Expert Systems With Applications 128 (2019) 301–315

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Integration of unsupervised and supervised machine learning algorithms for credit risk assessment

Wang Bao[a], Ning Lianju[a,*], Kong Yue[b]

[a] School of Economics and Management, Beijing University of Posts and Telecommunications, P.O. Box 164 10, Xitucheng Road, Haidian District, Beijing 100876, PR China

[b] State Key Laboratory of Chemical Resource Engineering, Department of Pharmaceutical Engineering, P.O. Box 53, Beijing University of Chemical Technology, 15 Beisanhuan East Road, Beijing 100029, PR China

ARTICLE INFO

Article history:
Received 18 November 2018
Revised 22 February 2019
Accepted 24 February 2019
Available online 19 March 2019

Keywords:
Credit scoring
Ensemble model
Unsupervised machine learning
Supervised machine learning
Kohonen's self-organizing maps (SOM)

ABSTRACT

For the sake of credit risk assessment, credit scoring has become a critical tool to discriminate "bad" applicants from "good" applicants for financial institutions. Accordingly, a wide range of supervised machine learning algorithms have been successfully applied to credit scoring; however, integration of unsupervised learning with supervised learning in this field has drawn little consideration. In this work, we propose a combination strategy of integrating unsupervised learning with supervised learning for credit risk assessment. The difference between our work and other previous work on unsupervised integration is that we apply unsupervised learning techniques at two different stages: the consensus stage and dataset clustering stage. Comparisons of model performance are performed based on three credit datasets in four groups: individual models, individual models + consensus model, clustering + individual models, clustering + individual models + consensus model. As a result, integration at either the consensus stage or dataset clustering stage is effective on improving the performance of credit scoring models. Moreover, the combination of the two stages achieves the best performance, thereby confirming the superiority of the proposed integration of unsupervised and supervised machine learning algorithms, which boost our confidence that this strategy can be extended to many other credit datasets from financial institutions.

1. Introduction

1.1. Background

Credit risk has been considered a crucial factor when commercial banks and financial institutions grant loans to potential customers or borrowers. Therefore, reliable evaluation models for credit risks plays an important role in loss control and revenue maximization (Chen, Ribeiro, & Chen, 2016; Luo, Kong, & Nie, 2016). Probability of default describes the likelihood of a default over a particular time horizon and is a major parameter used in a variety of credit risk evaluation systems, especially under the regulatory framework of Basel III. Since credit scores imply a certain probability of default, credit scoring has been emerged as an aid decision tool to identify "good" applicants and "bad" applicants when financial institutions issue loans (Thomas, Oliver, & Hand, 2005). Consequently, credit scoring models have been extensively investigated and developed for the credit approval assessment of new applicants. Traditionally, some statistical algorithms, such as linear discriminate analysis (LDA) (Altman, 1968) and logistic regression (LR) (Wiginton, 1980) were used to tackle this problem. These statistical methods are widely used due to their simple interpretability and easy implementation; however, their relatively poor predictive performance limit their use especially on large datasets with a vast number of feature dimensions. In order to be useful and efficient, credit scoring models must pursue a good balance between the classification performance and interpretability (Florez-lopez & Ramon-jeronimo, 2015). Nowadays, the classification performance has become more and more important for credit scoring, because even a small fraction of a percentage of improvement means a considerable amount of profits for financial institutions (Abellán & Castellano, 2017; Ala & Abbod, 2016b).

With the development of machine learning (ML) algorithms and accumulation of a great amount of multi-dimensional customer data, developing credit scoring models with ML methods has become a hotspot (Bequé & Lessmann, 2017; Dahiya, 2017; Xia, Liu, Da, & Xie, 2018). The ML methods can be roughly divided into two categories, namely supervised ML and unsupervised ML. The fundamental difference between these two learning algorithms is whether the examples given to the learning algorithm

* Corresponding author.
E-mail addresses: bao.wang@outlook.com (W. Bao), ninglj@bupt.edu.cn (N. Lianju), kongyue52@126.com (K. Yue).

https://doi.org/10.1016/j.eswa.2019.02.033
0957-4174/

are labeled or not. The supervised ML, applied to the labeled examples, has a wide range of available algorithms, such as support vector machines (SVM) (Sun & Li, 2015; Xu, Zhou, & Wang, 2009), decision trees (DT) (Zhou, Si, & Fujita, 2017), random forest (RF) (Malekipirbazari & Aksakalli, 2015), artificial neural network (ANN) (West, 2000), each with its strengths and weaknesses. The unsupervised ML, applied to the unlabeled examples, includes *k*-means (Kodinariya & Makwana, 2013), hierarchical clustering (Ruppert, 2004), and DBSCAN (Ester, Kriegel, Sander, & Xu, 1996), Kohonen's self-organizing maps (SOM) (Yan, Nie, Wang, & Wang, 2013) and isolation forest (Liu, Ting, & Zhou, 2008).

Both supervised ML and unsupervised ML have been extensively applied in credit risk assessment. The supervised ML algorithms are used in credit scoring models to find the relationship between the customer features and credit default risk and then predict the default classification usually in a binary format. In a large body of literature, the implementation of supervised ML algorithms in credit scoring models has shown good predictive accuracy (Ben-david & Frank, 2009; Sohn, Kim, & Yoon, 2016; Twala, 2010). Unsupervised ML algorithms, in most cases particularly referring to clustering algorithms, are used as an important data mining technique to cluster examples into groups of similar objects instead of giving predictions directly. Therefore, these unsupervised ML algorithms are often used as complimentary tools to supervised ones. For example, several researches have focused on clustering-launched SVM models using unsupervised ML algorithms like divisive hierarchical *k*-means (DHK) and SOM (Luo, Cheng, & Hsieh, 2009; Yu, Yue, Wang, & Lai, 2010). On the other hand, there are also some unsupervised ML algorithms like SOM that can be used to give predictions, but comparatively few applications have been reported in the field of credit risk assessment (AghaeiRad, Chen, & Ribeiro, 2017; Huysmans, Baesens, Vanthienen, & Van Gestel, 2006)

More recently, research has focused on ensemble strategy of integrating different ML models for credit scoring, and one of the mainstream ensemble strategies is to make consensus classification decision based on predictive outcomes of individual ML models. There are different approaches to perform ensemble strategy in terms of using different base learners (single classifiers or models) and different consensus techniques. There has been an explosion of papers adopting ensemble strategy which has shown promising results that can help improve the classification performance of credit scoring models (Ala, 2015; Ala & Abbod, 2016a; Cleofas-Sánchez, García, Marqués, & Sánchez, 2016; Lessmann, Baesens, Seow, & Thomas, 2015; Wang, Ma, Huang, & Xu, 2012; Xia et al., 2018). In terms of consensus techniques, the majority voting (MV) is the simplest and most commonly applied in literature (Zhou, Tam, & Fujita, 2016). Stacking (Lessmann et al., 2015; Wang, Hao, Ma, & Jiang, 2011), as a more complicated and superior consensus strategy than MV method, uses a high-level learner to train the outcomes of base learners, mostly with supervised algorithms used as the high-level learner. However, the over reliance on supervised algorithms can lead to overfitting problems. Based on these observations, the idea behind this work is to adopt an efficient unsupervised algorithm as the high-level and avoid the overfitting risk at the same time.

A problem for credit scoring models which needs to be emphasized is the unavailability of real-world credit data, since the customers' credit data is confidential in most of the financial institutions and the researchers could not get access to these data. World credit datasets published by the University of California Irvine (UCI) (Australian dataset, 1987; German dataset, 1994) have been widely used to launch credit scoring research over the years. However, factors related to customer credit have been significantly changed since the customer behaviors changed greatly and rapidly. As a result, researches based on up-to-date real-world dataset are important and urgent. Fortunately, we are able to use one private dataset from a Chinese P2P enterprise. Nonetheless, there are still many difficulties and challenges when using this source of data because of data incompleteness and inaccuracies, which is a common and tough issue faced by almost all the financial institutions (Florez-Lopez, 2010; Twala, 2013). Among these difficulties, the data incompleteness is a major and the most frequently encountered problem. Incompleteness means some certain data fields are blank—this is because some data may not be collectable from all customers, the data collecting system has been altered or customers neglect to submit some optional items when customers filling out forms. However, this "missing state" is worth mining and can be taken advantages of to do dataset segmentation. It is commonly expected that segmentation will improve the model performance (Bijak & Thomas, 2012). In this work, we propose an unsupervised clustering method based on the "missing state" as input to do simultaneous segmentation followed by supervised learning models, to explore whether this strategy can help improve performance of the credit scoring models. The intention of the proposed strategy for handling data missing need to be stated clear: we aimed to make use of the "missing state" other than to fill or handle the missing data with some data mining techniques, since we believe there is some certain pattern behind the "missing state" and this pattern can be a good indicator for us to do segmentation.

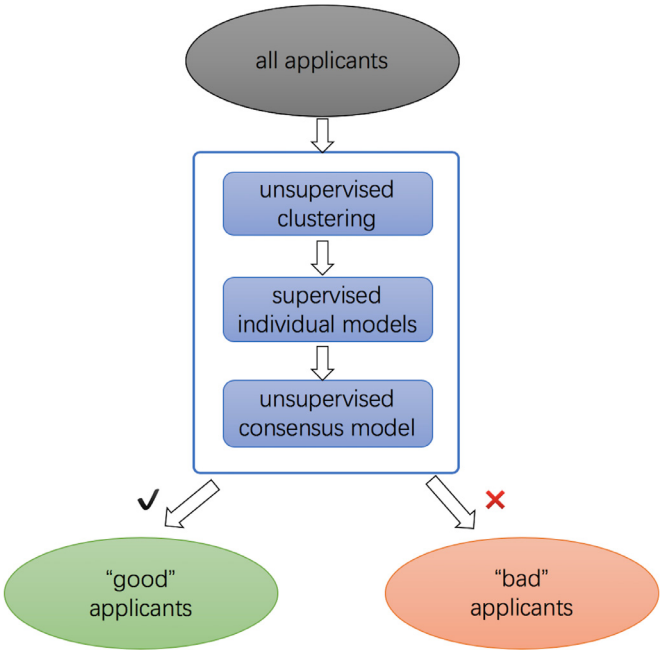

Fig. 1. Schematic diagram of main idea in this work.

