Applied Soft Computing 43 (2016) 73–86

Contents lists available at ScienceDirect

Applied Soft Computing

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

Ensemble classification based on supervised clustering for credit scoring

Hongshan Xiao, Zhi Xiao, Yu Wang*

School of Economics and Business Administration, Chongqing University, Chongqing 400030, China

ARTICLE INFO

Article history:
Received 12 December 2014
Received in revised form 26 January 2016
Accepted 12 February 2016
Available online 27 February 2016

Keywords:
Credit scoring
Ensemble classification
Random sampling
Supervised clustering
Diversity of base classifiers
Weighted voting

ABSTRACT

Credit scoring aims to assess the risk associated with lending to individual consumers. Recently, ensemble classification methodology has become popular in this field. However, most researches utilize random sampling to generate training subsets for constructing the base classifiers. Therefore, their diversity is not guaranteed, which may lead to a degradation of overall classification performance. In this paper, we propose an ensemble classification approach based on supervised clustering for credit scoring. In the proposed approach, supervised clustering is employed to partition the data samples of each class into a number of clusters. Clusters from different classes are then pairwise combined to form a number of training subsets. In each training subset, a specific base classifier is constructed. For a sample whose class label needs to be predicted, the outputs of these base classifiers are combined by weighted voting. The weight associated with a base classifier is determined by its classification performance in the neighborhood of the sample. In the experimental study, two benchmark credit data sets are adopted for performance evaluation, and an industrial case study is conducted. The results show that compared to other ensemble classification methods, the proposed approach is able to generate base classifiers with higher diversity and local accuracy, and improve the accuracy of credit scoring.

1. Introduction

Credit scoring aims to assess the risk associated with lending to individual consumers. With the market of credit products increasing enormously in recent years, accurate credit scoring models have aroused unprecedented attention of financial institutions since they could reduce the operational risk and cost, and increase the competitive ability in the market. Even 1% of improvement on the accuracy of recognizing applicants would greatly increase the profit of financial institutions [1].

In this paper, we address the application scoring problem among the four kinds of credit scoring problems, as summarized in Paleologo et al. [2]. Application scoring refers to the assessment of credit worthiness of new applicants. It quantifies the risk of credit requests with respect to their characteristics such as age, income, and occupation. Essentially, this problem can be considered as a general population classification task [3]. In other words, a classification model (sometimes also called a classifier) is needed to classify the credit requests into good credit or bad credit, based on their characteristics. Therefore, various classification techniques have been applied, which can be divided into two broad categories: statistical techniques and artificial intelligence (AI) techniques [4].

Among statistical techniques, the most popular methods are linear discriminant analysis (LDA) and logistic regression (Logit) [5,6]. LDA and Logit are relatively easy to implement and generate interpretable results. However, in real world applications, assumptions for the data such as being linear separable and following certain distributions are often violated. In recent years, it has been demonstrated that AI techniques are alternative and effective methods for credit scoring. AI techniques include *k*-nearest neighbors (KNN) [7], mathematical programming [8,9], artificial neural network (ANN) [10], artificial immune systems (AIS) [11,12], genetic algorithm (GA) [13], support vector machines (SVMs) [14–17], and their variations [18].

In most of these researches, only one classification model (classifier) is constructed for credit scoring. It has been shown theoretically and experimentally that ensemble classification tends to be an effective methodology for improving the accuracy and stability of a single classifier [19,20]. For credit scoring, Lim and Sohn [21] argue that a single classification rule would be ineffective because it cannot catch the fine nuance of various individual consumers. Therefore, it is beneficial to introduce ensemble classification to credit scoring.

* Corresponding author. Tel.: +86 23 65105261; fax: +86 23 65105261.
E-mail address: yuwang@cqu.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.asoc.2016.02.022
1568-4946/

In the framework of ensemble classification, a pool of different base classifiers, instead of a single one, is constructed and combined to predict the class label of unknown sample. The main idea of ensemble classification is to take advantage of the base classifiers and avoid their weakness. For effective ensemble classification, it is required that base classifiers in the ensemble are accurate and diverse [22–26]. With respect to credit scoring, diversity of base classifiers could not only improve the accuracy of ensemble classification, but also provide useful and insightful information of individual consumers' behavioral patterns even when they are in the same class (either good credit or bad credit). Unfortunately, despite the fact that many researches on ensemble classification and their applications to credit scoring have been carried out, little attention has been paid to the diversity of base classifiers.

In this paper, we propose an Ensemble Classification approach based on Supervised Clustering (ECSC) for credit scoring. We focus primarily on how to construct diverse base classifiers for improving the ensemble classification performance. In the proposed approach, supervised clustering is employed to partition the data samples of each class into a number of clusters. Clusters from different classes are then paired to form a number of training subsets for constructing base classifiers. The idea behind supervised clustering is that data samples from the same class may have different patterns or characteristics. Though supervised clustering, samples with similar patterns or characteristics are grouped into the same cluster. Therefore, the training subsets, obtained by pairwise combination of clusters from different classes, could better represent different patterns of samples, which is helpful to increase the accuracy and diversity of base classifiers. Another advantage of the propose ECSC is that we do not have to set the number of base classifiers in advance, which avoids the possible adverse impact of human intervention on ensemble classification performance. In supervised clustering, *K*-means clustering is adopted since it is effective for partitioning large data set. Besides, a validity index is developed to optimize the clustering result. For a sample whose class label needs to be predicted, the outputs of different base classifiers are combined by weighted voting. The weight associated with a base classifier is determined by its classification performance in the neighborhood of the sample. In the experimental study, two benchmark data sets and a real world mortgage loan data set are used to validate the proposed ECSC for credit scoring.

The rest of this paper is organized as follows. Section 2 reviews the related work on ensemble classification methodologies and their applications to credit scoring. In Section 3, the proposed ESCS approach is elaborated with analytical discussions. Section 4 reports the experimental study and results. In Section 5, an industrial case study is conducted. The paper ends with some concluding remarks in Section 6.

2. Related work

During the last two decades, various ensemble classification methods, as well as their applications to credit scoring, have been developed. In what follows, we first review the work on ensemble classification methodologies, and then investigate the researches on credit scoring, especially those based on ensemble classification.

2.1. Ensemble classification methodologies

In ensemble classification, the best known methods are Bagging [27], Random Subspace Method (RSM) [28], and Boosting [29]. In Bagging, a number of subsets for constructing different base classifiers are generated by randomly sampling from the original data set. Similar to Bagging, Boosting also uses subsets to train classifiers, but not randomly. In Boosting, difficult samples have higher probabilities of being selected for training, and easier samples have less chance of being used, so that most base classifiers focus on difficult samples, which would increase the classification accuracy. Different from random sampling in the data set, the RSM constructs different base classifiers based on different feature subsets in the feature space.

Currently, how to select an optimal set of base classifiers has become an important direction [30,31]. Liu and Yuan [32] propose an ensemble classification algorithm based on clustering and selection. In their study, clustering is employed to partition the feature space, and then different classifiers are constructed. The output of the best classifier with response for the vicinity of the input instance is adopted as the final ensemble result. Ruta and Gabrys [33] provide a classifier selection methodology, in which a number of search algorithms, such as greedy search, stochastic hill-climbing search and evolutionary algorithm, are employed to search for the best set of competent classifiers for ensemble classification. Xiao et al. [34] propose a novel dynamic classifier ensemble selection strategy by introducing a group method of data handling that considers both accuracy and diversity in the process of ensemble selection. Woloszynski and Kurzynski [35] use a probabilistic model to calculate the competence of base classifiers, and select the output of the most competent classifier, or use majority voting rule to predict the class label. Li et al. [36] focus on how to combine the outputs from base classifiers for efficient ensemble classification. In their work, a subset of classifiers is dynamically selected based on the optimization of margin distribution on the training data set. Inspired by the fact that simply abandoning the classifiers which fail in the competition of ensemble selection would causing a considerable waste of useful resources and information, Dai et al. [37] develop a greedy reverse reduce-error algorithm incorporated with subtraction operation for ensemble pruning. Mendialdua et al. [38] propose a novel approach to select the individual classifiers by means of an evolutionary algorithm, which is based on Estimation of Distribution Algorithms (EDAs) [39].

2.2. Credit scoring based on ensemble classification

In the field of credit scoring based on ensemble classification, Paleologo et al. [2] propose a subagging approach for credit scoring based on traditional Bagging, where the base classifiers are generated by random sub-sampling in order to deal with class imbalance problem. Finlay [40] investigates the accuracy of various multiple classifier systems, and concludes that Bagging and Boosting deliver better performance. Combining ensemble classification with cost-sensitive learning, Xiao et al. [41] propose a dynamic classifier ensemble method for imbalanced credit scoring data. Wang et al. [4] present a hybrid ensemble classification method based on Bagging and Random Subspace, and apply it to credit scoring. In their work, decision tree is used as the classifier. Kruppa et al. [42] estimate the default probabilities of consumers using random forests. Tsai et al. [43] conduct a comprehensive study of comparing classifier ensembles for credit scoring, and conclude that decision trees ensemble using the boosting method performs the best.

As can be seen from the above review, the existing researches on ensemble classification and their application to credit scoring mainly utilize random sampling (either in the instance space or in the feature space) for constructing base classifiers. They focus on how to generate accurate base classifiers, how to combine their results, or how to dynamically select the best base classifiers for better ensemble performance. Unfortunately, little attention has been paid to the diversity of the generated base classifiers. This limitation may lead to a degradation of overall classification performance, since an efficient ensemble classification requires that base classifiers have diversity in their predictions.

3. Ensemble classification based on supervised clustering

The main idea of the proposed ECSC is to use the clustering analysis to partition the samples of each class into a number of clusters, so that all samples in a cluster are from the same class. Clusters from different classes are then paired to form the training subsets for constructing base classifiers. Since in this approach, clustering procedure is guided by the class labels, it is called "supervised clustering".

3.1. Supervised clustering

Suppose a data set S consists of N samples $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_N$, where $\mathbf{X}_i = \left[x_1^i, x_2^i, \ldots, x_d^i\right]$, $(i = 1, 2, \ldots, N)$ is a d-dimension vector. Each sample $\mathbf{X}_i$ has a label y_i indicating which class $\mathbf{X}_i$ belongs to. Without loss of generality, we only consider two-class problem in this paper, which means $y_i \in \{+1, -1\}$. Consequently, the data set contains two subsets S^+ and S^-. Samples in S^+ share the same class label $y = +1$, and those in S^- have class label $y = -1$. Different from conventional clustering that partitions the whole data set into a number of groups, supervised clustering partitions samples from the same class.

