Expert Systems with Applications 38 (2011) 223–230

Contents lists available at ScienceDirect

Expert Systems with Applications

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

A comparative assessment of ensemble learning for credit scoring

Gang Wang [a,b,*], Jinxing Hao [b,c], Jian Ma [b], Hongbing Jiang [b]

[a] School of Management, Hefei University of Technology, Hefei, Anhui 230009, PR China
[b] Department of Information Systems, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong
[c] School of Economics and Management, BeiHang University, Beijing 100083, PR China

ARTICLE INFO

Keywords:
Credit scoring
Ensemble learning
Bagging
Boosting
Stacking

ABSTRACT

Both statistical techniques and Artificial Intelligence (AI) techniques have been explored for credit scoring, an important finance activity. Although there are no consistent conclusions on which ones are better, recent studies suggest combining multiple classifiers, i.e., ensemble learning, may have a better performance. In this study, we conduct a comparative assessment of the performance of three popular ensemble methods, i.e., Bagging, Boosting, and Stacking, based on four base learners, i.e., Logistic Regression Analysis (LRA), Decision Tree (DT), Artificial Neural Network (ANN) and Support Vector Machine (SVM). Experimental results reveal that the three ensemble methods can substantially improve individual base learners. In particular, Bagging performs better than Boosting across all credit datasets. Stacking and Bagging DT in our experiments, get the best performance in terms of average accuracy, type I error and type II error.

1. Introduction

The recent world financial tsunami arouses unprecedented attention of financial institutions on credit risk. Credit scoring has become one of the primary ways for financial institutions to assess credit risk, improve cash flow, reduce possible risks and make managerial decisions (Huang, Chen, & Wang, 2007).

The purpose of credit scoring is to classify the applicants into two types: applicants with good credit and applicants with bad credit. Applicants with good credit have great possibility to repay financial obligation. Applicants with bad credit have high possibility of defaulting. The accuracy of credit scoring is critical to financial institutions' profitability. Even 1% of improvement on the accuracy of credit scoring of applicants with bad credit will decreases a great loss for financial institutions (Hand & Henley, 1997).

Credit scoring was originally evaluated subjectively according to personal experiences, and later it was based on 5Cs: the character of the consumer, the capital, the collateral, the capacity and the economic conditions. But with the tremendous increase of applicants, it is impossible to conduct the work manually. Two categories of automatic credit scoring techniques, i.e., statistical techniques and Artificial Intelligence (AI) techniques, have been studied by prior researches (e.g., Huang, Chen, Hsu, Chen, & Wu, 2004).

Some statistical techniques have been widely applied to build the credit scoring models, such as Linear Discriminant Analysis (LDA) (Karels & Prakash, 1987; Reichert, Cho, & Wagner, 1983), Logistic Regression Analysis (LRA) (Thomas, 2000; West, 2000), Multivariate Adaptive Regression Splines (MARS) (Friedman, 1991). However, the problem with applying these statistical techniques to credit scoring is that some assumptions, such as the multivariate normality assumptions for independent variables, are frequently violated in the practice of credit scoring, which makes these techniques theoretically invalid for finite samples (Huang et al., 2004).

In recent years, many studies have demonstrated that AI techniques such as Artificial Neural Networks (ANN) (Desai, Crook, & Overstreet, 1996; West, 2000), Decision Tree (DT) (Hung & Chen, 2009; Makowski, 1985), Case-Based Reasoning (CBR) (Buta, 1994; Shin & Han, 2001), and Support Vector Machine (SVM) (Baesens et al., 2003; Huang et al., 2007; Schebesch & Stecking, 2005) can be used as alternative methods for credit scoring. In contrast with statistical techniques, AI techniques do not assume certain data distributions. These techniques automatically extract knowledge from training samples. According to previous studies, AI techniques are superior to statistical techniques in dealing with credit scoring problems, especially for nonlinear pattern classification (Huang et al., 2004).