1.2. Research motivation

This work focuses on integrating unsupervised ML techniques with supervised ML classifiers at different stages, aiming to improve the performance of credit scoring models. The main idea of this work is shown in Fig. 1, which shows that following the unsupervised clustering, the supervised predictive models are followed by the unsupervised consensus model. In order to test the validity of the integration of unsupervised and supervised ML techniques, this work compares different combinations of integration as shown in Fig. 2: individual models, individual models + consensus model, clustering + individual models, clustering + individual models + consensus model.

In terms of individual models, there are a wide range of supervised ML algorithms available, each with its own strengths and

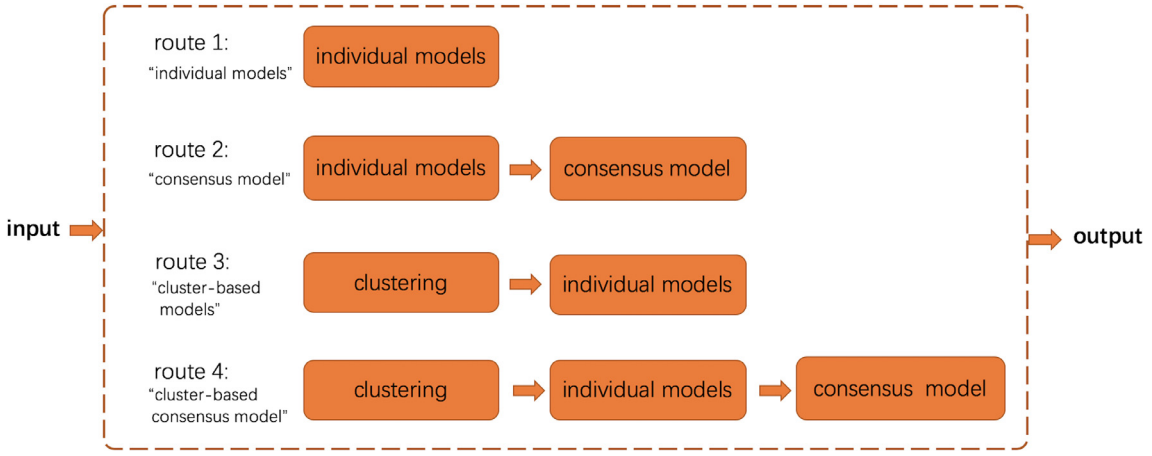

Fig. 2. Modeling system of integration of unsupervised and supervised machine learning algorithms. Four routes are designed in this work with their names referred in following parts.

weaknesses. As the "No Free Lunch" theorem indicates, there is no single ML algorithm can perform best on all practical learning problems. Therefore, we adopt a broad set of diverse, accurate and representative ML algorithms, namely logistic regression (LR), support vector machines (SVM), decision trees (DT), artificial neural network (ANN), *k*-nearest neighbor (kNN), random forest (RF), and gradient boosting decision trees (GBDT), to build individual models. These individual models are the base of this research on which a consensus model will be built using unsupervised methods. The predictive performance of test sets is evaluated against seven parameters, namely, Matthews correlation coefficient (MCC) (POWERS, 2011), area under the curve (AUC), overall accuracy, precision, the recall, Type I error and Type II error.

Moreover, a competent ensemble algorithm is a key requirement for a successful ensemble strategy. As discussed above in the background part, unsupervised ML algorithms not only can be used to construct consensus models, but also can avoid the overfitting risk. Given that SOM has shown a powerful ability to give predictions and find the intrinsic patterns behind the data, it has been widely adopted in many researches (Fernandes, Scotti, Ferreira, & Emerenciano, 2008; Kong & Yan, 2017); nonetheless, in the field of credit scoring, few applications of ensembles integrated with SOM have been reported in the field of credit risk assessment. AghaeiRad et al. (2017) reported an ensemble technique that use feedforward neural network (FNN) to build credit scoring models with dataset input clustered by SOM, with the conclusion that SOM clustering can contribute to boosting the performance of FNN. In this case, however, SOM was used as a clustering method before modeling other than as a consensus method. Therefore, the idea behind this paper is to perform an ensemble strategy that first builds a series of supervised models and then using SOM to make consensus classification decision, to see whether the performance of credit scoring models can be improved.

The rest of this paper is organized as follows. Section 2 provides an introduction for the datasets, data cleaning and feature selection method, machine learning methods, model settings and performance measuring parameters adopted in this work. Section 3 presents the experimental results in details. Finally, based on the analysis and results of these experiments, Section 4 concludes this work and discusses the future research directions.

2. Methodology

In this section, the methods related to this work are presented in the following five aspects: credit datasets, data cleaning and feature selection, machine learning algorithms, model settings and evaluation measures of model performance. This whole workflow which is used in route 4 (in Fig. 2) is presented in Fig. 3 showing the experimental process.

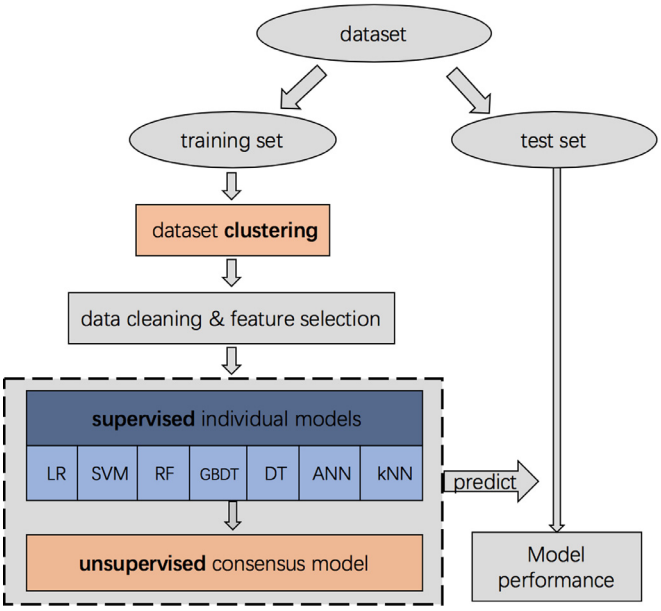

Fig. 3. The experimental process (route 4) in this work.

2.1. Credit datasets

Three credit datasets either from one Chinese P2P enterprise or traditional UCI machine learning repository are adopted in this work. The detailed information profiling the datasets in terms of number of samples, default ratio and feature dimensions are presented in Table 1. Given that the Chinese P2P credit dataset is up-to-date and contains a wider range of dimensions describing the characteristics and behaviors of customers than the other two datasets, although it is confidential, we emphasized and focused our research on this dataset, while using the UCI datasets as benchmarks for comparisons.

2.1.1. Chinese P2P credit dataset

The Chinese P2P credit dataset was provided by one of Chinese P2P lending enterprises. As described in Table 1, this dataset

Table 1
Descriptions of credit datasets in this work.

Credit dataset	Total	Non-defaults/defaults	Default ratio	Feature dimension
Chinese P2P	23,435	19,546/3889	16.6%	81
German	1000	700/300	30%	20
Australian	690	307/383	55.5%	14

comprises 23,435 samples, including 3889 defaults (bad applicants) and 19,546 non-defaults (good applicants). For this dataset, we adopted the 90-days delay or more after the loan is issued as the indicator of default. The data was sampled from the original database with date ranging from January 2014 to September 2017. The observation date is 1st September 2018, that is to say, we leave at least one year for applicants to present their credit status. The loan products launched by this Chinese P2P lending enterprise range from 10,000 RMB to 200,000 RMB, with an average of 50,000 RMB. Within the dataset, 81 dimensions (customer features) are included; however, they cannot be listed since this is proprietary and confidential information. Some examples of listed features are age, gender, marital status, education background, insurance status, monthly mortgage and credit card limit. Many dimensions are not provided by the applicants, leaving the dataset with a lot of empty points, which will be explored and analyzed in the *Experimental Results and Discussions* section of this paper.

2.1.2. German and Australian credit datasets

The German and Australian credit datasets come from the UCI Machine Learning Repository as mentioned in the Introduction section. For the German dataset, there are in total 1000 samples composed of 700 good applicants and 300 bad applicants. Each sample is described in twenty dimensions (or features), with three continuous features and seventeen categorical features. These features are related to personal information, age, employment status, job, housing, credit history, account balances, loan amount, loan purpose. While for the Australian dataset, there are 690 samples composed of 307 good applicants and 383 bad applicants and the dataset is described in 14 dimensions (six continuous and eight categorical).

2.2. Data cleaning and feature selection

In the process of data cleaning, we focused on two issues: the empty points and outliers. In terms of handling the empty points, four kinds of methods can be used: case deletion, missing data imputation, model-based procedures and machine learning methods (García-Laencina, Sancho-Gómez, & Figueiras-Vidal, 2010). In this work, the empty points were processed mostly according to our experience. For the features not filled out by 95% of applicants or above, we removed this feature; for the remaining features, we added a new feature to describe its empty status with empty as "1" and non-empty as "0", and meanwhile filled in the empty points with the mean values of these features. In terms of the outliers, it has been demonstrated that the use of filters for outliers can help improve the model performance (García, Marqués, & Sánchez, 2012). In this work, we manually checked the rationality of abnormal points, then we kept these outliers if they are rational, otherwise, we replace the outliers with the upper or lower values of box plots. Additionally, normalization was performed to scale the feature values so that they can fall into specified range, typically from 0 to 1.

Feature selection can help improve the prediction performance of the classifiers and provide faster and more cost-effective classifiers. The process of feature selection is often combined with the subset selection and the methods of selecting subsets of features can be roughly divided into three categories: wrappers, filters and embedded methods (Guyon, & Elisseeff, 2011; Saeys, Inza, & Larrañaga, 2007). In this work, we adopted one of the embedded methods—the tree-based feature selection method, i.e. selecting features based on random forest models (Breiman, 1999). Random forest can be used to compute feature importance, this in turn can be used to select the most important features and remove the irrelevant features.

2.3. Supervised machine learning methods

2.3.1. Logistic regression (LR)

Logistic regression is a simply parametric statistical approach and has been considered as the industry standard in the field of credit scoring (Lessmann et al., 2015). It is used to solve binary classification problems (default and non-default in this work) and regression problems. The target of LR model is to get the logarithm of the ratio of two probability outcomes of interest, and the formula is as the following equation (Eq. (1)),

$$log[p(1-p)] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n \quad (1)$$

where log *[p (1- p)]* is the dependent variable that is the logarithm of the ratio of two probability outcomes of interest, with p is the default probability. β_i as the intercept term is the coefficient related to the independent variables X_i ($i = 1, ..., n$). The objective of LR model is to predict the conditional probability of a certain sample belonging to a certain class.