K-means clustering is a very popular algorithm particularly suitable for partitioning large amount of objects [44]. Therefore, we adopt it in supervised clustering. In K-means clustering, an important problem is to evaluate the fitness of partitions produced by the clustering algorithms. A key to this problem is to find the optimal number of clusters, which is usually called cluster validity [45]. Until recently, many cluster validity indexes have been proposed to determine the optimal number of clusters and evaluate the goodness of clustering algorithm, as summarized in Rezaee [46] and Zhang et al. [47]. In this paper, we adopt the validity function proposed by Wu and Yang [48], in which two main aspects of clustering validity are considered, i.e., compactness of each cluster and separation between clusters. However, in their work, the validity function is for soft K-means clustering. In order to fit it to hard K-means clustering for supervised clustering, we modify the validity function as follows.

Assume the data set S consisting of N samples $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_N$ (class labels are not considered in unsupervised clustering) is partitioned into K clusters $C_1, C_2, \ldots, C_K$. The centroids of these clusters are denoted by $\mathbf{r}_0^1, \mathbf{r}_0^2, \ldots, \mathbf{r}_0^K$, in which $\mathbf{r}_0^k (k = 1, 2, \ldots, K)$ is defined as

$$\mathbf{r}_0^k = \sum_{i=1}^{m_k} \mathbf{X}_i^k / m_k, \tag{1}$$

where m_k is the number of samples in cluster C_k, and $\mathbf{X}_i^k (i = 1, 2, \ldots, m_k)$ denotes the ith sample that belongs to the cluster C_k. The index of compactness of clusters obtained is defined as

$$Intra = \frac{1}{K} \sum_{k=1}^{K} Intra(k). \tag{2}$$

In formula (2), the term $Intra(k)$ represents the relative distance between each sample and the centroid, with respect to the maximal distance between all samples and the centroid, in the kth cluster. It is defined as

$$Intra(k) = \left(\sum_{i=1}^{m_k} \left| \mathbf{X}_i^k - \mathbf{r}_0^k \right| \right) / \left(m_k \max_{j \in [1, m_k]} \left| \mathbf{X}_j^k - \mathbf{r}_0^k \right| \right). \tag{3}$$

Intuitively, a compact cluster requires the term $Intra$ defined in formula (2) be small. Since $0 < Intra(k) \leq 1$, it can be easily seen that $0 < Intra \leq 1$. For clusters with only one sample, we define $Intra(k) = 1$.

For a good clustering result, not only each cluster obtained is compact, but also these clusters are well separated from each other. For this purpose, we define the index of separation between clusters as follows

$$Inter = \exp(-D/\beta), \tag{4}$$

in which $D = 2 \sum_{1 \leq i < j \leq K} |\mathbf{r}_0^i - \mathbf{r}_0^j| / [K(K-1)]$ is the average distance between each pair of cluster centroids, and $\beta = \sum_{k=1}^{K} \left| \mathbf{r}_0^k - \mathbf{r}_0 \right| / K$ is the average distance of each cluster centroid $\mathbf{r}_0^k$ to the center of all cluster centroids $\mathbf{r}_0$. Apparently, $0 < Inter \leq 1$, and a smaller $Inter$ indicates that clusters obtained are more separated from each other.

The objective of determining the optimal value of K is to find out the optimal K clusters each of which is compact and separated from others. Therefore, the validity function considering both the compactness and separation of clusters is defined as follows

$$VF(K) = Intra \times Inter, \tag{5}$$

in which $Intra$ and $Inter$ are defined in formulas (2) and (4), respectively. Clearly, the smaller the $VF(K)$ is, the better the clustering results are.

Based on the validity function defined in formula (5), the algorithm of supervised clustering is shown in Fig. 1.

In Fig. 1, $|\cdot|$ denotes the cardinality of a data set. The maximal number of clusters for N samples is set to $\sqrt{N}$, as suggested by Ramze et al. [49].

3.2. Ensemble classification methodology

By supervised clustering, we obtain K_{opt}^+ positive subsets and K_{opt}^- negative subsets. Subsets from different classes are then paired to form a number of training subsets for constructing base classification. The main idea of supervised clustering and pairwise combination is to generate different training subsets, and consequently, construct diverse base classifiers in these subsets. Moreover, supervised clustering is particularly favorable for ensemble classification, since it explores the spatial characteristics

Algorithm 1: supervised clustering

Step 1: Initialize the positive subset S^+ and the negative subset S^-;

Step 2: Initialize $K_{\min}^+ = 2$, $K_{\min}^- = 2$, $K_{\max}^+ = \sqrt{|S^+|}$, $K_{\max}^- = \sqrt{|S^-|}$;

Step 3: For $K^+ = K_{\min}^+ : K_{\max}^+$

{

Apply the K-means clustering algorithm to S^+ to obtain K^+ clusters;

Compute $VF(K^+)$ according to (5);

} // End For

Step 4: For $K^- = K_{\min}^- : K_{\max}^-$

{

Apply the K-means clustering algorithm to S^- to obtain K^- clusters;

Compute $VF(K^-)$ according to (5);

} // End For

Step 5: Find $K_{opt}^+ = \underset{K_{\min}^+ \leq K^+ \leq K_{\max}^+}{\operatorname{argmin}} VF(K^+)$, $K_{opt}^- = \underset{K_{\min}^- \leq K^- \leq K_{\max}^-}{\operatorname{argmin}} VF(K^-)$;

Step 6: Output K_{opt}^+ positive clusters and K_{opt}^- negative clusters.

Fig. 1. Algorithm of supervised clustering (for binary classification).

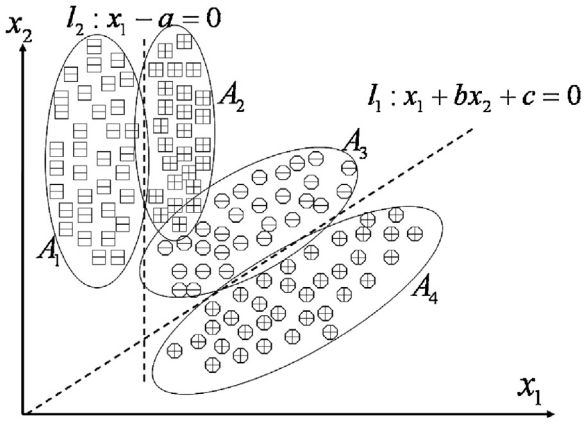

Fig. 2. Illustration of supervised clustering for ensemble classification.

of samples in each class, which is a key issue for better classification performance, as stated Vucetic and Obradovic [50]. Therefore, base classifiers could achieve high local accuracy.

To better explain the benefit of supervised clustering for ensemble classification, we consider a two-dimension binary classification problem illustrated in Fig. 2.

As shown in Fig. 2, the data set S consists of all square-shape and circle-shape points. The "+" and "−" inside the points indicate their class labels. For example, "+" means the class label y =+1. Obviously, for negative subset A_1 and positive subset A_2, the optimal classifier is $y = sign(x_1 - a)$, while for negative subset A_3 and positive subset A_4, the optimal classifier is $y = -sign(x_1 + bx_2 + c)$. Therefore, a single classifier is difficult to discriminate one class from the other with high accuracy. If we construct a pool of base classifiers using random sampling, it is highly probable that the randomly selected samples come from all subsets, which could lead to a performance degradation of base classifiers. Moreover, the diversity of base classifiers is not guaranteed.

Supervised clustering, in which the clustering procedure is supervised by class labels, is effective for dealing with the above problems. Through supervised clustering, the positive sample points can be partitioned into two subsets A_2 and A_4, and similarly, the negative sample points would be partitioned into two subsets A_1 and A_3. These subsets represent different spatial characteristics of samples in different classes. By pairwise combination and base classifiers construction in the training subsets, the two optimal classifiers described above can be obtained. Besides, they are diverse with respect to their predictions.

Pairwise combination of positive and negative subsets would generate $K^+_{opt}K^-_{opt}$ training subsets. Therefore, the number of base classifiers (NC) constructed on these subsets is $NC = K^+_{opt}K^-_{opt}$, and we do not have to determine the number of base classifiers in advance. This is another advantage of our approach that avoids the possible adverse impact of human intervening on classification.

Besides accuracy and diversity of base classifiers, another critical issue in ensemble classification is the integration mechanism, i.e., how the outputs of base classifiers are integrated [19]. Compared to major voting using all base classifiers or selecting only one optimal base classifier, dynamic classifier selection, which explores the use of different base classifiers for different test sample, is a better choice [20]. Therefore, we in this paper use the weighted voting based on the classifier performance in local area, which is a dynamic classifier selection method.

Denote by TS_1, TS_2, …, TS_{NC} the training subsets obtained through supervised clustering and pairwise combination, and H_1, H_2, …, H_{NC} the constructed base classifiers in TS_1, TS_2, …, TS_{NC}. For a sample $\mathbf{X}$ with unknown class label, denote by $\mathbf{X}_{NN(1)}$, $\mathbf{X}_{NN(2)}$, …, $\mathbf{X}_{NN(M)}$ the M training samples nearest to $\mathbf{X}$, and l_1, l_2, …, l_{NC}

Algorithm 2: ensemble classification based on supervised clustering

Step 1: Apply algorithm 1 to data set S to obtain the positive clusters $S^+_1, S^+_2, \cdots, S^+_{K^+_{opt}}$ and negative clusters $S^-_1, S^-_2, \cdots, S^-_{K^-_{opt}}$;

Step 2: Initialize $j = 1$;

Step 3: For $i = 1 : K^+_{opt}$

{ For $k = 1 : K^-_{opt}$

{ Set $TS_j = S^+_i \bigcup S^-_k$;

Construct a base classifier H_j in subset TS_j;

Set $j = j + 1$;

} // End For

} // End For

Step 4: For a sample $\mathbf{X}$, compute the voting weight $W(j)$ of each base classifier $H_j, (j = 1, 2, \cdots, NC)$ according to (6).