However, there is no overall best AI techniques used in building credit scoring models, for what is best depends on the details of the problem, the data structure, the characteristics used, the extent to which it is possible to segregate the classes by using those characteristics, and the objective of the classification (Hand & Henley, 1997; Yu, Wang, & Lai, 2008). Recently, there is a growing interest

* Corresponding author at: Department of Information Systems, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong. Tel.: +852 9799 0955; fax: +852 2788 8694.
E-mail address: wgedison@gmail.com (G. Wang).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.06.048

that existing applications of single AI technique can be further improved by ensemble methods. Latest researches (Hung & Chen, 2009; Yu et al., 2008) have shown that such ensemble methods have performed better than single AI technique for credit scoring. However, the application of ensemble methods in credit scoring is a relatively new and untried area. To the best of our knowledge, this may be the first attempt to systematically compare of classical ensemble methods for credit scoring.

Base on these considerations, we conduct a comparative assessment of the performance of three popular ensemble methods—Bagging, Boosting, and Stacking—on credit scoring problems. The aim of this study is to examine the performance of different ensemble methods for the field of credit scoring in terms of average accuracy, type I error and type II error. Besides two common used datasets, i.e., Australian and German credit datasets, which are from UCI machine learning repository (Asuncion & Newman, 2007), our studies use a new credit dataset from China, collected mainly by the Industrial and Commercial Bank of China. In experiments we choose four popular methods in the literature, i.e., LRA, DT, ANN and SVM, as base learner. The results reveal that the application of ensemble learning can bring substantial improvement for individual base learner. Especially in our experiments, Bagging performs better than Boosting across all datasets. In addition, Stacking, and Bagging DT get best results in terms of three performance indicators, i.e., average accuracy, type I error and type II error. And among four base learners, DT gets best improvement in terms of three performance indicators after the application of ensemble learning.

The remainder of the paper is organized as follows. In Section 2, the details of three different types of ensemble methods for credit scoring are presented. In Section 3, we present the details of experimental design. Section 4 reports the experimental results. Based on the observations and results of these experiments, Section 5 draws conclusions and future research directions.

2. Ensemble learning for credit scoring

2.1. Overviews of ensemble learning

Ensemble learning is a machine learning paradigm where multiple learners are trained to solve the same problem (Polikar, 2006). In contrast to ordinary machine learning approaches that try to learn one hypothesis from the training data, ensemble methods try to construct a set of hypotheses and combine them to use (Zhou, 2009). Learners composed of an ensemble are usually called base learners.

One of the earliest studies on ensemble learning is Dasarathy and Sheela's research (1979), which discusses partitioning the feature space using two or more classifiers. In 1990, Hansen and Salamon showed that the generalization performance of an ANN can be improved using an ensemble of similarly configured ANNs (Hansen & Salamon, 1990). While Schapire proved that a strong classifier in probably approximately correct (PAC) sense can be generated by combining weak classifiers through Boosting (Schapire, 1990), the predecessor of the suite of AdaBoost algorithms. Since these seminal works, studies in ensemble learning have expanded rapidly, appearing often in the literature under many creative names and ideas (Polikar, 2006).

The generalization ability of an ensemble is usually much stronger than that of a single learner, which makes ensemble methods very attractive (Dietterich, 1997). In practice, to achieve a good ensemble, two necessary conditions should be satisfied: accuracy and diversity (Windeatt & Ardeshir, 2004). In the next three subsections, we will introduce three popular ensemble methods, i.e., Bagging, Boosting, and Stacking, respectively.

2.2. Bagging

Bagging (short for bootstrap aggregating) is one of the earliest ensemble learning algorithms (Breiman, 1996). It is also one of the most intuitive and simplest to implement, with a surprisingly good performance. Diversity in Bagging is obtained by using bootstrapped replicas of the training data: different training data subsets are randomly drawn—with replacement—from the entire training data. Each training data subset is used to train a different base learner of the same type.