2.3.2. Decision trees (DT)

DT is a non-parametric classification approach and has been commonly used in the field of credit scoring (Lee, Chiu, Chou, & Lu, 2006). The predictive results of DT can be represented as a directed, acyclic graph in a form of a tree, thus the results of model can be easily understood by humans. DT model classifies a sample input by sorting it through the tree then allocate it to the most appropriate leaf node (a class label). In a DT graph, each node represents some feature of the sample and each branch represents a possible value of that feature. In this work, we used C4.5 DT (Quinlan, 1993) which is an extension of Quinlan's earlier Iterative Dichotomiser 3 (ID3) algorithm. ID3 and C4.5 both use information entropy to choose features as decision nodes, while C4.5 made some improvements to ID3 in terms of handling missing data points and pruning trees to avoid over-fitting problems.

2.3.3. Random forest (RF)

RF is considered an advanced technique of DTs, as proposed by Breiman (1999), with the idea behind RF being combining bagging and random subspace feature selection to merge individual DTs. RF model uses randomness in two stages: first, randomly select subsets from the original dataset; secondly, randomly select subsets of features derived from the original full feature dimensions. In this way, the correlation between the DTs in the forest are reduced. And the final decision is made based on voting procedure, where the input sample will be labeled as the class with the majority of votes.

2.3.4. Gradient boosting decision trees (GBDT)

Gradient boosting algorithm (Friedman, 2001) has gained notice in machine learning in recent years and is popular for its use

to solve classification and regression problems. The term "gradient boosting" comes from the idea of "boosting" which is to ensemble weak base learners with high bias and low variance, typically decision trees, in order to generate a more reliable and accurate model. Gradient boosting is an extension of boosting where the boosting-based error minimization strategy is used to additively generate models. The modeling process is to add decision trees at a time, then the next tree is added and trained to reduce the loss by moving in the right direction. The model keeps adding trees until the number of trees reaches a fixed number or the loss reaches an acceptable level or no longer improves.

2.3.5. Support vector machines (SVM)

SVM is another popular supervised machine learning algorithm, first proposed by Cortes and Vapnik (1995), and has been extensively applied in field of credit scoring owing to its powerful predictive capabilities. It stands out in comparison with other algorithms because of its superior solutions of solving the problem of sparsity. Basically, its main idea is to project the input data into a high-dimensional feature space and then find a hyper plane supported by the support vectors to separate the two classes with a maximal margin. Based on the features of the support vectors, the label of the new input sample can be predicted. Many functions (or called kernels) in SVM can be chosen to map the input data into the high-dimensional feature space, namely, linear, polynomial, radial basis function (RBF) and sigmoid (Zhou, Lai, & Yu, 2010).

2.3.6. k-Nearest Neighbors (kNN)

KNN is one of the most venerable algorithms in statistical pattern recognition (Cover & Hart, 1967). It has been extensively used in constructing credit scoring models (Henley & Hand, 1996; Islam, Wu, Ahmadi, & Sid-Ahmed, 2007). The main idea behind the kNN model is that it predicts the labels of the new input samples according to the nearest set (or *k*-nearest neighbors) of previously labeled samples. Euclidean distance is commonly used in kNN models to measure the distance between the new sample and the previous training samples. This type of model can be easily handled since there is only one parameter – the size of neighborhood "*k*" in kNN algorithm. Holmes and Adams (2002) have proposed a probabilistic strategy to establish a formal framework to set this parameter.

2.3.7. Artificial neural network (ANN)

ANN (Bishop, 1997), usually as a nonparametric approach, has been used for a wide range of classification and regression problems (Marqués, García, & Sánchez, 2012; West, 2000). It is inspired by the design of the biological neuron and it mimics the human brain functions in terms of capturing complex relationships between the inputs and outputs. There are many types of ANNs and one of the most common architectures is the multi-layer perceptron (MLP), which consists of one input layer, one or more hidden layers and one output layer. In terms of credit scoring, the ANN model starts by passing the features of each applicant to the input layer, and then processes these features through the hidden layers, finally reaching the output layer where the final answer will be given based on the weights. These weights are assigned to each feature based on its relative importance. Then an activation function, i.e. sigmoid, tangent-sigmoid, collects all the weighted feature to produce outputs (Malhotra & Malhotra, 2003). The process of adjusting weights is repeated in many loops, aiming to minimizing the errors between the predicted class and true class.

2.4. Unsupervised machine learning methods

2.4.1. k-means

k-means is a simple and efficient clustering (or unsupervised learning) method (Asgharbeygi & Maleki, 2008). The basic steps for *k*-means clustering are as follows: a) randomly select *k* cluster centers; b) assign each data points to its nearest cluster center; c) replace the original center with the position center in each cluster; d) relocate each data points to a new cluster which it is nearest to; e) repeat the previous steps c and d until no data point changes position or some convergence criterion is met. There is only one main parameter in *k*-means model, that is, the number of clusters *k*. In this work, k-means was used to clustering samples according to their presence conditions (missing or not) to split the dataset into several subsets based on which supervised machine learning models were built.

2.4.2. Kohonen's self-organizing maps (SOM)

SOM is an unsupervised neural network introduced by introduced by Kohonen (1998). The idea of SOM is to project a nonlinear data vector from high-dimensional space into a two-dimensional space, which makes the patterns graphically visualized and easily recognizable. In an SOM map, the neurons are arranged in a two-dimensional array where a winner neuron will be found for each data vector. In the process of training maps, the winner neuron and its neighbors are adjusted according to topology distance, consequently the topological similarity is preserved between the neighboring neurons. That is to say, if two samples are projected to adjacent neurons, these two samples are similar in terms of input feature descriptions. In this work, we labeled each neuron in an SOM map based on the training set according to the majority voting principle and then predicted the samples in the test set with the pre-defined labels. In cases where the samples from the test set were projected to the neurons without any sample from the training set being projected to, we then turned to look at their surrounding neurons and labeled these undefined neurons according to the majority voting principle based on their surrounding neurons.

2.5. Model settings

In this work, all the supervised models together with *k*-means were completed by Scikit-learn (Pedregosa et al., 2011). Grid search was used to optimize the combination of hyper-parameters within 5-cross-validation. Since grid search uses an exhaustive search of pre-defined hyper-parameter space, we provide the search space for these algorithms here: for kNN models, the k_num was searched in range of 3 to 10; for DT models, tree_depth was searched in range of 5 to 15 and min_samples_spit in range of 5 to 20; for RF models, tree_num was searched in range of 100 to 500, min_samples_leaf in range of 3 to 10 and max_depth in range of 5 to 25; for SVM models, log2c and log2g were searched in range of −10 to 10 with the RBF kernel adopted; for GBDT models, tree_num was searched in range of 100 to 500, max_depth in range of 5 to 25 and learning_rate in range of (0.001, 0.01, 0.1, 1); for ANN models, the hidden_nodes was searched in range of 10 to 50 and the log(alpha) in range of (0.001, 0.01, 0.1, 1); for k-means method, n_clusters was set to 8 for the Chinese P2P credit dataset and to 5 for the German and Australian credit datasets. Additionally, other parameters not mentioned were set as default.

Besides this, SONNIA (2016) was used to generate the SOMs in this work. Parameters were set as default.

2.6. Evaluation measures of model performance

The whole dataset was split into training set and test with the ratio of 2:1 and 5-fold cross validation (5-CV) was performed on the training set to validate the credit scoring models. A series of performance measures were employed, including accuracy (ACC), Matthews correlation coefficient (MCC) (Powers, 2011), the area

Table 2
Confusion matrix used in this study.

True label	Predicted label	
	Default[a]	Non-default[b]
Default[a]	True positive (TP)	False negative (FN)
Non-default[b]	False positive (FP)	True negative (TN)

[a] Also referred as good applicants.
[b] Also referred as bad applicants.

under the curve (AUC), recall, precision, type I error and type II error.

ACC measures the overall predictive effectiveness of models; however, it is not a reliable parameter as it yields misleading results if the dataset is not balanced, which is often seen in real-world credit datasets. Therefore, MCC as a more reliable measuring parameter, describing the confusion matrix more comprehensively than ACC, is often used to measure the model performance on an unbalanced dataset (Kong & Yan, 2017). MCC ranges from −1 to 1, and commonly, an MCC of 0.6 or above means the model has a good classification capability. AUC is a widely used parameter derived from the receiver operating characteristic (ROC) curve (Fawcett, 2006). AUC represents the probability for a model to rank a good applicant randomly rather than rank a bad applicant randomly. It ranges from 0 to 1 and the higher the AUC is, the better the model performance is. Recall is the fraction of correctly predicted defaults over the total amount of defaults, while precision is the fraction of correctly predicted defaults over the predicted defaults. Type I error indicates the proportion of non-defaults (good applicants) who are wrongly predicted to be defaults (bad applicants) among all the non-defaults; while type II error indicates the proportion of defaults who are wrongly predicted to be non-defaults among all the defaults.

The parameters mentioned above are calculated based on a confusion matrix shown in Table 2. True positive (TP) refers the number of defaults that are correctly predicted as defaults; false positive (FP) refers the number of non-defaults that are mistakenly predicted as defaults; true negative (TN) refers the number of non-defaults that are correctly predicted as non-default; false negative (FN) refers the number of defaults that are mistakenly predicted as non-defaults. The parameters used in this work are calculated with the following equations (Eqs. (2)–(7)).

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Type\ I\ error = \frac{FP}{FP + TN} \tag{6}$$

$$Type\ II\ error = \frac{FN}{TP + FN} \tag{7}$$

3. Experimental results and discussions

In this section, an unsupervised consensus approach based on seven supervised ML methods is proposed, along with using a clustering-based algorithm producing datasets for models to be trained on. This work mainly focused on testing the proposed approach on the Chinese P2P credit dataset, and at the same time we performed the same strategy on two benchmarking datasets – the German and Australian credit datasets to make comparisons. Therefore, the structure of this section is that the results obtained on the Chinese P2P dataset are stated in Sections 3.1–3.5, while the comparison results obtained on the two benchmarking datasets are given in Sections 3.6 and 3.7. In the last part ROC curve analysis is presented. Details of Sections 3.1–3.5 are extended as follows:

Firstly, we explored the data space of the Chinese P2P dataset. Then model performance analysis of four routes (see Fig. 2) are presented: see Section 3.2 for the performance of seven supervised individual models; see Section 3.3 for the performance of SOM consensus model based on the previously-built individual models; see Section 3.4 for the performance of cluster-based individual models; see Section 3.5 for the performance of SOM consensus models based on the previously-built cluster-based individual models.