Step 5: Output the predicted class label of $\mathbf{X}$ according to (7).

Fig. 3. Algorithm of ensemble classification based on supervised clustering.

the accuracy (in percentage) of base classifiers H_1, H_2, …, H_{NC} on $\mathbf{X}_{NN(1)}$, $\mathbf{X}_{NN(2)}$, …, $\mathbf{X}_{NN(M)}$, the performance of each base classifier H_j in the neighborhood of $\mathbf{X}$ can be measured by l_j. Therefore, the voting weight of base classifier H_j(j = 1, 2, …, NC) on the sample $\mathbf{X}$ is set as

$$W(j) = l_j / \sum_{i=1}^{NC} l_i. \tag{6}$$

For the sample $\mathbf{X}$, if base classifier H_j predicts its class label as $y = C_i$, a binary variable $v_{i,j}$ is set to be 1, else 0. Therefore, the weighted voting $\sum_{j=1}^{NC} v_{i,j} W(j)$ represent the probability that $\mathbf{X}$ belongs to class C_i. According to the voting rule, the predicted class of ensemble classification for $\mathbf{X}$ is

$$C_i = \arg_i \max \sum_{j=1}^{NC} v_{i,j} W(j). \tag{7}$$

In summary, the algorithm of the proposed ensemble classification approach based on supervised clustering is shown in Fig. 3.

As can be seen from Fig. 3, the proposed ECSC is a meta-level ensemble classification methodology where any specific classification technique can be embedded. To validate its effectiveness and efficiency, in this study, three different kinds of classification techniques, i.e., logistic regression (Logit), decision tree (DT) and support vector machine (SVM), are employed as the base methods for classifiers construction. Logistic regression is a typical statistical method widely applied to many practical classification problems. Decision tree is one of the most popular classification algorithms in data mining. Support vector machine is a state-of-the-art machine learning technique, which has proven its performance in many applications such as credit scoring and bankruptcy prediction. For more details of decision tree we refer the readers to [51–55].

4. Experimental study

4.1. Experimental setup

In order to validate the proposed ECSC approach for credit scoring, German and Australian data sets are adopted as the benchmark data sets in the experimental study. These two data sets are

widely used in credit scoring researches and can be easily downloaded from UCI Machine Learning Repository (http://www.ics.uci.edu/~mlearn/MLRepository.html). German data set consists of 1000 customer records, 700 with good credit and 300 with bad credit. Each record includes 20 attributes (7 numerical attributes and 13 qualitative attributes), and 1 class label with two different states: good credit and bad credit. Australian data set consists of 700 customer records, 307 with good credit and 393 with bad credit. Each record includes 14 attributes (8 numerical attributes and 6 nominal attributes), and 1 class label indicating good or bad credit.

In the experiments, the widely used Bagging and RSM for ensemble classification are used as the benchmarking methods. In both methods, the outputs of base classifiers are combined through major voting (MV) as the final result. To further investigate the performance of the proposed approach, we use different combining mechanisms, i.e., OLA, LCA, KNORA-Eliminate, and KNORA-Union, as described in Ko et al. [20]. The basic idea of OLA scheme is to estimate each individual classifier's accuracy in local regions surrounding a test sample, and then use the decision of the most locally accurate classifier. The LCA mechanism is similar to the OLA, the only difference being that the local accuracy is estimated in respect of output classes. Different from OLA and LCA mechanisms where only one optimal base classifier is selected, the mechanism KNORA (*K*-Nearest-Oracles) selects the most suitable subset of base classifiers for each test sample. KNORA-Eliminate mechanism figures out which classifiers correctly classify all the neighbors of a given test sample and uses them as the ensemble for classification with equal weights. Compared with KNORA-Eliminate, KNORA-Union uses classifiers that correctly classify any of the *K*-nearest neighbors. The more neighbors a classifier classifies correctly, the more votes this classifier will have for a test pattern except for major voting, all mechanisms need to set the number of nearest neighbors *M*. In this study, we arbitrarily set this parameter as $M = 10$.

In addition to Bagging and RSM, two other state-of-the-art ensemble classification methodologies are utilized as the competitors. The first one, proposed by Wang et al. [4], is a hybrid ensemble classification method based on Bagging and Random Subspace (RS). In their work, two dual ensemble strategies, namely Bagging-RS and RS-Bagging, are applied to credit scoring. The second one is the DCE-CC proposed by Li et al. [36], in which a subset of base classifiers is dynamically selected and combined with weighted voting according to their confidence.

In the performance evaluation of classification methods, cross validation is often used to obtain robust and consistent predictions for instances with unknown class labels [56]. In this procedure, data set is randomly partitioned into two subsets: a training set and a testing set. The training set is used to construct the classification models, and the testing data is used to test their accuracies. To minimize the influence of the variability caused by random partitioning on classification results, each data set is randomly partitioned into a training set and a testing set with proportions 80% and 20%, respective, and 30 runs of cross validation is implemented on the Australian and German data sets in this study. The average misclassification rate over 30 runs is regarded as the overall classification performance.

Since the values of different features lie within different ranges, to avoid the dominance of large-scale features over small-scale features in clustering analysis, we normalize each feature via its estimates of mean and variance before supervised clustering. For feature $x_l (l = 1, 2, \ldots, d)$, the normalized value $\hat{x}_{i,l}$ of the ith sample is

$$\hat{x}_{i,l} = (x_{i,l} - \bar{x}_l)/\sigma_l, \tag{8}$$

where $x_{i,l}$ is the original value of feature x_l of the ith sample. $\bar{x}_l$ and σ_l are the estimated mean and standard deviation of feature x_l, respectively. By the above normalization, all the resulted features $\hat{x}_l (l = 1, 2, \ldots, d)$ have zero mean and unit variance.

Table 1
Misclassification rates of ECSC.

Classifier	Number of base classifiers	Average misclassification rate (%)	
		Australian	German
DT	50.27/29.63[a]	12.38 ± 2.06	23.37 ± 1.85
Logit	50.27/29.63[a]	11.95 ± 1.89	22.24 ± 2.35
SVM	50.27/29.63[a]	13.14 ± 1.85	29.40 ± 2.64

[a] The number of base classifiers in Australian/German data sets.

The experiments are performed on a PC with a 3.20 GHz Intel i5 CPU and 4 GB RAM, and Matlab 7.0 is used for coding. The implementation of logistic regression and support vector machine are directly adopted from our previous studies [52,55] with small modifications, such that the optimal regularization parameter *C* of support vector machine is set by cross validation among discrete values 0.01, 0.1, 0.5, 1, 5, 10, 50, 100. Decision tree and *K*-means clustering are implemented by using the toolkit boxes of Matlab 7.0.

4.2. Experimental results

In the experimental study, we mainly focus on three important aspects: the performance of different ensemble classification methods, the diversity of the generated based classifiers, and the influence of diversity and integration mechanisms on the ensemble performance.

4.2.1. Performance of ensemble classification methods

We compare the proposed ECSC to Bagging and RSM with respect to classification performance. The misclassification rates of ECSC with different kinds of classifiers over 30 runs of cross validation on German and Australian data sets are shown in Table 1. Note that the number of base classifiers is automatically generated by multiplying the optimal numbers of clusters from positive and negative classes. Therefore, it remains the same for different kinds of classifiers. For Australian data set, the number of base classifiers of ECSC averaged over 30 runs of cross validation is 50.27, while for German data set, the number is 29.63.

In both Bagging and RSM, the number of base classifiers has to be set. In order to investigate the impact of base classifier number on the classification result, for Australian data set, we set the number of base classifiers to 10, 50 (approximately equal to that in ECSC), and 100 in Bagging and RSM. For German data set, we set the number to 10, 30 (approximately equal to that in ECSC), and 50. The misclassification rates of Bagging and RSM with different kinds of classifiers over 30 runs of cross validation on German and Australian data sets are shown in Figs. 4 and 5, where the dotted lines represent the misclassification rates of ECSC.

Table 1, Figs. 4 and 5 show that compared to Bagging and RSM with different number of base classifiers, ECSC delivers better classification performance in most cases. Figs. 4 and 5 also indicate that the number of base classifiers has non-negligible impact on the classification result, but not well-regulated. For example, in Australian data set, the misclassification rate of Bagging-DT, with LCA combining mechanism, decreases as the number of base classifiers increases, but this does not hold for other combining mechanisms. This is in accordance with the analytical and experimental results presented in Zhou et al. [57] that increase of ensemble size does not necessarily improve the classification accuracy. In contrast, ECSC does not have such problems. Recall that the number of base classifiers is automatically generated by multiplying the optimal numbers of clusters from positive and negative classes. Therefore,

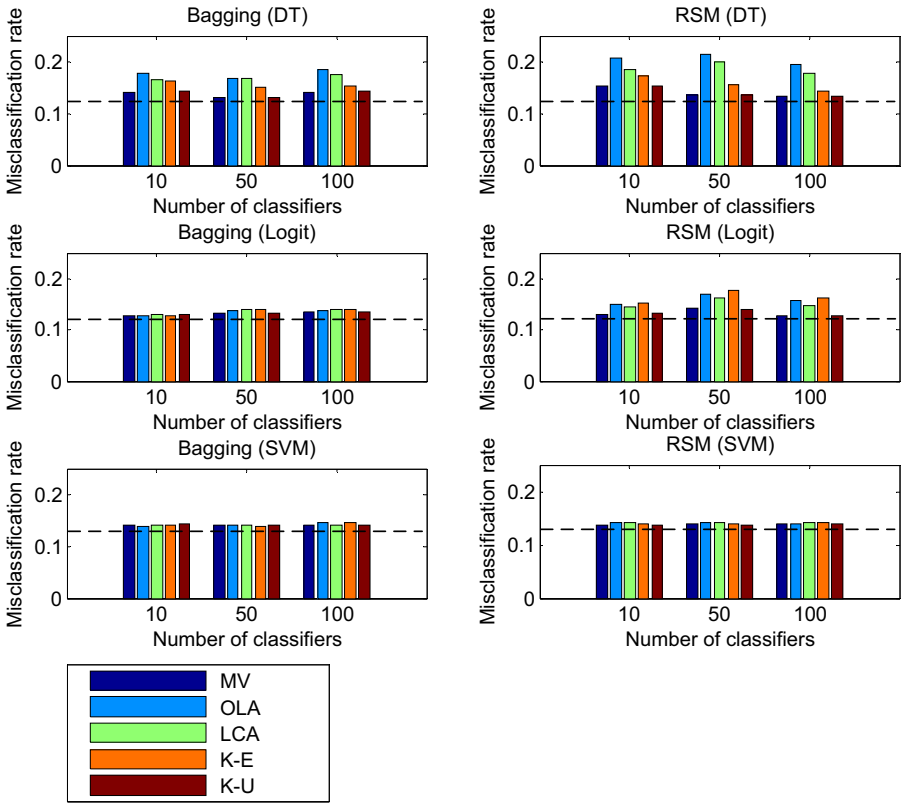

Fig. 4. Average misclassification rates of Bagging and RSM with different numbers of base classifiers over 30 runs (Australian data set).