The base learners' combination strategy for Bagging is majority vote. Simple as it is, this strategy can reduce variance when combined with the base learner generation strategies. The pseudo-code of Bagging algorithm is shown in Fig. 1.

Bagging is particularly appealing when the available data is of limited size. To ensure that there are sufficient training samples in each subset, relatively large portions of the samples (75–100%) are drawn into each subset. This causes individual training subsets to overlap significantly, with many of the same instances appearing in most subsets, and some instances appearing multiple times in a given subset. In order to ensure diversity under this scenario, a relatively unstable base learner is used so that sufficiently different decision boundaries can be obtained for small perturbations in different training datasets.

2.3. Boosting

Boosting (Freund & Schapire, 1996; Schapire, 1990) encompasses a family of methods. Unlike Bagging, Boosting creates different base learners by sequentially reweighting the instances in the training dataset. Each instance misclassified by the previous base learner will get a larger weight in the next round of training.

Input: Data set $D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\}$;
Base learning algorithm L;
Number of learning rounds T.

Process:
For $t = 1, 2, \cdots, T$:
$D_t = Bootstrap(D)$; % Generate a bootstrap sample from D
$h_t = L(D_t)$ % Train a base learner h_t from the bootstrap sample
end.

Output: $H(x) = \arg\max_{y \in Y} \sum_{t=1}^{T} 1(y = h_t(x))$ % the value of $1(\alpha)$ is 1 if α is true
% and 0 otherwise

Fig. 1. The Bagging algorithm.

The basic idea of Boosting is to repeatedly apply a base learner to modified versions of the training dataset, thereby producing a sequence of base learners for a predefined number of iterations. To begin with, all the instances are initialized with uniform weights. After this initialization, each boosting iteration fits a base learner to the weighted training data. Error is computed and the weight of the correctly classified instances is lowered while the incorrectly classified instances will get higher weights. The final model obtained by the Boosting algorithm is a linear combination of several base learners weighted by their own performance.

Even though there are several versions of the Boosting algorithms, the most widely used is the one proposed by Freund and Schapire (1996) which is known as AdaBoost. Therefore, we use the AdaBoost algorithm in this study. The pseudo-code of AdaBoost algorithm is given in Fig. 2.

2.4. Stacking

Stacking is another popular ensemble learning and general method of using a high-level base learner to combine lower level base learners to achieve greater predictive accuracy (Wolpert, 1992). Although developed some years ago, it is less widely used than Bagging and Boosting, partly because it is difficult to analyze theoretically. Unlike Bagging and Boosting, Stacking is not normally used to combine base learners of the same type. Instead it is applied to base learners built by different learning algorithms.

Input: Data set $D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_m,y_m)\}$;
Base learning algorithm L;
Number of learning rounds T.

Process:

$D_1(i)=1/m.$ % Initialize the weight distribution

For $t=1,2,\cdots,T$:

$h_t=L(D,D_t)$; % Train a base learner h_t from D using distribution D_t

$\varepsilon_t=\Pr_{i\sim D_t}[h_t(x_i)\neq y_i)]$; % Measure the error of h_t

$\alpha_t=\frac{1}{2}\ln\frac{1-\varepsilon_t}{\varepsilon_t}$; % Determine the weight of h_t

$D_{t+1}(i)=\frac{D_t(i)}{Z_t}\times\begin{cases}\exp(-\alpha_t) & if\ h_t(x_i)=y_i\\ \exp(\alpha_t) & f\ h_t(x_i)\neq y_i\end{cases}$ % Update the distribution, where Z_t is a

$=\frac{D_t(i)\exp(-\alpha_t y_i h_t(x_i))}{Z_t}$ % normalization factor with enables D_{t+1} to be a distribution

end.

Output: $H(x)=\mathrm{sign}(f(x))=\mathrm{sign}\sum_{t=1}^{T}\alpha_t h_t(x)$

Fig. 2. The AdaBoost algorithm.