3.1. Dataset space

The data space in terms of data presence condition was explored before building classification models. In the experimental process, for each sample, the empty points were labeled as “0” and non-empty points as “1”. After removing constant values, we used hierarchical k-means algorithm (Lamrous & Taïleb, 2007) to cluster the input samples labeled either “0” or “1”. As a result, a hierarchical clustering map was produced (see Fig. 4). From Fig. 4, we can see that there are some obvious patterns in data distributions for many practical reasons. Although obvious pattern cannot be seen from the default distribution of each clusters, the following experiments based on clusters were used to further investigate the default patterns.

3.2. Performance of supervised individual models (route 1)

Individual models were built with seven base learners (ANN, DT, GBDT, kNN, LR, RF and SVM), see route 1 in Fig. 2. Table 3 summarizes the model results based on the test set of the Chinese P2P credit dataset in terms of MCC, AUC, ACC, recall, precision, Type I error and Type II error. These results are also presented graphically in Fig. 5. Since these evaluation parameters describe the model performance from different aspects and there is no one single classifier which is shown to be preferable to the others on all parameters, we emphasis using the MCC to evaluate the models as it considers the confusion matrix more comprehensively than the other parameters and it is more efficient to reflect the model performance of un-balanced dataset. According to MCC values, we can conclude that the GBDT model performs the best among all the individual models with the MCC of 0.663. And the GBDT model also has the best AUC (0.949) and ACC (0.915). ANN, SVM and RF models also perform well, slightly behind the GBDT model, with the MCCs of 0.661, 0.660, and 0.656, respectively. Note that the kNN and the DT models have the best Type I error (0.012) and Type II error (0.157), respectively, though they do not perform well in terms of MCC values.

3.3. Performance of SOM consensus model based on individual models (route 2)

Based on the probabilities given by the seven individual models, we built the SOM consensus model (see route 2 in Fig. 2). The MCC, AUC, ACC, recall, precision, Type I error and Type II error values of this consensus model are shown in Table 3. As we can see from these results, the MCC is improved to 0.671 compared to individual models, which means the consensus model outperforms the

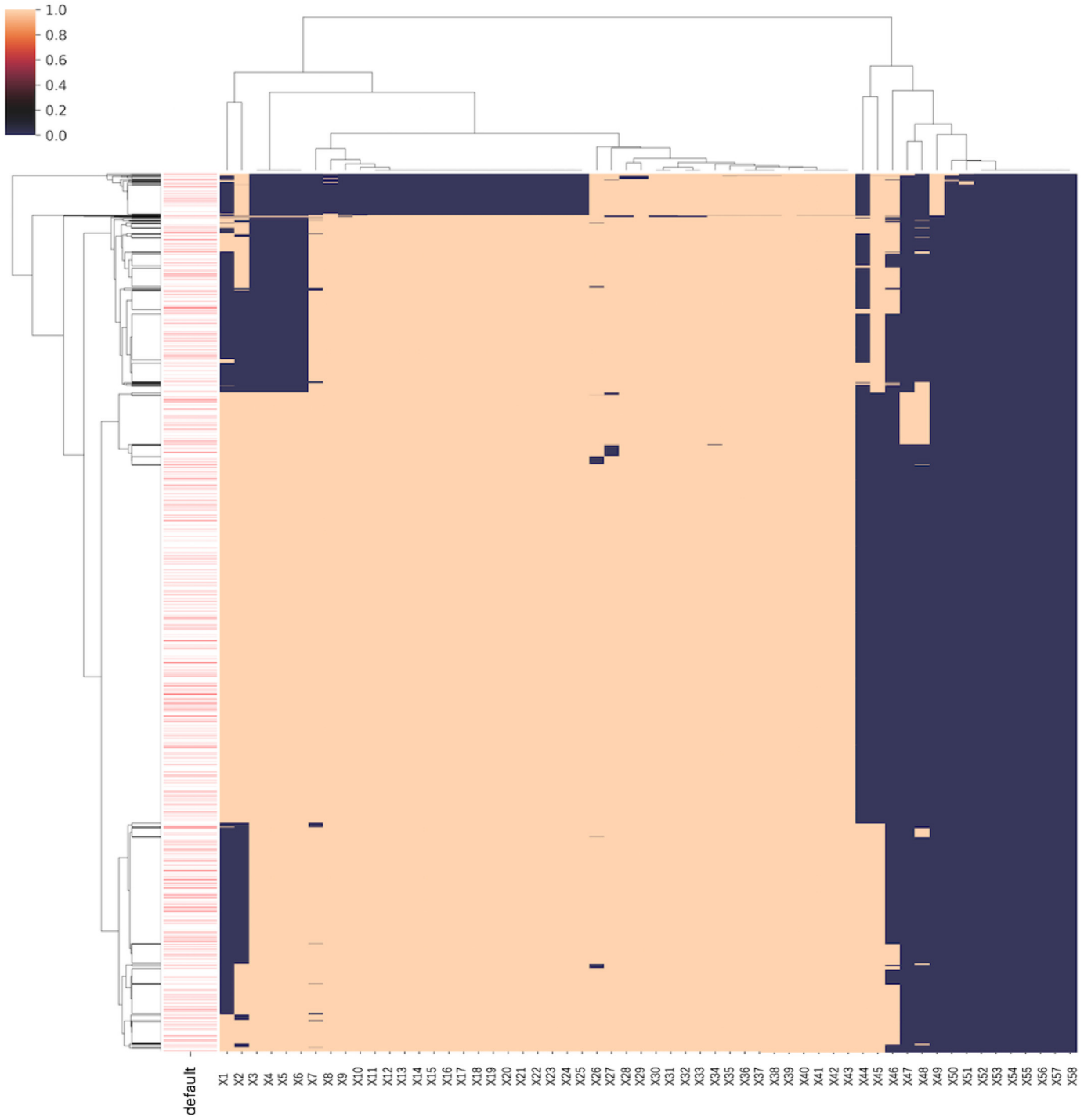

Fig. 4. The hierarchical clustering map describing data presence condition. Empty points and non-empty points are represented by dark color and light color, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

base learners. Although the other evaluation parameters are not the best ones, the SOM consensus model tends to produce more balanced performance, that is, balanced between recall and precision, and balanced between Type I error and Type II error. Note that the probabilities that one sample belonging to certain class cannot be measured quantitatively in SOM model, thus the AUC values based on probabilities are not calculated in this work.

The resulting SOM maps of consensus model based on individual models are shown in Fig. 6. Fig. 6a shows the map with conflicting neurons based on training set and Fig. 6b shows the map with neurons labeled by the most frequent class based on training set. Fig. 6b can be regarded as the "model" derived from Fig. 6a and it is used to classify the sample inputs of the test set that are accordingly shown in Fig. 6c. By comparing the SOM map of test set (Fig. 6c) and the "model" (Fig. 6b) built with the training set, we calculate the model performance parameters. It can be seen from Fig. 6a that most default samples of the training set are projected to the top right of the map, which means these samples follow a certain pattern. Moreover, most of the conflicting neurons (ones with both defaults and non-defaults) appear around the

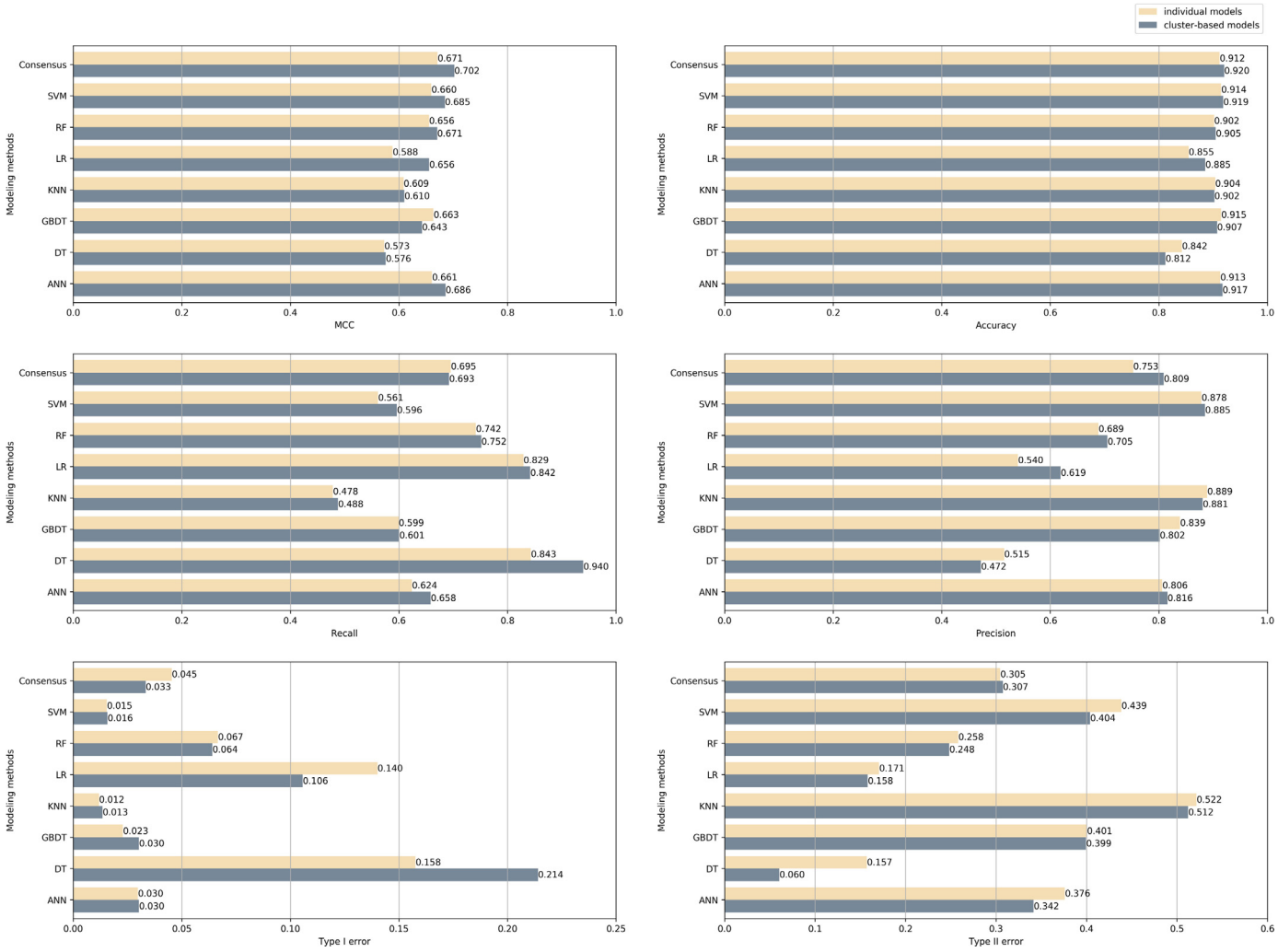

Fig. 5. Model performance comparisons between individual models and cluster-based model in terms of MCC, Accuracy, recall, precision, Type I error and Type II error. These models are built on based on Chinese P2P credit dataset.