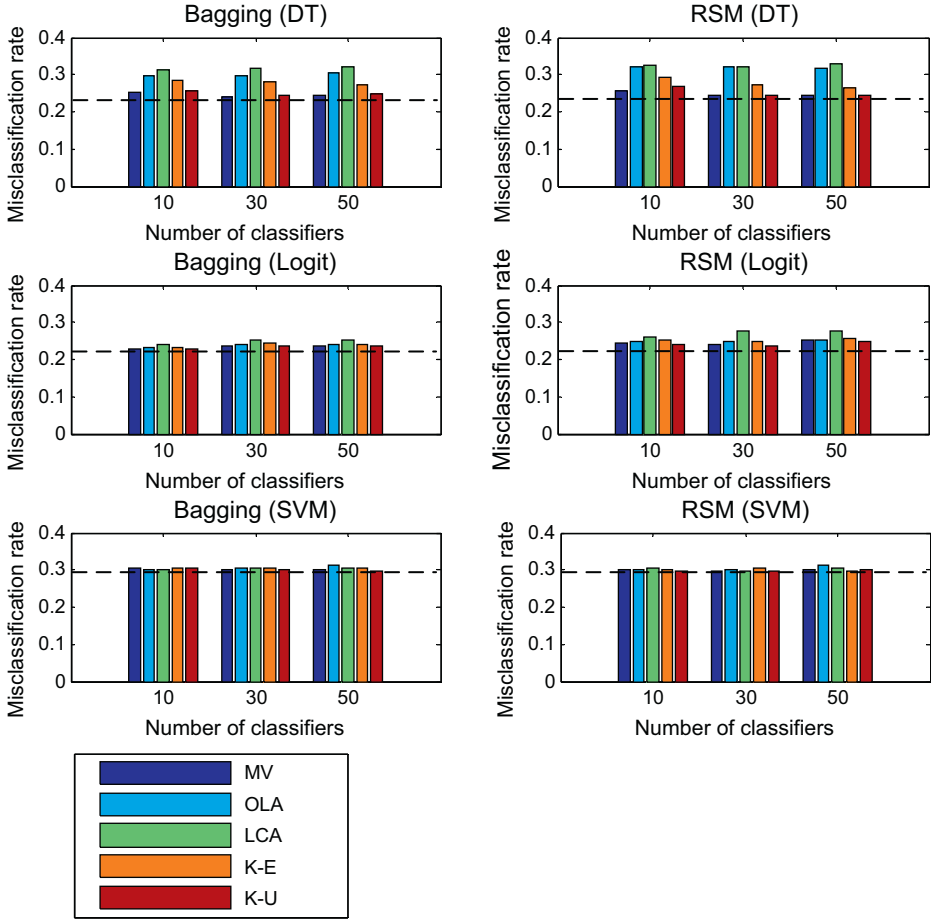

Fig. 5. Average misclassification rates of Bagging and RSM with different numbers of base classifiers over 30 runs (German data set).

it avoids the possible performance degradation caused by empirically setting the number of base classifiers without any prior or domain knowledge.

In what follows, we compare the proposed ECSC to other two state-of-the-art ensemble classification methodologies, i.e., dual ensemble classification strategies Bagging-RS and RS-Bagging, and DCE-CC. In Bagging-RS and RS-Bagging, there are several parameters that have to be pre-determined, such as number of learning rounds T_B for Bagging, number of learning rounds T_R for Random Space, and number of selected features rate k. It is stated in Wang et al. [4] that RS-Bagging and Bagging-RS have different accuracies with different ensemble sizes and subspace rates. For the purpose of simplicity, the optimal parameters corresponding to different data set in their work are directly adopted. For example, Bagging-RS gets the best results when the ensemble size is 100 and subspace rate is 0.7.

DCE-CC employs double rotation and bootstrap sampling to generate base classifiers and dynamically selects a subset of classifiers for test samples according to classification confidence. Double rotation incorporates two steps, i.e., Principal Component Analysis (PCA) and Locality Sensitive Discriminant Analysis (LSDA), for feature extraction. Although feature extraction is not the focus of this study, we still follow these steps for the completeness of the experiments. In DCE-CC, the number of base classifiers L, the number of subsets G, and the sampling ratio γ have to be set in advance. According to Li et al. [36], L, G, and γ are set to 100, 2 and 0.75, respectively.

The misclassification rates of Bagging-RS, RS-Bagging, DCE-CC and ECSC with different kinds of classifiers over 30 runs of cross validation on German and Australian data sets are shown in Tables 2–4. The results show that in comparison with Bagging-RS, RS-Bagging and DCE-CC, the proposed ECSC achieves the lowest misclassification rates in most cases, although in some cases the improvement is marginal with no statistical significance (at the 0.05 level). It is noted that for Australian data set, the performance of DCE-CC with logit as the classifier is slightly better than that of ECSC. This may be attributed to the feature extraction procedure in DCE-CC that employs PCA and LSDA to get uncorrelated features for logistic regression.

Since we perform 30 replications of 2-fold cross validation and the training data sets over different replications may not be completely independent and disjoint, it is necessary to consider the combined 5×2 cv F-test [26,58–60]. In what follows, we perform another five replications of 2-fold cross validation. In each run, the data set is divided into two equal-sized sets. Denote by $p_i^{(j)}$ the difference between the error rates of two classification methods on fold $j = 1, 2$ of replication $i = 1, 2, \ldots, 5$, the average on replication i is $\bar{p}_i = (p_i^{(1)} + p_i^{(2)})/2$, and the corresponding estimated variance is $s_i^2 = (p_i^{(1)} - \bar{p}_i)^2 + (p_i^{(2)} - \bar{p}_i)^2$. The statistic f defined below is approximately F distributed with 10 and 5 degrees of freedom, and the hypothesis that the two classification models have the same error rate with 0.95 confidence level would be rejected if the statistic f is greater that 4.74.

$$f = (N/10)/(M/5) = \left(\sum_{i=1}^{5} \sum_{j=1}^{2} \left(p_i^{(j)} \right)^2 \right) / \left(2 \sum_{i=1}^{5} s_i^2 \right). \quad (9)$$

We compute the statistic f between the error rates of the proposed ECSC and three other competing models, i.e., Bagging-RS, RS-Bagging, and DCE-CC. The results are shown in Table 5.

Table 5 shows that except for Australian data set with DT and SVM as the classifiers, the misclassification rates of ECSC are not statistically significant (with 0.95 confidence level) from those of Bagging-RS, RS-Bagging and DCE-CC. However, it can be found that in most cases, the statistic f is greater that 3.3, which corresponds to the 0.90 confidence level. Overall, the experimental results suggest that the proposed ECSC is an effective and promising approach for credit scoring.

4.2.2. Diversity of base classifiers

Efficient ensemble classification requires that base classifiers are not only accurate, but also have diversity in their predictions [23,25]. Tsymbal et al. [19] present five different measures of the diversity of base classifiers, i.e., Plain Disagreement Measure, Fail/Non-Fail Disagreement Measure, Q Statistic, Correlation Coefficient, and Kappa D Degree-of-Agreement Measure. Since the calculations of the first, second and fourth measures are similar, and the first measure equals to the second measure for binary

Table 2
Misclassification rates of Bagging-RS, RS-Bagging, DCE-CC and ECSC (Logit as classifier).

Method	Number of base classifiers[a]	Average misclassification rate (%)	
		Australian	German
Bagging-RS	100/100	12.73 ± 1.76**	23.05 ± 2.53
RS-Bagging	100/150	12.58 ± 1.43**	23.00 ± 2.04**
DCE-CC	100/100	11.73 ± 1.91	22.63 ± 2.55
ECSC	50.27/29.63	11.95 ± 1.89	22.24 ± 2.35

[a] The number of base classifiers in Australian/German data sets.
** The paired-t test result (vs. ECSC) is significant at the 0.05 level.

Table 3
Misclassification rates of Bagging-RS, RS-Bagging, DCE-CC and ECSC (DT as classifier).

Method	Number of base classifiers[a]	Average misclassification rate (%)	
		Australian	German
Bagging-RS	100/100	13.07 ± 2.42	24.12 ± 2.07
RS-Bagging	100/150	13.24 ± 2.22	24.17 ± 2.09
DCE-CC	100/100	13.31 ± 2.22**	24.74 ± 2.23**
ECSC	50.27/29.63	12.38 ± 2.06	23.37 ± 1.85

[a] The number of base classifiers in Australian/German data sets.
** The paired-t test result (vs. ECSC) is significant at the 0.05 level.

Table 4
Misclassification rates of Bagging-RS, RS-Bagging, DCE-CC and ECSC (SVM as classifier).

Method	Number of base classifiers[a]	Average misclassification rate (%)	
		Australian	German
Bagging-RS	100/100	14.18 ± 2.19**	30.97 ± 2.52**
RS-Bagging	100/150	13.99 ± 2.01	31.09 ± 2.64**
DCE-CC	100/100	13.75 ± 0.17**	30.01 ± 2.38
ECSC	50.27/29.63	13.14 ± 1.85	29.40 ± 2.64

[a] The number of base classifiers in Australian/German data sets.
** The paired-t test result (vs. ECSC) is significant at the 0.05 level.

Table 5
Results of combined 5×2 cv F-test between misclassification rates of Bagging-RS, RS-Bagging, DCE-CC and ECSC.