Input: Data set $D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_m,y_m)\}$;
First-level learning algorithm $L_1,\cdots,L_T$;
Second-level learning algorithm L;

Process:

For $t=1,2,\cdots,T$:

$h_t=L_t(D)$ % Train a first-level individual learner h_t by applying the first-level
end; % learning a algorithm L_t to the original data set D

$D'=\Phi$; % Generate a new data set

For $t=1,2,\cdots,m$:

For $t=1,2,\cdots,T$:

$z_{it}=h_t(x_i)$ % Use h_t to classify the training example x_i

end;

$D'=D'\bigcup\{((z_{i1},z_{i2},\cdots,z_{iT}),y_i)\}$

end;

$h'=L(D')$. % Train the second-level learner h' by applying the second-level
% learning algorithm L to the new data set D'

Output: $H(x)=h'(h_1(x),h_2(x),\cdots,h_T(x))$

Fig. 3. The Stacking algorithm.

Stacking deals with the task of learning a meta-level base learner to combine the prediction of multiple base-level base learners. The set of base-level base learners is generated by applying different learning algorithms to a given dataset. The first step of stacking is to collect the output of each base learner into a new set of data. For each instance in the original training set, this dataset represents every base learner's prediction of that instance's class, along with its true classification. During this step, care is taken to ensure that the base learners are formed from a batch of training data that does not include the instance in question, in just the same way as ordinary cross-validation. The new data is treated as the data for another learning problem, and in the second step a learning algorithm is employed to solve this problem. The original data and the base learners constructed for it in the first step are referred to as level-1 data and level-1 classifiers, respectively, while the set of cross-validated data and the second-stage learning algorithm are referred to as level-2 data and the level-2 classifier. The pseudo-code of Stacking algorithm is shown in Fig. 3.

3. Experimental design

3.1. Real world credit dataset

Three real world credit datasets are used to evaluate the performance of the three ensemble methods: Australian credit dataset, German credit dataset and China credit dataset. The first two are from UCI machine learning repository (Asuncion & Newman, 2007) and have been widely used in credit scoring researches. The third is derived from 239 companies that were granted loans from the Industrial and Commercial Bank of China, a premier bank of China, between the year of 2006 and 2007. This dataset includes these companies' detailed financial records and corresponding credit scoring. Eighteen financial variables are chosen as the criteria for the credit scoring. These variables cover their financial structure, their ability of paying debt, the management's ability and the operations profitability, as listed in Table 1.

A summary of the characteristics of above three datasets is reported in Table 2. In the experiments, each dataset was divided into 80% training dataset and 20% testing dataset randomly.

3.2. Evaluation criteria

The evaluation criteria of our experiments are adopted from the established standard measures in the fields of credit scoring. These measures include average accuracy, type I error and type II error. Each measure has its merits and limitations. In this study, we prefer to use a combination of these measures, rather than a single measure, to measure the performance of three ensemble methods. The definition of these measures can be explained with respect to a confusion matrix as shown in Table 3.

Formally speaking, they are defined as follows:

$$\text{Average accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

$$\text{Type I error} = \frac{FN}{TP + FN} \quad (2)$$

$$\text{Type II error} = \frac{FP}{FP + TN} \quad (3)$$

Table 1
Financial variables for China credit scoring.

	Variable
Financial structure	Net asset/loan ratio Asset/liability ratio Net fix asset/fix asset book value ratio Long-term asset/shareholder equity ratio
Ability of paying debt	Current ratio Quick ratio Non-financing cash inflow/liquidity liability ratio Operating cash inflow/liquidity liability Interest coverage Contingent debt/net asset ratio
Management ability	Cash revenue/operating revenue ratio Account receivable turnover ratio Inventory turnover ratio Fix asset turnover ratio
Operation profitability	Gross profit margin Operating profit ratio Return on equity Return on assets

Table 2
The characteristics of three datasets used in the experiments.