joints of two classes, which means samples projected to these neurons are more difficult to be correctly predicted (usually with low confidence). Furthermore, the distance to the cluster centers can be interpreted as an indicator of confidence that one neuron belonging to a certain class, which we aim to measure quantitatively in our future work.

3.4. Performance of cluster-based models (route 3)

With respect to the route 3 in Fig. 2, supervised models were built based on subsets clustered by k-means. We adopted the same seven base learners (ANN, DT, GBDT, kNN, LR, RF and SVM) to make comparisons, aiming to explore whether the use of unsupervised clustering on a dataset can help improve predictive performance of credit scoring models. The results of cluster-based models can be seen in Table 3 and these results are also presented graphically in Fig. 5. According to the MCC values, six cluster-based models (cluster-based ANN, cluster-based DT, cluster-based kNN, cluster-based LR, cluster-based RF and cluster-based SVM) outperform their counterparts built with base learners to different degrees, with the only one exception of cluster-based GBDT. The clustering method improve the LR model by the greatest degree, with the MCC of 0.588 improved to 0.656. Followed by the clustering results based on the SVM model and ANN model, with the MCC of 0.660 improved to 0.685 and the MCC of 0.661 improved to 0.686, respectively. The MCC values of the other three models (RF, kNN, DT) were also slightly improved. Among all the cluster-based models, the cluster-based ANN has the best performance with the MCC of 0.686, which is better than that of the consensus model based on seven individual models (the MCC of 0.671). From the results above, we can conclude that the clustering method implemented on the Chinese P2P credit dataset indeed helps improve the performance of individual models.

3.5. Performance of SOM cluster-based consensus model (route 4)

With respect to the route 4 in Fig. 2, the SOM consensus strategy was applied to the previously built cluster-based models. Similar to the idea of the consensus model based on seven individual models, the cluster-based consensus model was built with the probabilities produced by the seven cluster-based models. We can find the performance of the cluster-based consensus model in terms of the MCC, AUC, ACC, recall, precision, Type I error and Type II error values in Table 3 and Fig. 5. It can be seen that the cluster-based consensus model achieves the best MCC of 0.702 among all the models built in this work. This suggests that the cluster-based consensus model is capable of obtaining a more accurate and reliable classifications. Moreover, it is inferred that the strategy of

Table 3
Model performance based on test set of Chinese P2P credit dataset.

Parameter	Individual models							Consensus model	Cluster-based models							cluster-based consensus model
	ANN	DT	GBDT	kNN	LR	RF	SVM		ANN	DT	GBDT	kNN	LR	RF	SVM	
MCC	0.661	0.573	0.663	0.609	0.588	0.656	0.660	0.671	0.686	0.576	0.643	0.610	0.656	0.671	0.685	**0.702**
AUC	0.918	0.931	0.949	0.868	0.941	0.937	0.939	–	0.939	0.950	0.933	0.848	**0.958**	0.954	0.947	–
ACC	0.913	0.842	0.915	0.904	0.855	0.902	0.914	0.912	0.917	0.812	0.907	0.902	0.885	0.905	0.919	**0.920**
recall	0.624	0.843	0.599	0.478	0.829	0.742	0.561	0.695	0.658	**0.940**	0.601	0.488	0.842	0.752	0.596	0.693
precision	0.806	0.515	0.839	0.889	0.540	0.689	0.878	0.753	0.816	0.472	0.802	0.881	0.619	0.705	**0.885**	0.809
Type I error	0.030	0.158	0.023	**0.012**	0.140	0.067	0.015	0.045	0.030	0.214	0.030	0.013	0.106	0.064	0.016	0.033
Type II error	0.376	0.157	0.401	0.522	0.171	0.258	0.439	0.305	0.342	**0.060**	0.399	0.512	0.158	0.248	0.404	0.307

Fig. 6. Resulting Kohonen's self-organizing maps of consensus model based on individual models for Chinese P2P credit dataset. (a) Map with conflicting neurons based on training set; (b) map with neurons labeled by the most frequent class based on training set; (c) map with conflicting neurons based on test set. Red ones represent defaults, green ones represent non-defaults, black ones represent the neurons with conflicting classes (defaults and non-defaults both exist), white ones represent the neurons without samples projected into. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

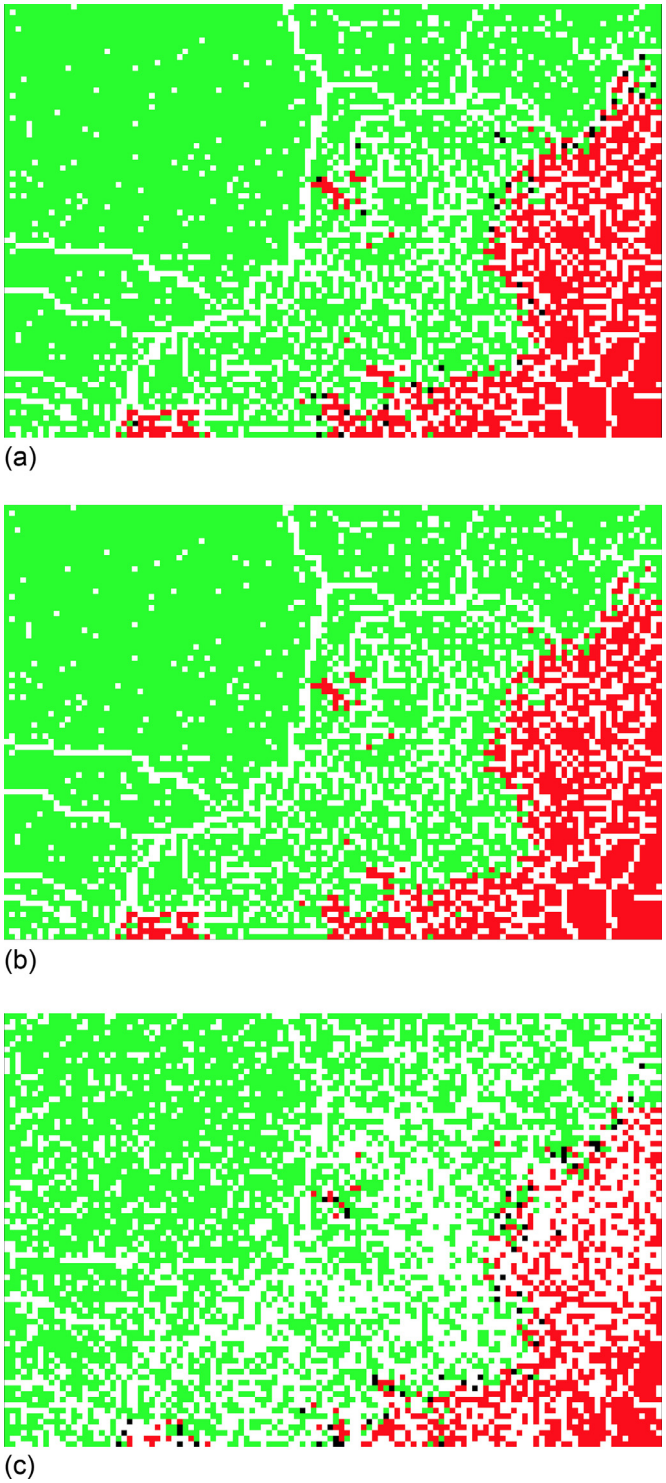

Fig. 7. Resulting Kohonen's self-organizing maps of the cluster-based consensus model for Chinese P2P credit dataset. (a) Map with conflicting neurons based on training set; (b) map with neurons labeled by the most frequent class based on training set; (c) map with conflicting neurons based on test set. Red ones represent defaults, green ones represent non-defaults, black ones represent the neurons with conflicting classes (defaults and non-defaults both exist), white ones represent the neurons without samples projected into. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 4
Model performances based on test sets of German and Australian credit dataset, respectively.