Methods	Classifiers	f statistic	
		Australian	German
Bagging-RS vs. ECSC	Logit	2.4000	1.7758
	DT	4.0397	3.1242
	SVM	3.9563	1.9401
RS-Bagging vs. ECSC	Logit	2.6153	1.8496
	DT	3.3254	3.3328
	SVM	3.8354	2.7851
DCE-CC vs. ECSC	Logit	4.7142	2.1666
	DT	6.1081**	3.8517
	SVM	6.4190**	4.0408

** The result is significant with 0.95 confidence level.

classification, we only adopt the first, third and fifth measures. In what follows, these three measures are denoted by Div-Plain, Div-Q and Div-Kappa, respectively.

For two classifiers i and j, denote by N the number of samples in the data set, and $C_i(\mathbf{x}_k)$ the class assigned by classifier i to instance k, the measure Div-Plain is the proportion of the instances on which the classifiers make different predictions:

$$\text{Div-Plain}_{i,j} = \sum_{k=1}^{N} \text{Diff}(C_i(\mathbf{x}_k), C_j(\mathbf{x}_k))/N, \tag{10}$$

where $\text{Diff}(a,b)=0$, if $a=b$, otherwise $\text{Diff}(a,b)=1$. When the classifiers i and j return the same classes for each instance, this measure is equal to 0, and it is equal to 1 when the predictions are always different.

Denote by N^{ab} the number of instances in the data set classified correctly ($a=1$) or incorrectly ($a=0$) by the classifier i, and correctly ($b=1$) or incorrectly ($b=0$) by the classifier j. The measure Div-Q is based on Yule's statistic used to assess the similarity of two classifiers output [23]:

$$\text{Div-}Q^*_{i,j} = (N^{11}N^{00} - N^{01}N^{10})/(N^{11}N^{00} + N^{01}N^{10}). \tag{11}$$

This measure varies from -1 to 1, where -1 corresponds to the maximal diversity and 1 corresponds to the minimal diversity. In this study, we normalize it to vary from 0 to 1, where 1 corresponds to the maximal diversity, as represented in Tsymbal et al. [19]:

$$\text{Div-}Q_{i,j} = (1 - \text{Div-}Q^*_{i,j})/2. \tag{12}$$

Let N_{ij} be the number of instances in the data set, recognized as class i by the first classifier and as class j by the second one, N_{i*} is the number of instances recognized as i by the first classifier, and N_{*i} is the number of instances recognized as i by the second classifier. Define θ_1 and θ_2 as

$$\theta_1 = \sum_{i=1}^{l} N_{ii}/N, \quad \theta_2 = \sum_{i=1}^{l} (N_{i*}N_{*i})/N^2, \tag{13}$$

where l is the number of classes. The pairwise diversity $\text{Div-Kappa}_{i,j}$ is defined as

$$\text{Div-Kappa}_{i,j} = (\theta_1 - \theta_2)/(1 - \theta_2). \tag{14}$$

The value of this measure varies from -1 to 1, where 1 corresponds to the minimal diversity. In this study, we normalize this measure to vary from 0 to 1 in the same way we do for $\text{Div-}Q_{i,j}$ in (12).

It is noteworthy that the above three measures are quantify the diversity between each pair of base classifiers. Therefore, the diversity of the ensemble is the averaged value over all of the pairs of base classifiers in the ensemble, as given in:

$$\begin{aligned} \text{Div-Plain} &= \sum_{i \neq j} \text{Div-Plain}_{i,j}/(nc(nc-1)/2), \\ \text{Div-}Q &= \sum_{i \neq j} \text{Div-}Q_{i,j}/(nc(nc-1)/2), \end{aligned} \tag{15}$$

where nc is the number of base classifiers. With respect to Div-Kappa, since we only consider the binary classification problem, which means $l=2$, the overall diversity measure is define as

$$\text{Div-Kappa} = \sum_{i \neq j} \text{Div-Kappa}^{i,j}_{1,2}/(nc(nc-1)/2), \tag{16}$$

where $\text{Div-Kappa}^{i,j}_{1,2}$ is the pairwise diversity measure in (14). The superscript i,j refers to two different base classifier i and j, and the subscript 1, 2 refers to the two classes.

Table 6
Diversity measures of base classifiers in ECSC over 30 runs.

Data set	Classifier	Measure		
		Div-Plain	Div-Q	Div-Kappa
Australian	Logit	0.2415	0.2494	0.4982
	DT	0.4721	0.4781	0.4998
	SVM	0.3585	0.3390	0.4997
German	Logit	0.3711	0.3778	0.4998
	DT	0.5000	0.5037	0.4997
	SVM	0.2002	0.1890	0.4997

We first investigate the diversity of base classifiers in ECSC. The three diversity measures of base classifiers in ECSC over 30 runs are illustrated in Table 6.

In order to investigate the impact of base classifiers number on the diversity, we set the number of base classifiers in Bagging and RSM to 10, 50, and 100 for German data set, and 10, 30, and 50 for Australian data set, as we have done in the above experiments. The diversity measures Div-Plain, Div-Q and Div-Kappa in Bagging and RSM with different number of base classifiers are shown in Figs. 6 and 7.

By comparing the diversity measures of base classifiers in ECSC, as shown in Table 6, to those in Bagging and RSM shown in Figs. 6 and 7, we can find that with respect to the diversity measures Div-Plain and Div-Q, diversity of base classifiers in ECSC is much larger than those in Bagging and RSM in all cases.

With respect to diversity measure Div_Kappa, ECSC, Bagging and RSM are similar to each other with small fluctuation. This is mainly because the measure Div_Kappa considers the proportions of samples classified to each class by different base classifiers over the whole data set, while the measure Div-Plain considers the diversity of different base classifier on each sample, and Div-Q reflects the extent to which different base classifiers make different predictions on the samples. Recall that base classifiers in ECSC are generated through supervised clustering and pairwise combination, and aimed to achieve higher classification accuracy in corresponding training subsets instead of all training data set. Therefore, the diversity measures Div-Kappa of ESCS, Bagging and RSM of these methods are similar.

We next investigate the diversity of base classifiers in Bagging-RS, RS-Bagging and DCE-CC. The parameters, such as the number of learning rounds for Bagging, the number of learning rounds for Random Space, and the number of subsets, of these methods are set the same as in the above experiments. The diversity measures of base classifiers in Bagging-RS, RS-Bagging and DCE-CC over 30 runs on different data sets are shown in Fig. 8. For the sake comparison, the diversity measures of ECSC shown in Table 6 are included.

Fig. 8 indicates the similar result that with respect to the diversity measures Div-Plain and Q statistic, diversity of base classifiers in ECSC is much larger than other methods in all cases. With respect to diversity measure Div_Kappa, ECSC, Bagging-RSM, RSM-Bagging and DCE-CC are similar to each other. It is noteworthy that the overall classification performance is, in general, positively related to diversity of base classifiers. For example, using DT as the classifier, the ECSC achieves the lowest misclassification rate, while the base classifiers have the largest diversity. The diversity measures of base classifiers in Bagging-RSM, RSM-Bagging, as well as their classification accuracy, are often higher than those in Bagging and RSM.

According to the above experimental study on classification performance and diversity of base classifiers, we can draw the following conclusions. First, base classifiers generated through supervised clustering and pairwise combination in ECSC are much more diverse than those in other methods, which is quite favorable for effective ensemble classification. Second, compared to Bagging,

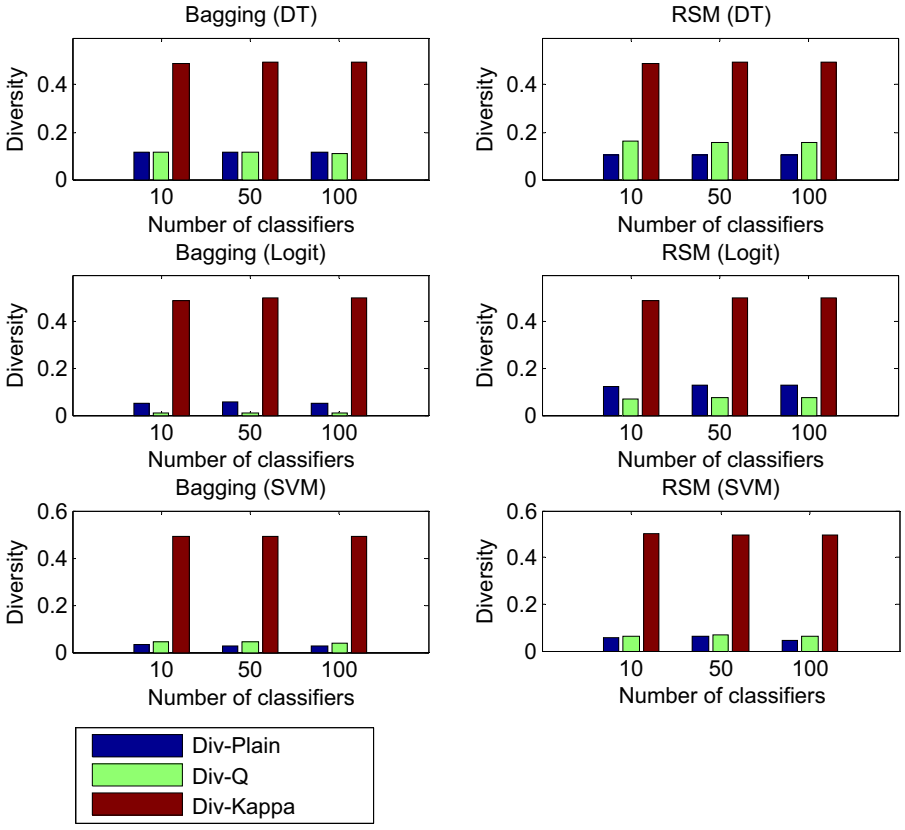

Fig. 6. Diversity measures of base classifiers in Bagging and RSM (Australian).