	Total cases	Good/bad cases	No. of attributes
Australian credit dataset	690	307/382	14
China credit dataset	239	148/91	18
German credit dataset	1000	700/300	20

3.3. Experimental procedure

Ensemble methods are composed of several base learners. Based on the prior literature (Huang et al., 2007; Hung & Chen, 2009), we chose four widely used base learners, i.e., LRA, DT, ANN, and SVM, for our experiments.

In contrast to the linear relationships between the decision variable and the independent variables in linear regression models, LRA applies an additional logistic function and transform the linear probabilities into logit ones.

DT has been widely used in building classification models because it closely resemble human reasoning and is easy to understand. DT is a sequential model, which logically combines a sequence of simple tests. Each test compares a numeric attribute against a threshold value or a nominal attribute against a set of possible values. In this study, we chose widely used C4.5 for our experiments.

ANN is structured in layers, which generally consists of at least one input layer, one output layer, and a number of hidden layers existing in between. Each layer can have one or more nodes, and there are weights to connect the nodes in different layers. ANN has a number of variations in terms of possible algorithms. We chose the most commonly and widely used back-propagation networks for our experiments.

SVM is a state-of-the-art AI technique that has proven their performance in many applications, such as credit scoring, financial time series prediction and so on. The strength of this technique lies with its ability to model non-linearity, resulting in complex mathematical models.

Based on the above four base learners, Bagging, Boosting, and Stacking are implemented respectively. For Stacking, in order to keep diversity, all above four base learners are chosen as level-1 classifiers. As follow the literature, simple linear models usually

Table 3
Confusion matrix for credit scoring.

		Actual condition	
		Positive (Non-Risk)	Negative (Risk)
Test result	Positive(Non-Risk)	True Positive (TP)	False Positive (FP)
	Negative (Risk)	False Negative (FN)	True Negative (TN)

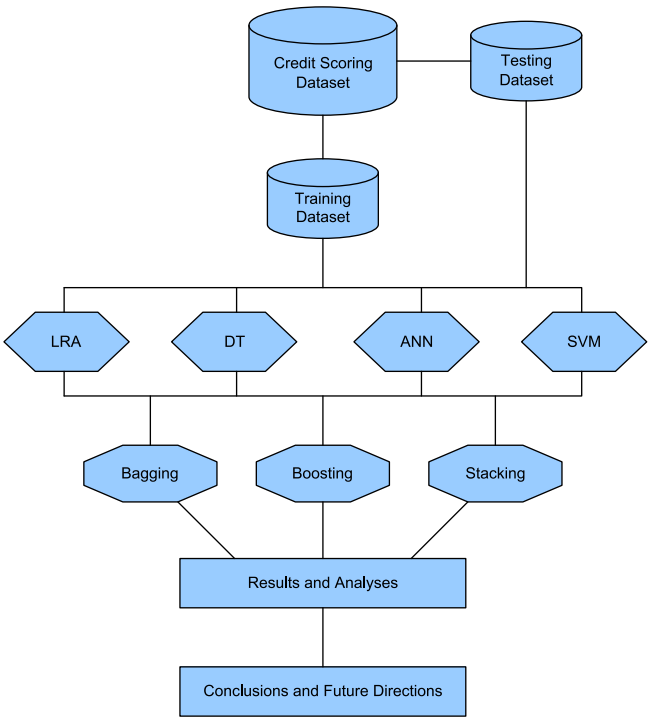

Fig. 4. Experimental procedure.

work well in level-2 classifier (Witten & Frank, 2005). In experiments, we chose LRA as level-2 classifier. Subsequently, experimental results are then evaluated and compared. Finally, the conclusions and the future directions are provided. The above experimental procedure is shown in Fig. 4.

4. Results and analyses

The experiments described in this section were performed on a PC with a 3.00 GHz Intel Core Duo CPU and 4 GB RAM, using Windows XP operating system. Data mining toolkit WEKA (Waikato Environment for Knowledge Analysis) version 3.6.0 was used for classification. WEKA is an open source toolkit, and it consists of a collection of machine learning algorithms for solving data mining problems (Witten & Frank, 2005).