	Parameter	Individual models						Consensus model	cluster-based models						cluster-based consensus model
		ANN	DT	GBDT	kNN	LR	RF	SVM		ANN	DT	GBDT	kNN	LR	RF	SVM	
German dataset	MCC	0.479	0.335	0.338	0.280	0.497	0.448	0.393	0.490	0.480	0.302	0.403	0.407	0.503	0.420	0.529	**0.542**
	AUC	0.837	0.759	0.766	0.732	0.834	0.811	0.805	–	0.843	0.693	0.789	0.788	**0.859**	0.811	0.853	–
	ACC	0.787	0.647	0.74	0.708	0.74	0.735	0.760	0.792	0.786	0.633	0.753	0.768	0.758	0.721	**0.813**	0.808
	recall	0.574	0.77	0.418	0.443	**0.844**	0.754	0.459	0.574	0.658	0.748	0.604	0.541	0.820	0.757	0.649	0.721
	precision	**0.68**	0.454	0.607	0.524	0.548	0.548	0.651	0.693	0.603	0.411	0.549	0.588	0.542	0.497	0.667	0.635
	Type I error	**0.119**	0.406	0.119	0.176	0.306	0.273	0.108	0.112	0.166	0.410	0.190	0.145	0.266	0.293	0.124	0.159
	Type II error	0.426	0.230	0.582	0.557	**0.156**	0.246	0.541	0.426	0.342	0.252	0.396	0.459	0.180	0.243	0.351	0.279
Australia dataset	MCC	0.680	0.442	0.696	0.702	0.689	0.722	0.724	**0.725**	–	–	–	–	–	–	–	–
	AUC	0.937	0.904	0.927	0.889	0.934	0.936	0.941	–	–	–	–	–	–	–	–	–
	ACC	0.844	0.645	0.851	0.855	0.848	0.862	**0.866**	0.862	–	–	–	–	–	–	–	–
	recall	0.803	**0.983**	0.821	0.786	0.829	0.872	0.803	0.889	–	–	–	–	–	–	–	–
	precision	0.825	0.545	0.828	0.860	0.815	0.816	**0.870**	0.806	–	–	–	–	–	–	–	–
	Type I error	0.126	0.604	0.126	0.094	0.138	0.145	**0.088**	0.157	–	–	–	–	–	–	–	–
	Type II error	0.197	**0.017**	0.179	0.214	0.171	0.128	0.197	0.111	–	–	–	–	–	–	–	–

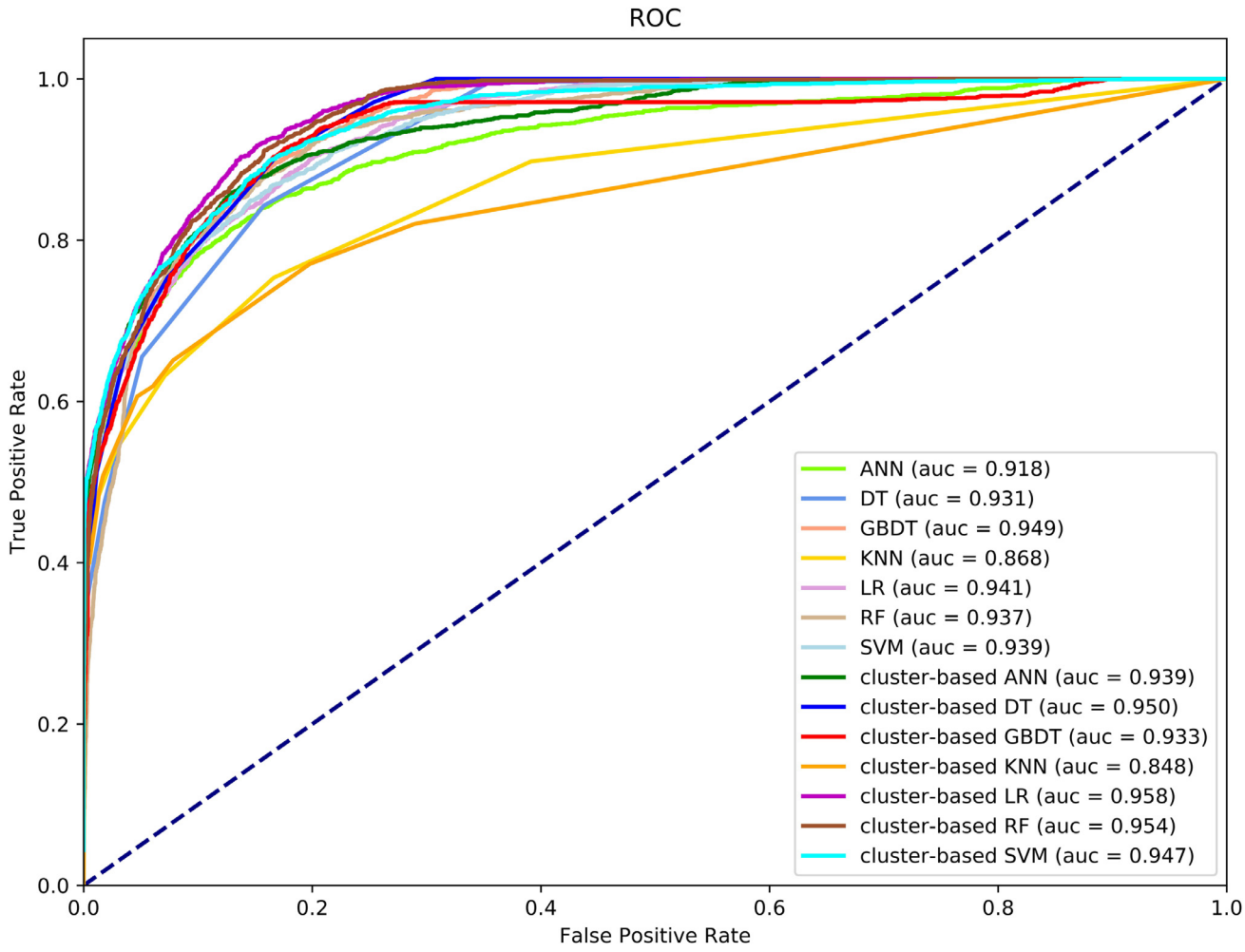

Fig. 8. The ROCs of individual models and cluster-based models with AUC values shown in the figure. Models are built on Chinese P2P credit dataset.

combining unsupervised learning and supervised learning at multiple stages proves to be effective based on the Chinese P2P credit dataset.

The resulting SOM maps of the cluster-based consensus model are shown in Fig. 7. The analysis and interpretation of Fig. 7 can be referred to that of Fig. 6. It can be seen from Fig. 7a that most default samples of the training set are projected to the right bottom of the map, which means the defaults follow a certain pattern. The distribution of the conflicting neurons sees the same pattern as that in Fig. 6 where conflicting neurons frequently appear around the joints of two classes. Additionally, in comparison with Fig. 6c, the neurons with conflicts in Fig. 7c are obviously reduced, which is consistent with the fact that the cluster-based consensus model outperforms the consensus model.

3.6. Evaluating on German credit dataset

In order to verify the effectiveness and stability of the ensemble strategy, we also applied the same methodologies to two benchmarking datasets – the German and Australian credit datasets.

As for the German credit dataset, models were built following the four routes as described in Fig. 2, with results in terms of MCC, AUC, ACC, recall, precision, Type I errors and Type II errors (expressed as percent number) shown in Table 4. Comparisons between individual models and cluster-based models and those between individual models and consensus model fare analyzed in the following aspects:

- The cluster-based consensus model achieves the best performance with the MCC of 0.542, which is consistent with the findings found based on the Chinese P2P credit dataset.
- The MCC values obtained from the German credit dataset are lower than those from the Chinese P2P credit dataset. This may be because of the differences between these two datasets with respect to the dataset size, data balance, feature dimensions and data quality. Nonetheless, in comparison with Xia's work (Xia et al., 2018) where the best ACC is reported to be 0.783, our cluster-based consensus model gives the slightly better performance with the ACC of 0.808.
- To prove the effectiveness of the consensus strategy, we compare the consensus model and individual models, the cluster-based consensus model and cluster-based models, respectively. As a result, consensus strategy helps consensus models to outperform most of the individual models.
- To prove the effectiveness of the dataset clustering strategy, we respectively compare the cluster-based models and individual models and find that this strategy helps to improve performance in most cases. In addition, the clustering strategy can help reduce the Type II error to a great extent.

3.7. Evaluating on Australian credit dataset

As for the Australian credit dataset, we only compare the performance between the consensus model and the individual models. Since there are not many missing points in this dataset, we

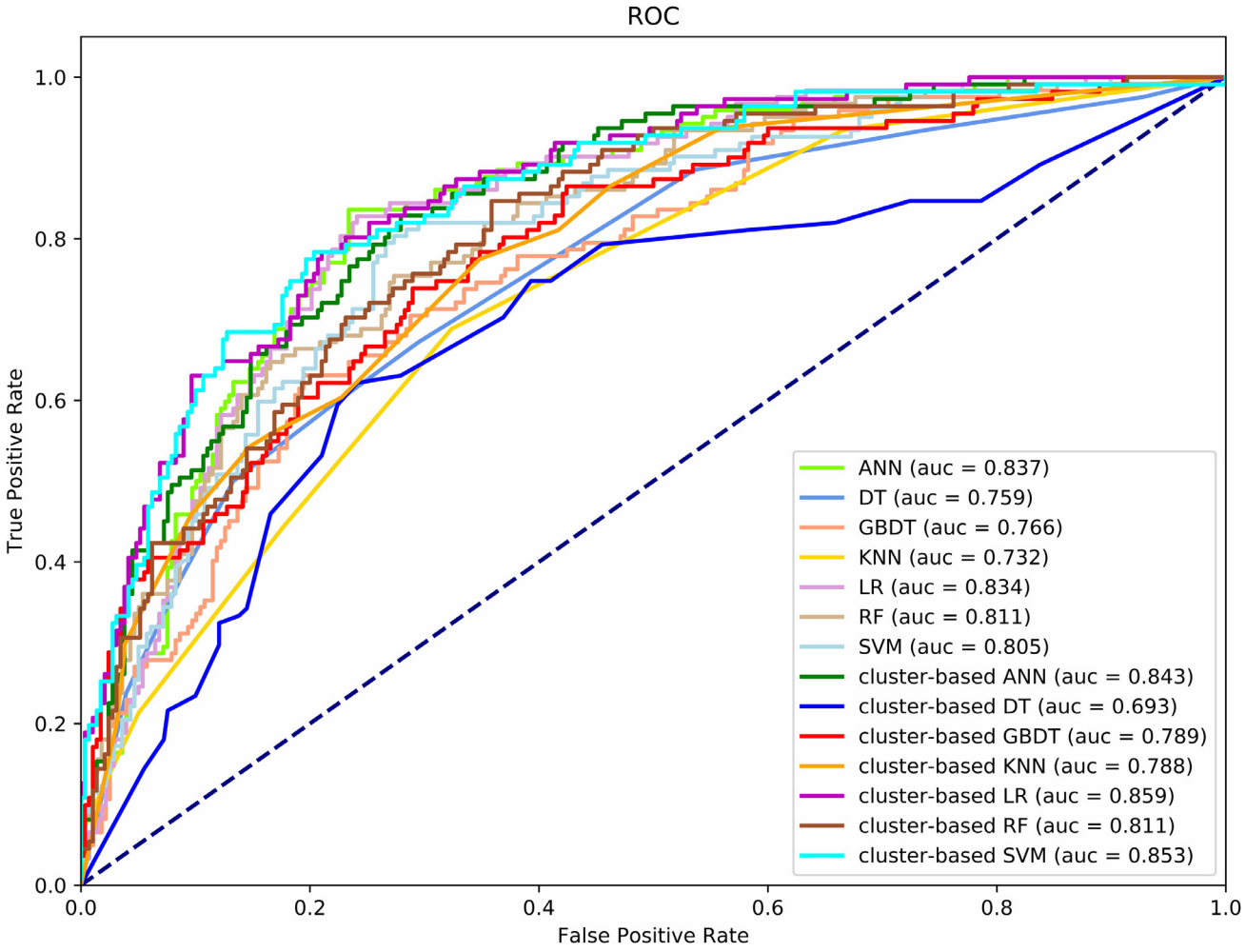

Fig. 9. The ROCs of individual models and cluster-based models with AUC values shown in the figure. Models are built on German credit dataset.

didn't cluster the dataset before building the models. The results are shown in Table 4, and analyzed in the following aspects:

- The consensus model achieves the best performance with the MCC of 0.725, which is consistent with the findings found based on the Chinese P2P credit dataset and the German credit dataset. It is inferred that consensus strategy is effective to help improve the model performance of base learners.
- In comparison with Xia's work (Xia et al., 2018) where the best ACC is reported to be 0.863, our consensus model gives the similar performance with the ACC of 0.862.