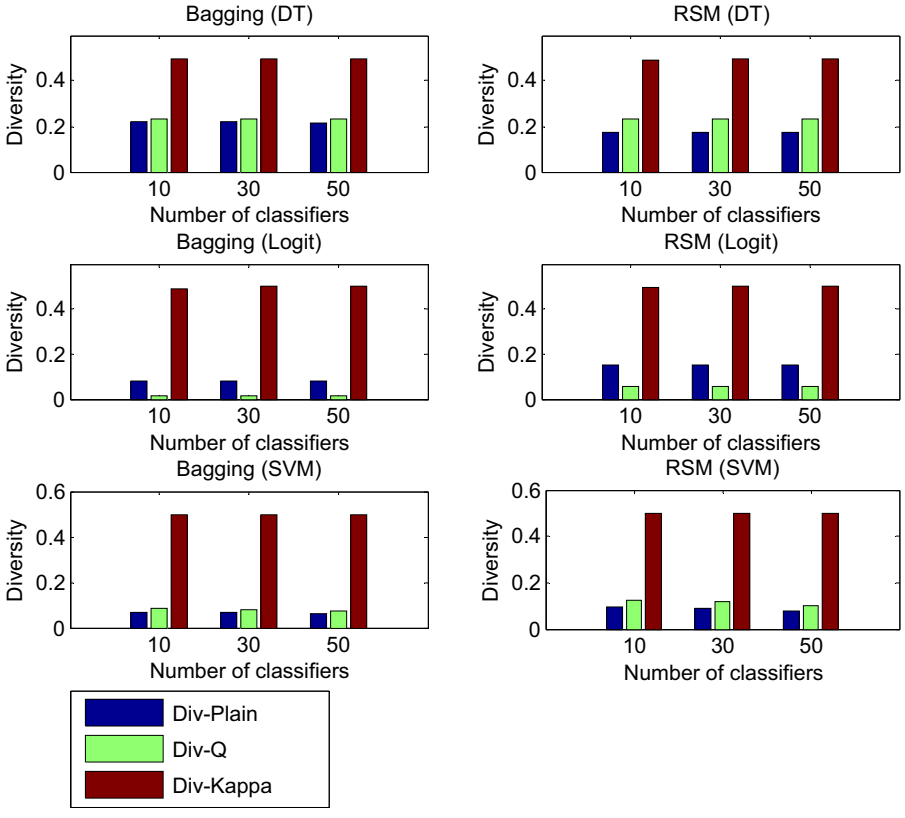

Fig. 7. Diversity measures of base classifiers in Bagging and RSM (German).

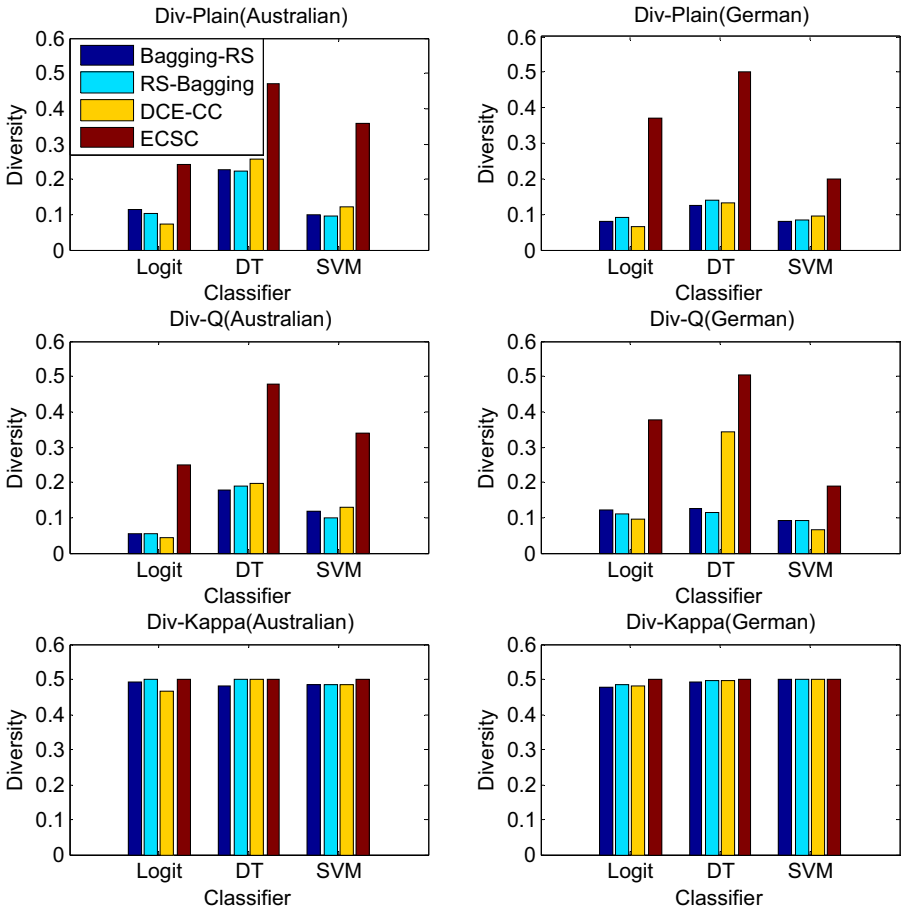

Fig. 8. Diversity measures of base classifiers in Bagging-RS, RS-Bagging, DCE-CC and ECSC.

RSM, and two state-of-the-art ensemble classification methodologies, i.e., dual ensemble strategy and DCE-CC, the proposed ECSC could achieve better classification performance. Therefore, it is effective and promising for credit scoring.

4.2.3. *Influence of diversity and integration mechanisms on the performance*

The performance of an ensemble depends mainly on the accuracy of the base classifiers, the diversity between base classifiers, and the integration mechanisms. To study the influence of diversity and integration mechanisms on ensemble performance, we first measure the correlation between the ensemble diversity and the gain of ensemble, which is defined as the difference between the ensemble accuracy and the average base classifier accuracy. Similar to Tsymbal [19], the Pearson's linear correlation coefficient r is used for the measurement. Since there are many combinations of ensemble methods (i.e., Bagging and RSM), integration mechanisms (i.e., MV, OLA, LCA, K-E, and K-U), and the number of base classifiers (i.e., 10, 30, 50, and 100), for the sake of simplicity, we only select the integration mechanism and the number of base classifiers corresponding to the lowest misclassification rate. The results are shown in Tables 7 and 8.

Tables 7 and 8 indicate that with respect to the diversity measure Div-Plain and Div-Q, there is a positive correlation between diversity and ensemble gain. Moreover, in all cases, the correlation in the ECSC is the strongest. This can be explained by the fact that in ECSC, the base classifiers are locally accurate since they are constructed in different local area, which leads to a low average base classifier accuracy over all samples. Consequently, the ensemble gain is high, and the correlation between diversity and ensemble

Table 7
Pearson's correlation coefficient r between the ensemble diversity and the gain of ensemble (Australian data set).

Base classifier	Ensemble method	Diversity measure		
		Div-Plain	Div-Q	Div-Kappa
Logit	Bagging	0.35	0.29	−0.12
	RSM	0.39	0.33	−0.05
	ECSC	0.52	0.57	−0.05
DT	Bagging	0.50	0.52	−0.05
	RSM	0.43	0.42	−0.08
	ECSC	0.59	0.56	0.02
SVM	Bagging	0.19	0.16	0.06
	RSM	0.18	0.21	−0.08
	ECSC	0.29	0.32	−0.02

Table 8
Pearson's correlation coefficient r between the ensemble diversity and the gain of ensemble (German data set).

Base classifier	Ensemble method	Diversity measure		
		Div-Plain	Div-Q	Div-Kappa
Logit	Bagging	0.24	0.29	0.05
	RSM	0.23	0.25	0.02
	ECSC	0.38	0.42	−0.01
DT	Bagging	0.44	0.42	−0.07
	RSM	0.53	0.59	−0.02
	ECSC	0.61	0.68	−0.11
SVM	Bagging	0.26	0.25	0.08
	RSM	0.25	0.31	0.08
	ECSC	0.41	0.47	−0.04

Table 9
Misclassification rates of majority voting versus weighted voting in ECSC.

Data set	Base classifier	Integration mechanism	Misclassification rate
Australian	Logit	MV	13.48 ± 2.50
		WV	11.95 ± 1.89
	DT	MV	17.25 ± 2.88
		WV	12.38 ± 2.06
	SVM	MV	14.40 ± 2.28
		WV	13.14 ± 1.85
German	Logit	MV	25.03 ± 2.43
		WV	22.24 ± 2.35
	DT	MV	29.08 ± 2.80
		WV	23.37 ± 1.85
	SVM	MV	31.32 ± 2.70
		WV	29.40 ± 2.64

gain is strong. However, it can also be seen that with respect to the diversity measure Div-Kappa, the correlation is near zero. It may be attributed to the fact that the diversity measures of different ensemble methods are similar with small fluctuation. Therefore, the correlation is quite weak.

To study the influence of integration mechanisms on ensemble performance in ECSC, we simply compare the results obtained by using simple majority voting (MV) versus those obtained by using weighted voting (WV) in Eq. (7) over 30 replications of 2-fold cross validation. The results are shown in Table 9.

Table 9 clearly indicates that the integration mechanism has critical impact on the ensemble performance. This has been demonstrated in Figs. 4 and 5 that represent the misclassification rates of Bagging and RSM using different integration mechanisms. But with respect to the ECSC, the impact is more significant. Recall that in ECSC, the base classifiers are locally constructed in different local area. Therefore, they are locally but not globally competent. Therefore, overall classification performance is seriously degraded by the integration mechanism of majority voting. In contrast, the weighted voting mechanism that better utilizes the local competence of base classifiers achieves significantly better performance.

5. A case study

5.1. Data description

The data used in the case study consists of 6000 records of consumers who get mortgage loans for their housing from a local bank in east China. Among them, 2713 consumers default more than one time, which is considered as bad credit, and other 3287 consumers repay on time and never default, which is considered as good credit. Each record of consumer contains more than 100 attributes (features) such as account number, birth date, job type, industry, address, zip code, loan commitment, etc. However, many features such as account number, telephone number, and code of bank branches, are not informative for credit risk. Therefore, we eliminate them in order to avoid the over-fitting problem caused by too many features. A detailed description of the remaining features used for classification is shown in Table 10.

To make the data set suitable for algorithms coping with numerical or integer variables, categorical features are coded as integers. For examples, the feature "gender" is coded as a binary number 0 or 1, and the feature "industry", which has 13 different values such as commerce, social service, insurance, government and education, is coded as integers 1–13. By this preprocessing procedure, all features are transformed into numerical or integer. In what follows, the normalization used in the above experimental study is applied to all the resulted features.