For implementation of base learners, i.e., LRA, DT, ANN, and SVM, we chose logistic module, J48 (WEKA's own version of C4.5) module, multilayerperceptron module and SMO module in WEKA. And for implementation of ensemble learning, i.e., Bagging, Boosting, and Stacking, we chose the Bagging module, the AdaBoostM1 module and the Stacking module in WEKA. To minimize the influence of the variability of the training set, we repeated experiments 100 times. At each iteration, all algorithms were trained on the same training partition of the data and evaluated on the same test partition of the data. All the default parameters in WEKA were used.

Our goal in this empirical evaluation is to show that ensemble methods, which compete quite well against base learner, are plausible methods for credit scoring. Stronger statements can only be made after a more extensive empirical evaluation. Tables 4–7 summarize the performance indicators of base learners, Bagging, Boosting, and Stacking on the three credit datasets.

We first consider the results of the Australian credit dataset. The application of Bagging has brought a substantial improvement for DT and ANN: Bagging DT (86.30%, 14.59%, 12.99%) and Bagging ANN (85.01%, 16.13%, 14.08%) outperform base learners DT (84.39%, 18.00%, 13.70%) and ANN (83.28%, 19.27%, 14.68%) in term of all three performance indicators. Among four base learners, DT gets the biggest improvement (85.22%, 16.76%, 13.19%) after the application of Boosting. And the application of Stacking has also

Table 4
Performance obtained by the base learner.

		Australian credit dataset			China credit dataset			German credit dataset		
		Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)
LRA	Mean	86.56	12.68	14.05	72.07	18.49	43.23	76.14	11.70	52.23
	SD	2.51	4.16	4.14	5.40	7.10	10.39	2.31	2.64	5.20
DT	Mean	84.39	18.00	13.70	77.85	16.56	31.23	72.10	17.06	53.20
	SD	2.75	5.67	3.97	6.07	7.56	12.91	2.76	3.35	6.94
ANN	Mean	83.28	19.27	14.68	71.12	17.61	47.20	71.43	19.32	50.17
	SD	3.03	5.33	4.37	6.71	11.59	15.24	2.59	3.41	6.73
SVM	Mean	85.67	7.20	20.04	67.63	3.24	79.76	76.28	10.57	54.40
	SD	2.71	2.94	4.41	4.10	4.75	12.34	2.19	2.66	5.42

Table 5
Performance obtained by the Bagging.

		Australian credit dataset			China credit dataset			German credit dataset		
		Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)
Bagging (LRA)	Mean	86.64	12.46	14.08	74.18	16.45	41.03	76.08	11.64	52.57
	SD	2.42	4.19	4.17	6.41	7.46	11.72	2.37	2.63	5.44
Bagging (DT)	Mean	86.30	14.59	12.99	81.07	11.86	30.42	74.92	13.66	51.72
	SD	2.63	4.89	3.98	5.93	6.29	12.53	2.73	3.47	6.88
Bagging (ANN)	Mean	85.01	16.13	14.08	76.54	12.46	41.36	75.56	12.80	51.60
	SD	2.75	4.81	4.16	6.19	7.60	13.13	2.26	2.89	5.92
Bagging (SVM)	Mean	85.71	7.12	20.04	71.06	5.16	67.59	75.93	10.14	56.58
	SD	2.71	2.94	4.41	5.45	5.20	16.19	2.27	2.76	5.28

Table 6
Performance obtained by the Boosting.

		Australian credit dataset			China credit dataset			German credit dataset		
		Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)
Boosting (LRA)	Mean	86.56	12.68	14.05	72.12	18.46	43.18	76.14	11.70	52.23
	SD	2.51	4.16	4.14	5.37	7.01	10.46	2.31	2.64	5.20
Boosting (DT)	Mean	85.22	16.76	13.19	80.52	12.32	31.11	72.77	17.04	51.03
	SD	2.46	5.14	3.85	5.24	6.29	10.48	2.70	3.44	6.59
Boosting (ANN)	Mean	83.65	18.04	15.00	71.88	18.15	44.34	73.30	16.37	50.80
	SD	2.99	5.26	4.18	7.10	11.08	14.44	2.86	3.87	6.54
Boosting (SVM)	Mean	84.19	15.68	15.92	70.37	14.16	54.72	76.30	10.56	54.38
	SD	2.65	6.64	5.39	6.33	12.51	18.76	2.18	2.65	5.38

Table 7
Performance obtained by the Stacking.