To summarize, regarding to the two benchmarking credit datasets, the consensus strategy has proven to be efficient and reliable as the consensus models can outperform most of the individual models. With regard to the German credit dataset, the clustering method can improve the performance of individual models and integrated with consensus strategy can further improve the performance. These results demonstrate the superiority of the integration of unsupervised learning methods to the supervised individual models.

3.8. ROC curve analysis

ROC curve can be a good indicator of the balance between type I error and type II error which are two key parameters with a great effect on the profitability of financial institutions.

The ROC comparison between classifiers based on the Chinese P2P credit dataset is depicted in Fig. 8, with AUC values shown in the figure. It can be seen from Fig. 8 that the GBDT model with AUC of 0.949 and the cluster-based LR model with AUC of 0.958 achieve the highest AUCs among the individual models and cluster-based models, respectively. In most cases, performances of the cluster-based models are better than those of the individual models, which is consistent with the conclusion drawn from other measuring parameters (e.g. MCC and ACC) that dataset clustering can help to improve the predictive performance. Given that LR models perform well in terms of AUC, although these LR models perform comparatively badly in terms of MCC and ACC, we cannot ignore the fact that the LR models exhibit a good level of capability on ranking the defaults against non-defaults and on achieving balance between the type I error and type II error. Except kNN models, all the other fourteen models have the AUC values above 0.9, which means the models built in this work perform well on discriminating defaults and non-defaults on the Chinese P2P credit dataset.

Very similar trend can be observed in the German and Australian credit datasets, shown in Figs. 9 and 10. The cluster-based LR model with AUC of 0.859 and the SVM model with AUC of 0.941 score the best AUCs for the German and Australian credit datasets, respectively.

It is worth noting the AUC is not the only parameter to evaluate the ROC curves. We can also observe that there are some intersects among these curves of different models, which means

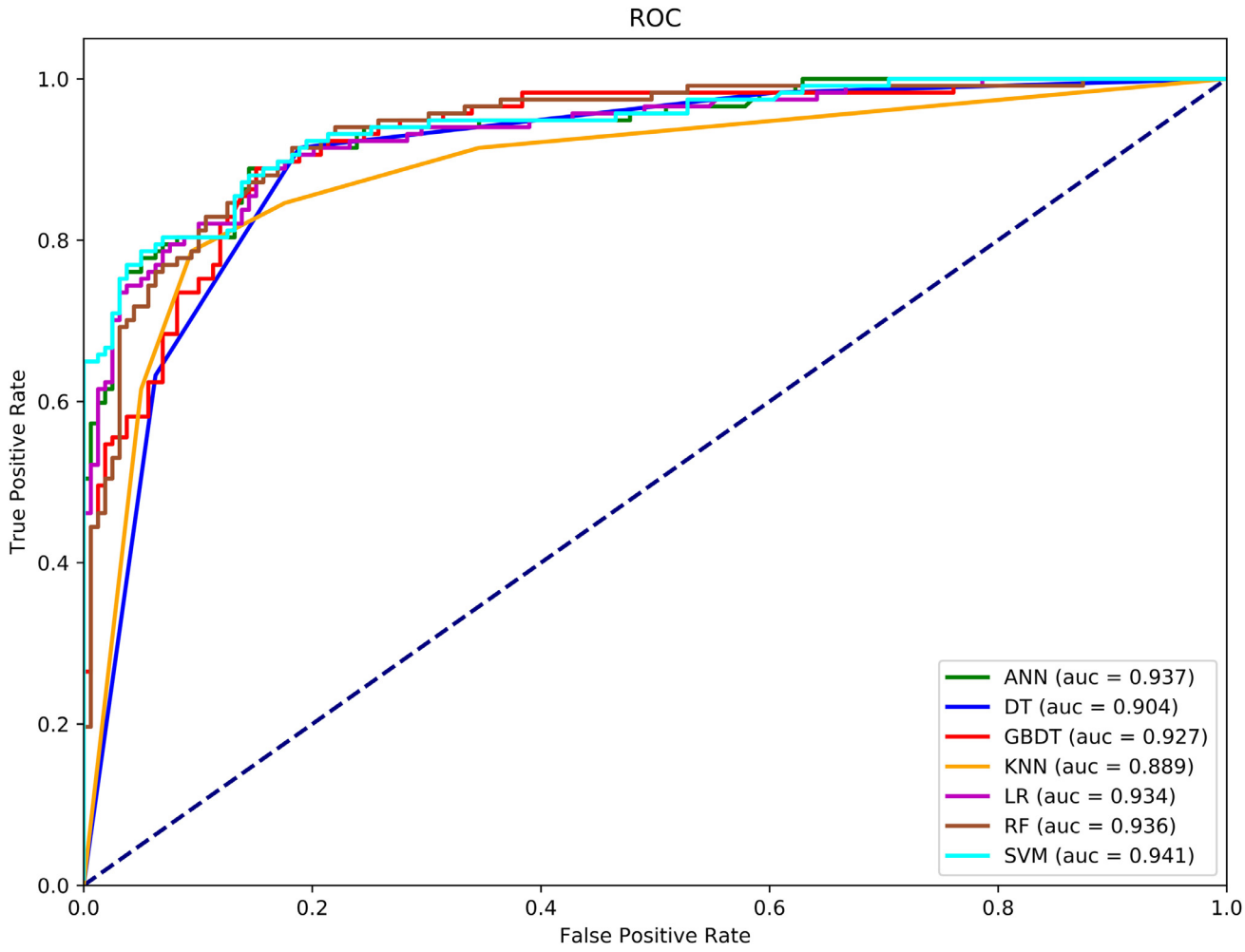

Fig. 10. The ROCs of individual models and cluster-based models with AUC values shown in the figure. Models are built on Australian credit dataset.

model performance varies with the cut-off thresholds (related to the ratio between the type I error and type II error). Under these circumstances, using a reasonable ratio between the type I error and type II error calculated based on the reality of operating for each financial institution would be a wise choice.

4. Conclusions and future work

With the credit industry growing and prospering rapidly, credit scoring has become a more and more important tool to discriminate good applicants and bad applicants and thus to manage credit risk, as the predictive performance of credit scoring models has a great effect on the profitability of financial institutions. In this work, we propose the strategy of integrating unsupervised learning with supervised learning in two different stages (building consensus model and clustering datasets) to construct credit scoring models.

The strategy is tested on three datasets following four routes: 1) building individual models; 2) building SOM consensus model based on the individual models; 3) building models based on clustered datasets; 4) building SOM consensus model based on the cluster-based models. Tables 3 and 4 comprehensively present the performance comparisons of models from four routes in terms of seven evaluation parameters. The MCC values of supervised individual models have mostly observed increases when using consensus model and dataset clustering method, which means these two scenarios both are effective at improving the performance of credit scoring models. Moreover, the combination of the two scenarios achieves the best performance, thereby suggesting that integration of unsupervised and supervised learning at different stages contributes to the model performance. The results in this work confirm the superiority of this proposed strategy, which boost our confidence that this strategy can be extended to many other credit datasets from financial institutions especially Chinese P2P enterprises.

Our work can be improved or complemented in several aspects. Firstly, the unsupervised machine learning methods can be investigated more, with an aim to reducing the noise of data, for instance, identifying outliers and removing abnormal features, since we believe improving the data quality is of critical importance and of practical use in credit scoring models. Secondly, efforts can be made to improve the interpretability of the ensemble model, that is, making the knowledge obtained by the ensemble model more easily comprehensible by the average users. In addition, the use of SOM in this work can be explored so that we can measure the class probability quantitatively.

In sum, this paper provides a successful application of integration of unsupervised and supervised machine learning in credit risk assessment. It is hoped that it will motivate future theoretical and empirical investigations into combining unsupervised learning techniques with supervised models and perhaps help develop more effective combination strategies in credit risk assessment.

Conflict of interest

The authors confirm that they have no conflicts of interest.

Credit authorship contribution statement

Wang Bao: Formal analysis, Methodology, Software, Formal analysis, Investigation, Resources, Data curation, Writing - original draft, Writing - review & editing, Visualization, Project administration. **Ning Lianju:** Formal analysis, Writing - review & editing, Supervision, Project administration, Funding acquisition. **Kong Yue:** Formal analysis, Methodology, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Writing - review & editing, Visualization.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No. 71271032) and supported by Beijing Municipal Natural Science Foundation (No. 9182012). We thank the Molecular Networks GmbH, Erlangen, Germany for providing the programs SONNIA for our scientific work.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.eswa.2019.02.033.

