci.*, vol. 45, pp. 2213–2222, Oct. 2014.
[18] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.
[19] Z. Huang, H. Chen, C.-J. Hsu, W.-H. Chen, and S. Wu, "Credit rating analysis with support vector machines and neural networks: A market comparative study," *Decis. Support Syst.*, vol. 37, pp. 543–558, Sep. 2004.
[20] Y.-C. Lee, "Application of support vector machines to corporate credit rating prediction," *Expert Syst. Appl.*, vol. 33, pp. 67–74, Jul. 2007.
[21] L. Zhou, K. K. Lai, and L. Yu, "Credit scoring using support vector machines with direct search for parameters selection," *Soft Comput.*, vol. 13, pp. 149–155, Jan. 2009.
[22] S. Maldonado, C. Bravo, J. López, and J. Pérez, "Integrated framework for profit-based feature selection and SVM classification in credit scoring," *Decis. Support Syst.*, vol. 104, pp. 113–121, Dec. 2017.
[23] W.-Y. Lin, Y.-H. Hu, and C.-F. Tsai, "Machine learning in financial crisis prediction: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 4, pp. 421–436, Jul. 2012.

[24] N.-C. Hsieh, "Hybrid mining approach in the design of credit scoring models," *Expert Syst. Appl.*, vol. 28, pp. 655–665, May 2005.
[25] C.-L. Huang, M.-C. Chen, and C.-J. Wang, "Credit scoring with a data mining approach based on support vector machines," *Expert Syst. Appl.*, vol. 33, pp. 847–856, Nov. 2007.
[26] A. AghaeiRad, N. Chen, and B. Ribeiro, "Improve credit scoring using transfer of learned knowledge from self-organizing map," *Neural Comput. Appl.*, vol. 28, pp. 1329–1342, Jun. 2017.
[27] F.-J. Hsu, M.-Y. Chen, and Y.-C. Chen, "The human-like intelligence with bio-inspired computing approach for credit ratings prediction," *Neurocomputing*, vol. 279, pp. 11–18, Mar. 2017.
[28] G. Wang, J. Hao, J. Ma, and H. Jiang, "A comparative assessment of ensemble learning for credit scoring," *Expert Syst. Appl.*, vol. 38, pp. 223–230, Jan. 2011.
[29] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.
[30] R. E. Schapire, "The strength of weak learnability," *Mach. Learn.*, vol. 5, no. 2, pp. 197–227, 1990.
[31] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. 13rd Int. Conf. Mach. Learn.* Bari, Italy: Morgan Kaufman, Jul. 1996, pp. 148–156.
[32] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001.
[33] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992.
[34] A. Verikas, Z. Kalsyte, M. Bacauskiene, and A. Gelzinis, "Hybrid and ensemble-based soft computing techniques in bankruptcy prediction: A survey," *Soft Comput.*, vol. 14, pp. 995–1010, Jul. 2010.
[35] S.-H. Min, J. Lee, and I. Han, "Hybrid genetic algorithms and support vector machines for bankruptcy prediction," *Expert Syst. Appl.*, vol. 31, pp. 652–660, Oct. 2006.
[36] F.-L. Chen and F.-C. Li, "Combination of feature selection approaches with SVM in credit scoring," *Expert Syst. Appl.*, vol. 37, no. 7, pp. 4902–4909, 2010.
[37] Y. Xia, C. Liu, B. Da, and F. Xie, "A novel heterogeneous ensemble credit scoring model based on bstacking approach," *Expert Syst. Appl.*, vol. 93, pp. 182–199, Mar. 2018.
[38] J. Abellán and J. G. Castellano, "A comparative study on base classifiers in ensemble methods for credit scoring," *Expert Syst. Appl.*, vol. 73, pp. 1–10, May 2017.
[39] R. Martinez-Cantin, "BayesOpt: A Bayesian optimization library for nonlinear optimization, experimental design and bandits," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3735–3739, 2014.
[40] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philos. Mag.*, vol. 2, no. 11, pp. 559–572, 1901.
[41] J. Abellán and C. J. Mantas, "Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring," *Expert Syst. Appl.*, vol. 41, pp. 3825–3830, Jun. 2014.
[42] W. Li, S. Ding, Y. Chen, and S. Yang, "Heterogeneous ensemble for default prediction of peer-to-peer lending in China," *IEEE Access*, vol. 6, pp. 54396–54406, 2018.
[43] C. Wang, D. Han, Q. Liu, and S. Luo, "A deep learning approach for credit scoring of Peer-to-Peer lending using attention mechanism LSTM," *IEEE Access*, vol. 7, pp. 2161–2168, 2018.
[44] N.-C. Hsieh and L.-P. Hung, "A data driven ensemble classifier for credit scoring analysis," *Expert Syst. Appl.*, vol. 37, pp. 534–545, Jan. 2010.
[45] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Proc. 27th Int. Conf. Mach. Learn.* Haifa, Israel: Morgan Kaufman, Jun. 2010, pp. 1015–1022.
[46] A. Asuncion and D. J. Newman, "UCI machine learning repository," School Inf. Comput. Sci., Univ. California, Irvine, CA, USA, 2007. [Online]. Available: http://www.ics.uci.edu/~mlearn/MLRepository.html
[47] G.-E. Teng, C.-Z. He, J. Xiao, and X.-Y. Jiang, "Customer credit scoring based on HMM/GMDH hybrid model," *Knowl. Inf. Syst.*, vol. 36, pp. 731–747, Sep. 2013.
[48] J. Wang, A.-R. Hedar, S. Wang, and J. Ma, "Rough set and scatter search metaheuristic based feature selection for credit scoring," *Expert Syst. Appl.*, vol. 39, pp. 6123–6128, May 2012.
[49] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 19–67, 2005.
[50] S. Sadatrasoul, M. Gholamian, and K. Shahanaghi, "Combination of feature selection and optimized fuzzy apriori rules: The case of credit scoring," *Int. Arab J. Inf. Technol.*, vol. 12, pp. 138–145, Mar. 2015.
[51] N. Chen, B. Ribeiro, and A. Chen, "Comparative study of classifier ensembles for cost-sensitive credit risk assessment," *Intell. Data Anal.*, vol. 19, pp. 127–144, Jan. 2015.



SHANSHAN GUO received the College Diploma degree in accounting from the Anhui University of Economic Management, in 1996. She is currently a Lecturer with the Library, Zhejiang University of Finance and Economics, China. She has published five papers in international journals in the recent five years, covering a wide range of data mining, business intelligence, and manufacturing automation.

HONGLIANG HE is currently pursuing the M.S. degree with the School of Information, Zhejiang University of Finance and Economics, China. His current research interests include ensemble learning and artificial intelligence.

XIAOLING HUANG received the master's degree in physical education from Zhejiang University, China, in 2014. She is currently a Lecturer with the School of International Education, Zhejiang University of Finance and Economics, China. She has published three papers in international journals in the recent three years, covering a wide range of data mining and artificial intelligence and supply chain optimization.