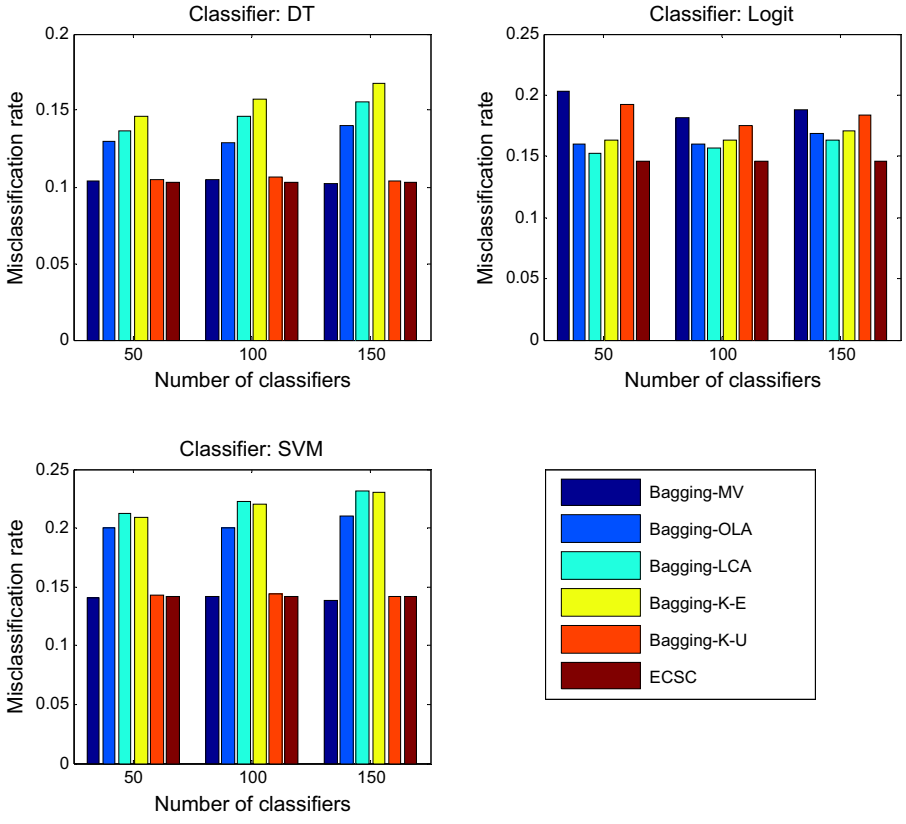

Fig. 9. Comparison of misclassification rates of Bagging and ECSC with different numbers of base classifiers over 30 runs.

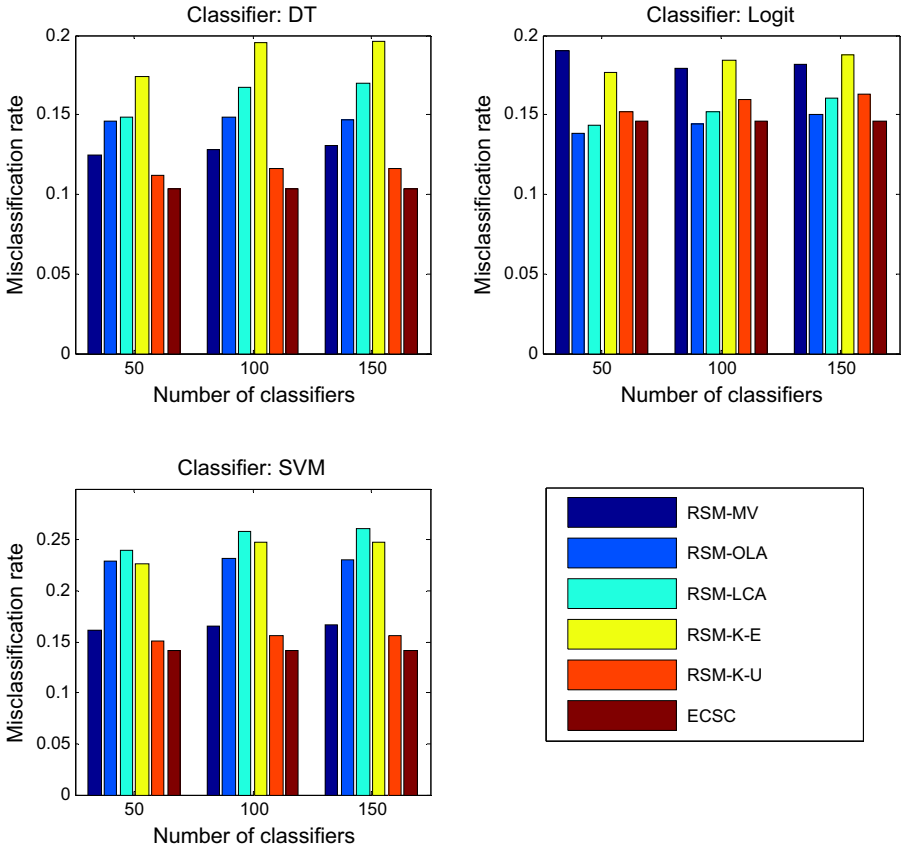

Fig. 10. Comparison of misclassification rates of RSM and ECSC with different numbers of base classifiers over 30 runs.

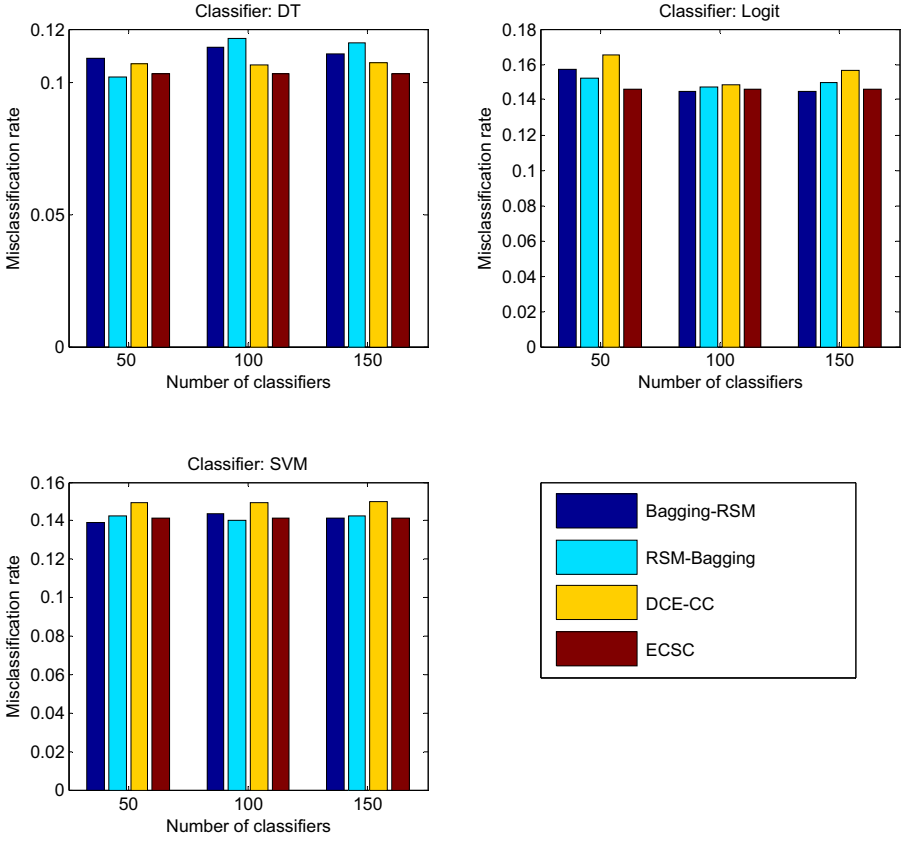

Fig. 11. Comparison of misclassification rates of Bagging-RSM, RSM-Bagging, DCE-CC and ECSC with different numbers of base classifiers over 30 runs.

Table 10
Description of features used for classification.

Feature ID	Feature description	Feature type
X1	Age	Numerical
X2	Gender	Categorical
X3	Educational level	Categorical
X4	Yearly income	Numerical
X5	Industry	Categorical
X6	Job type	Categorical
X7	Loan term	Numerical
X8	Loan commitment	Numerical
X9	Warrant	Categorical

5.2. Performance evaluation of different methods

In performance evaluation, cross validation is still used to evaluate the performance of different ensemble classification methods for credit scoring. The parameters of Bagging-RS, RS-Bagging and DCE-CC are also set the same as those in the above experimental study. The misclassification rates of Bagging, RSM, Bagging-RS, RS-Bagging, DCE-CC and ECSC with different kinds of classifiers over 30 runs of cross validation are shown in Figs. 9–11. Note that for Bagging, RSM, Bagging-RS, RS-Bagging and DCE-CC, the number of base classifiers is set as 50, 100 and 150 in order to investigate the impact of classifiers number on the classification performance. Remind that for ECSC, the number of classifiers is determined automatically by the supervised clustering procedure. However, for the sake of clear illustration, the misclassification rate of ECSC still appears in all figures.

Figs. 9–11 show that in comparison with other methods, the proposed ECSC achieves competitive or better classification performance. Although in a few cases, other methods outperform ECSC marginally, the proposed ECSC is promising and robust since it does not need to set the number of base classifiers, which has a non-negligible impact on the overall classification performance. For example, RSM-Bagging outperforms ECSC using DT as the classifier with ensemble size 50, but its performance decreases and become worse than that of ECSC when the ensemble size increases. Therefore, the proposed ECSC is an effective and promising approach for credit scoring.

6. Conclusion

In recent years, credit scoring has become one of the primary ways for financial institutions to improve their cash flow and reduce possible risks. Therefore, the accuracy of credit scoring is critically important to financial institutions' profitability. In this paper, we propose an ensemble classification approach based on supervised clustering for credit scoring. In the proposed approach, supervised clustering is first employed to partition the original data set into a number of subsets such that instances in one subset are from the same class. Subsets from different classes are then paired to form a number of training subsets, and in each subset, a base classifier is constructed. The outputs of different base classifiers are combined through weighted voting. We have illustrated that the proposed approach is able to generate diverse and locally accurate base classifiers through theoretical analysis and experimental study. Therefore, it is an effective and promising approach to credit scoring. For the further research, noises that often exist in real world credit data should be considered for better classification performance. Another important research direction is to extend the proposed approach to multi-class classification problems.