References

Abellán, J., & Castellano, J. G. (2017). A comparative study on base classifiers in ensemble methods for credit scoring. *Expert Systems with Applications, 73*, 1–10.
AghaeiRad, A., Chen, N., & Ribeiro, B. (2017). Improve credit scoring using transfer of learned knowledge from self-organizing map. *Neural Computing and Applications, 28*(6), 1329–1342.
Ala, M. (2015). A systematic credit scoring model based on heterogeneous classifier ensembles. *The proceedings of the IEEE 2015 international symposium on innovations in intelligent systems and applications (INISTA)*.
Ala, M., & Abbod, M. F. (2016a). A new hybrid ensemble credit scoring model based on classifiers consensus system approach. *Expert Systems with Applications, 64*, 36–55.
Ala, M., & Abbod, M. F. (2016b). Classifiers consensus system approach for credit scoring. *Knowledge-Based Systems, 104*, 89–105.
Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance, 23*(4), 589–609.
Asgharbeygi, N., & Maleki, A. (2008). Geodesic K-means clustering. *2008 19th international conference on pattern recognition*. doi:10.1109/ICPR.2008.4761241.
Australian-dataset. (1987). Australian dataset. (1987). Australian credit approval data. http://archive.ics.uci.edu/ml/datasets/Statlog+%28Australian+Credit+Approvl%29/. Last Checked on 17 Oct. 2018.
Ben-david, A., & Frank, E. (2009). Accuracy of machine learning models versus "hand crafted" expert systems – A credit scoring case study. *Expert Systems with Applications, 36*(3), 5264–5271.
Bequé, A., & Lessmann, S. (2017). Extreme learning machines for credit scoring : An empirical evaluation. *Expert Systems with Applications, 86*, 42–53.
Bijak, K., & Thomas, L. C. (2012). Does segmentation always improve model performance in credit scoring ? *Expert Systems with Applications, 39*(3), 2433–2442.
Bishop, C. M. (1997). Neural networks for pattern recognition. *Journal of the American Statistical Association, 92*(440), 1642–1645.
Breiman, L. (1999). Random forest. *Machine Learning, 45*(5), 1–35.
Chen, N., Ribeiro, B., & Chen, A. (2016). Financial credit risk assessment: A recent review. *Artificial Intelligence Review, 45*(1), 1–23.
Cleofas-Sánchez, L., García, V., Marqués, A. I., & Sánchez, J. S. (2016). Financial distress prediction using the hybrid associative memory with translation. *Applied Soft Computing, 44*, 144–152.
Cortes, C., & Vapnik, V. (1995). Support vector machine. *Machine Learning*, 1303–1308.
Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 13*(1), 21–27.
Dahiya, S. (2017). A feature selection enabled hybrid - bagging algorithm for credit risk evaluation. *Expert Systems, 34*(6), e12217.
Ester, M., Kriegel, H., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *International Conference on Knowledge Discovery and Data Mining*, 226–231.
Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters, 27*(8), 861–874.
Fernandes, M. B., Scotti, M. T., Ferreira, M. J. P., & Emerenciano, V. P. (2008). Use of self-organizing maps and molecular descriptors to predict the cytotoxic activity of sesquiterpene lactones. *European Journal of Medicinal Chemistry, 43*(10), 2197–2205.
Florez-Lopez, R. (2010). Effects of missing data in credit risk scoring. A comparative analysis of methods to achieve robustness in the absence of sufficient data. *Journal of the Operational Research Society, 61*(3), 486–501.
Florez-lopez, R., & Ramon-jeronimo, J. M. (2015). Enhancing accuracy and interpretability of ensemble strategies in credit risk assessment. A correlated-adjusted decision forest proposal. *Expert Systems with Applications, 42*(13), 5737–5753.
Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics, 29*(5), 1189–1232.
García-Laencina, P. J., Sancho-Gómez, J. L., & Figueiras-Vidal, A. R. (2010). Pattern classification with missing data: A review. *Neural Computing and Applications, 19*(2), 263–282.
García, V., Marqués, A. I., & Sánchez, J. S. (2012). On the use of data filtering techniques for credit risk prediction with instance-based models. *Expert Systems with Applications, 39*(18), 13267–13276.
German-dataset. (1994). German dataset. (1994). German cash loan data. http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german/. Last Checked on 17 Oct. 2018.
Guyon, I., & Elisseeff, A. (2011). An introduction to variable and feature selection. *Journal of Machine Learning Research, 3*, 1157–1182.
Henley, W. E., & Hand, D. J. (1996). A k-nearest-neighbour classifier for assessing consumer credit risk. *Journal of the Royal Statistical Society, 45*(1), 77–95.
Holmes, C. C., & Adams, N. M. (2002). A probabilistic nearest neighbour method for statistical pattern recognition. *Journal of the Royal Statistical Society, 64*(2), 295–306.
Huysmans, J., Baesens, B., Vanthienen, J., & Van Gestel, T. (2006). Failure prediction with self organizing maps. *Expert Systems with Applications, 30*(3), 479–487.
Islam, M. J., Wu, Q. M. J., Ahmadi, M., & Sid-Ahmed, M. A. (2007). Investigating the performance of Naive- Bayes classifiers and K- nearest neighbor classifiers. *2007 international conference on convergence information technology, ICCIT 2007*.
Kodinariya, T. M., & Makwana, P. R. (2013). Review on determining number of cluster in *k*-means clustering. *International Journal of Advance Research in Computer Science and Management Studies, 1*(6), 90–95.
Kohonen, T. (1998). The self-organizing map. *Neurocomputing, 21*(1–3), 1–6.
Kong, Y., & Yan, A. (2017). QSAR models for predicting the bioactivity of Polo-like Kinase 1 inhibitors. *Chemometrics and Intelligent Laboratory Systems, 167*, 214–225.
Lamrous, S., & Taïleb, M. (2007). Divisive hierarchical K-means. *CIMCA 2006: international conference on computational intelligence for modelling, control and automation, jointly with IAWTIC 2006: international conference on intelligent agents web technologies*.
Lee, T. S., Chiu, C. C., Chou, Y. C., & Lu, C. J. (2006). Mining the customer credit using classification and regression tree and multivariate adaptive regression splines. *Computational Statistics & Data Analysi, 50*(4), 1113–1130.
Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research, 247*(1), 124–136.
Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008). Isolation forest. In *Proceedings - IEEE international conference on data mining*. ICDM.
Luo, S., Cheng, B., & Hsieh, C. (2009). Prediction model building with clustering-launched classification and support vector machines in credit scoring. *Expert Systems with Applications, 36*(4), 7562–7566.
Luo, S., Kong, X., & Nie, T. (2016). Spline based survival model for credit risk modeling. *European Journal of Operational Research, 253*(3), 869–879.
Malekipirbazari, M., & Aksakalli, V. (2015). Risk assessment in social lending via random forests. *Expert Systems with Application, 42*(10), 4621–4631.
Malhotra, R., & Malhotra, D. K. (2003). Evaluating consumer loans using neural networks. *Omega, 31*(2), 8396.
Marqués, A. I., García, V., & Sánchez, J. S. (2012). Exploring the behaviour of base classifiers in credit scoring ensembles. *Expert Systems with Applications, 39*(11), 10244–10250.
Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*(10), 2825–2830.
Powers, D. M. W. (2011). Evaluation: From precision, recall and F-measure to Roc, informedness, markedness & correlation. *Journal of Machine Learning Technologies, 2*(1), 37–63.
Quinlan, J. R. (1993). *C4.5: programs for machine learning*. California: Morgan Kaufmann San Mateo.
Ruppert, D. (2004). The elements of statistical learning: Data mining, inference, and prediction. *Journal of the American Statistical Association, 99*(466), 567.
Saeys, Y., Inza, I., & Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics, 23*(19), 2507–2517.
Sohn, S. Y., Kim, D. H., & Yoon, J. H. (2016). Technology credit scoring model with fuzzy logistic regression. *Applied Soft Computing Journal, 43*, 150–158.
SONNIA, version 4.2; Molecular Networks GmbH: Germany and Altamira, LLC, USA, (2016).; https://www.mn-am.com/products/sonnia.
Sun, J., & Li, H. (2015). Dynamic credit scoring using B & B with incremental-SVM-ensemble. *Kybernetes, 44*(4), 518–535.
Thomas, L. C., Oliver, R. W., & Hand, D. J. (2005). A survey of the issues in consumer credit modelling research. *Journal of the Operational Research Society, 56*(9), 1006–1015.
Twala, B. (2010). Multiple classifier application to credit risk assessment. *Expert Systems With Applications, 37*(4), 3326–3336.
Twala, B. (2013). Impact of noise on credit risk prediction : Does data quality really matter? *Intelligent Data Analysis, 17*(6), 1115–1134.
Wang, G., Hao, J., Ma, J., & Jiang, H. (2011). A comparative assessment of ensemble learning for credit scoring. *Expert Systems with Applications, 38*(1), 223–230.
Wang, G., Ma, J., Huang, L., & Xu, K. (2012). Two credit scoring models based on dual strategy ensemble trees. *Knowledge-Based Systems, 26*, 61–68.

West, D. (2000). Neural network credit scoring models. *Computers and Operations Research, 27*(11), 1131–1152.
Wiginton, J. C. (1980). A note on the comparison of logit and discriminant models of consumer credit behavior. *The Journal of Financial and Quantitative Analysis, 15*(3), 757–770.
Xia, Y., Liu, C., Da, B., & Xie, F. (2018). A novel heterogeneous ensemble credit scoring model based on bstacking approach. *Expert Systems with Applications, 93*, 182–199.
Xu, X., Zhou, C., & Wang, Z. (2009). Credit scoring algorithm based on link analysis ranking with support vector machine. *Expert Systems with Applications, 36*(2), 2625–2632.
Yan, A., Nie, X., Wang, K., & Wang, M. (2013). Classification of Aurora kinase inhibitors by self-organizing map (SOM) and support vector machine (SVM). *European Journal of Medicinal Chemistry, 61*, 73–83.
Yu, L., Yue, W., Wang, S., & Lai, K. K. (2010). Support vector machine based multiagent ensemble learning for credit risk evaluation. *Expert Systems with Applications, 37*(2), 1351–1360.
Zhou, L., Lai, K. K., & Yu, L. (2010). Least squares support vector machines ensemble models for credit scoring. *Expert Systems with Applications, 37*(1), 127–133.
Zhou, L., Si, Y. W., & Fujita, H. (2017). Predicting the listing statuses of Chinese-listed companies using decision trees combined with an improved filter feature selection method. *Knowledge-Based Systems, 128*, 93–101.
Zhou, L., Tam, K. P., & Fujita, H. (2016). Predicting the listing status of Chinese listed companies with multi-class classification models. *Information Sciences, 328*, 222–236.