	Australian credit dataset			China credit dataset			German credit dataset		
	Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)	Average accuracy (%)	Type I error (%)	Type II error (%)
Mean	86.57	12.91	13.85	78.60	13.50	34.20	75.97	10.76	55.00
SD	2.71	4.54	4.16	5.66	7.29	13.25	2.41	2.86	5.31

gotten the better performance (86.57%, 12.91%, 13.85%) than four base learners.

For the China credit dataset, the application of Bagging has also brought a substantial improvement for DT and ANN: Bagging DT (81.07%, 11.86%, 30.42%) and Bagging ANN (76.54%, 12.46%, 41.36%) outperform base learners DT (77.85%, 16.56%, 31.23%) and ANN (71.12%, 17.61%, 47.20%) in term of all three performance indicators. Among four base learners, DT gets the biggest improvement (80.52%, 12.32%, 31.11%) after the application of Boosting. And the application of Stacking has also brought improvement (78.60%, 13.50%, 34.20%) in terms of three performances indicators.

For the German credit dataset, the application of Bagging has also brought a substantial improvement for DT and ANN: Bagging DT (74.92%, 13.66%, 51.72%) and Bagging ANN (75.56%, 12.80%, 51.60%) outperform base learners DT (72.10%, 17.06%, 53.20%) and ANN (71.43%, 19.32%, 50.17%) in term of all three performance indicators. Among four base learners, DT (72.77%, 17.04%, 51.03%) and ANN (73.30%, 16.37%, 50.80%) get the bigger improvement after the application of Boosting. And the application of Stacking has also gotten the better performance (75.97%, 10.76%, 55.00%) than four base learners.

To further evaluate the influence of ensemble learning methods, we averaged the performance indicators across three credit datasets. Figs. 5–7 show the average performance indicators, i.e., average accuracy, type I error and type II error, on the three dataset.

As illustrated by the Figs. 5–7, we compare the credit scoring performance indicators of LRA, DT, ANN and SVM, with their Bagging, Boosting, and Stacking ensemble methods respectively. For the average accuracy, Bagging DT performs best (80.76%). Stacking (80.38%) performs slightly worse. Moreover, after the application

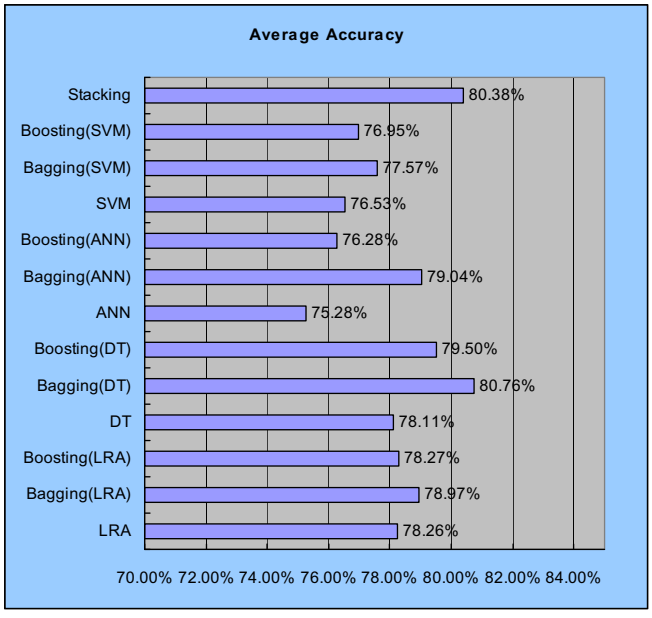

Fig. 5. Average results of average accuracy.