Acknowledgements

The authors are grateful to the editor and the anonymous reviewers for their constructive comments and suggestions, which have significantly improved the paper. The authors would like to thank Dr. Feng Ji for providing the data used in the case study. This research is supported by the National Natural Science Foundation of China (NSFC Grants No. 71001112 and 71471022) and Fundamental Research Funds for the Central Universities (Project No. 106112015CDJXY020001).

References

[1] D.J. Hand, W.E. Henley, Statistical classification methods in consumer credit scoring: a review, J. R. Stat. Soc. Ser. A (Stat. Soc.) 160 (1997) 523–541.
[2] G. Paleologo, A. Elisseeff, G. Antonini, Subagging for credit scoring models, Eur. J. Oper. Res. 201 (2010) 490–499.
[3] Y. Yang, Adaptive credit scoring with kernel learning methods, Eur. J. Oper. Res. 183 (2007) 1521–1536.
[4] G. Wang, J. Ma, L. Huang, K. Xu, Two credit scoring models based on dual strategy ensemble trees, Knowl. Based Syst. 26 (2012) 61–68.
[5] R.A. Eisenbeis, Problems in applying discriminant in credit scoring models, J. Bank. Financ. 2 (1978) 205–219.
[6] W.E. Henley, Statistical Aspects of Credit Scoring (Ph.D. Thesis), Open University, 1995.
[7] W.E. Henley, D.J. Hand, A *k*-NN classifier for assessing consumer credit risk, Statistician 65 (1996) 77–95.
[8] Y. Shi, Y. Peng, W. Xu, X. Tang, Data mining via multiple criteria linear programming: applications in credit card portfolio management, Int. J. Inf. Technol. Decis. Mak. 1 (2002) 131–151.
[9] Y. Peng, G. Kou, Y. Shi, Z.X. Chen, A multi-criteria convex quadratic programming model for credit data analysis, Decis. Support Syst. 44 (4) (2008) 1016–1030.
[10] T. Lee, C. Chiu, C. Lu, I. Chen, A two stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines, Expert Syst. Appl. 28 (2005) 743–752.
[11] N.S. Halvaiee, M.K. Akbari, A novel model for credit card fraud detection using artificial immune systems, Appl. Soft Comput. 24 (2011) 40–49.
[12] S.Y. Chang, T.Y. Yeh, An artificial immune classifier for credit scoring analysis, Appl. Soft Comput. 12 (2012) 611–618.
[13] C. Ong, J. Huang, G. Tzeng, Building credit scoring models using genetic programming, Expert Syst. Appl. 29 (2005) 41–47.
[14] Z. Huang, H.C. Chen, C.J. Hsu, W.H. Chen, S.S. Wu, Credit rating analysis with support vector machines and neural networks: a market comparative study, Decis. Support Syst. 37 (4) (2004) 543–558.
[15] S.T. Luo, B.W. Cheng, C.H. Hsieh, Prediction model building with clustering-launched classification and support vector machines in credit scoring, Expert Syst. Appl. 36 (2009) 7562–7566.
[16] S. Bhattacharyya, S. Jha, K. Tharakunnel, J.C. Westland, Data mining for credit card fraud: a comparative study, Decis. Support Syst. 50 (2011) 602–613.
[17] J.M. Tomczak, M. Zieba, Classification restricted Boltzmann machine for comprehensible credit scoring model, Expert Syst. Appl. 42 (2015) 1789–1796.
[18] T. Harris, Credit scoring using the clustered support vector machine, Expert Syst. Appl. 42 (2015) 741–750.
[19] A. Tsymbal, M. Pechenizkiy, P. Cunningham, Diversity in search strategies for ensemble feature selection, Inf. Fusion 6 (2005) 83–98.
[20] A.H.R. Ko, R. Sabourin, A.S. Britto Jr., From dynamic classifier selection to dynamic ensemble selection, Pattern Recognit. 41 (2008) 1718–1731.
[21] M.K. Lim, S.Y. Sohn, Cluster-based dynamic scoring model, Expert Syst. Appl. 32 (2007) 427–431.
[22] L. Kuncheva, A theoretical study on six classifier fusion strategies, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 281–286.
[23] L. Kuncheva, C. Whitaker, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Mach. Learn. 51 (2003) 181–207.
[24] G. Brown, J. Wyatt, R. Harris, X. Yao, Diversity creation methods: a survey and categorization, Inf. Fusion 6 (2005) 5–20.
[25] R. Pasti, L.N. de Castro, Bio-inspired and gradient-based algorithms to train MLPs: the influence of diversity, Inf. Sci. 179 (2009) 1441–1453.
[26] M.P. Sesmero, J.M. Alonso-Weber, G. Gutierrez, A. Ledezma, A. Sanchis, An ensemble approach of dual base learners for multi-class classification problem, Inf. Fusion 24 (2015) 122–136.
[27] L. Brieman, Bagging predictors, Mach. Learn. 24 (1996) 123–140.
[28] T.K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 832–844.
[29] R.E. Schapire, Y. Freund, P. Bartlett, W.S. Lee, Boosting the margin: a new explanation for the effectiveness of voting methods, Ann. Stat. 26 (1998) 1651–1686.
[30] M. Wozniak, M. Grana, E. Corchado, A survey of multiple classifier systems as hybrid systems, Inf. Fusion 16 (2014) 3–17.
[31] A.S. Britto Jr., R. Sabourin, L.E.S. Oliveira, Dynamic selection of classifiers – a comprehensive review, Pattern Recognit. 47 (2014) 3665–3680.
[32] R. Liu, B. Yuan, Multiple classifier combination by clustering and selection, Inf. Fusion 2 (2001) 163–168.

[33] D. Ruta, B. Gabrys, Classifier selection for majority voting, Inf. Fusion 6 (2005) 63–81.
[34] J. Xiao, C. He, X. Jiang, D. Liu, A dynamic classifier ensemble selection approach for noise data, Inf. Sci. 180 (2010) 3402–3421.
[35] T. Woloszynski, M. Kurzynski, A probabilistic model of classifier competence for dynamic ensemble selection, Pattern Recognit. 44 (2011) 2656–2668.
[36] L. Li, B. Zou, Q. Hu, X. Wu, D. Yu, Dynamic classifier ensemble using classification confidence, Neurocomputing 99 (2013) 581–591.
[37] Q. Dai, T. Zhang, N. Liu, A new reverse reduce-error ensemble pruning algorithm, Appl. Soft Comput. 28 (2015) 237–249.
[38] I. Mendialdua, A. Arruti, E. Jauregi, E. Lazkano, B. Sierra, Classifier subset selection to construct multi-classifiers by means of estimation of distribution algorithms, Neurocomputing 157 (2015) 46–60.
[39] P. Larranaga, J. Lozano, Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation, Kluwer Academic Press, 2001.
[40] S. Finlay, Multiple classifier architectures and their application to credit risk assessment, Eur. J. Oper. Res. 210 (2011) 368–378.
[41] J. Xiao, L. Xie, C. He, X. Jiang, Dynamic classifier ensemble model for customer classification with imbalanced class distribution, Expert Syst. Appl. 39 (2012) 3668–3675.
[42] J. Kruppa, A. Schwarz, G. Arminger, A. Ziegler, Consumer credit risk: individual probability estimates using machine learning, Expert Syst. Appl. 40 (2013) 5125–5131.
[43] C.F. Tsai, Y.F. Hsu, D.C. Yen, A comparative study of classifier ensembles for bankruptcy prediction, Appl. Soft Comput. 24 (2014) 977–984.
[44] M.N. Vrahatis, B. Boutsinas, P. Alevizos, G. Pavlides, The new k-windows algorithm for improving the k-mean clustering algorithm, J. Complex. 18 (2002) 375–391.
[45] J.C. Bezdek, Cluster validity with fuzzy sets, J. Cybern. 3 (1974) 58–73.
[46] B. Rezaee, A cluster validity index for fuzzy clustering, Fuzzy Sets Syst. 161 (2010) 3014–3025.
[47] Y. Zhang, W. Wang, X. Zhang, Y. Li, A cluster validity index for fuzzy clustering, Inf. Sci. 178 (2008) 1205–1218.
[48] K.L. Wu, M.S. Yang, A cluster validity index for fuzzy clustering, Pattern Recognit. Lett. 26 (9) (2005) 1275–1291.
[49] R.M. Ramze, B.P.F. Lelieveldt, J.H.C. Reiber, A new cluster validity indexes for the fuzzy c-mean, Pattern Recognit. Lett. 19 (1998) 237–246.
[50] S. Vucetic, Z. Obradovic, Discovering homogeneous regions in spatial data through competition, in: Proceedings of the 17th International Conference on Machine Learning, Stanford, CA, 2000, pp. 1091–1098.
[51] D.W. Hosmer, L. Stanley, Applied Logistic Regression, Wiley, New York, 2000.
[52] X. Xu, Y. Wang, Financial failure prediction using efficiency as a predictor, Expert Syst. Appl. 36 (2009) 366–373.
[53] J. Han, M. Kamber, Data Mining: Concepts and Techniques, 2nd ed., Morgan Kaufmann Publishers, San Francisco, CA, 2006.
[54] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 1995.
[55] Z. Hua, Y. Wang, X. Xu, B. Zhang, L. Liang, Predicting corporate financial distress based on integration of support vector machine and logistic regression, Expert Syst. Appl. 33 (2007) 434–440.
[56] K. Coussement, W. Buckinx, A probability-mapping algorithm for calibrating the posterior probabilities: a direct marketing application, Eur. J. Oper. Res. 214 (2011) 732–738.
[57] Z.H. Zhou, J. Wu, W. Tang, Ensembling neural networks: many could be better than all, Artif. Intell. 137 (1–2) (2002) 239–263.
[58] T.G. Dietterich, Approximate statistical tests for comparing supervised classification learning algorithms, Neural Comput. 10 (7) (1998) 1895–1923.
[59] E. Alpaydin, Combined 5×2 cv F test for comparing supervised classification learning algorithms, Neural Comput. 11 (8) (1999) 1885–1892.
[60] E. Alpaydin, Introduction to Machine Learning, 2nd ed., MIT Press, 2010.