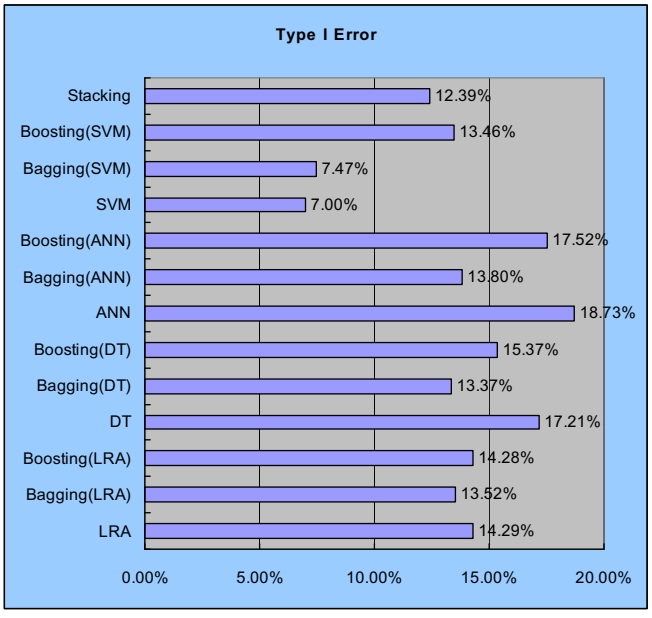

Fig. 6. Average results of type I error.

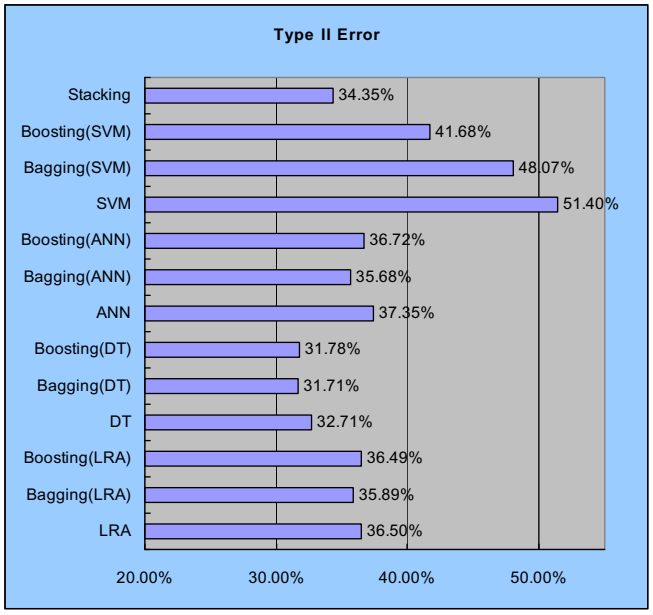

Fig. 7. Average results of type II error.

of ensemble learning methods, the performance of four base learners all has been enhanced. It is interesting to note that the application of Bagging has a bigger improvement than Boosting in terms of average accuracy.

For the type I error, the base learner, SVM, performs best (7.00%). Except SVM and Bagging SVM, Stacking (12.39%) gets better results. In addition, after the application of ensemble methods, the type I errors of three base learners, LRA, DT and ANN, all have been reduced. It is also interesting to note that the application of Bagging has a bigger improvement than Boosting in terms of type I error.

For the type II error, Bagging DT also performs the best (31.71%). Boosting DT (31.78%) performs slightly worse. And Stacking (34.35%) also gets better results. Moreover, after the application of ensemble methods, the type II errors all have been brought down. Note that except SVM, the remainder three base learners get better results after the application of Bagging than after the application of Boosting. For SVM, the Boosting SVM (41.68%) performs better than the Bagging SVM (48.07%).

Subsequently, in order to ensure that the assessment does not happen by chance, we tested the significance of these results by means of the non-parametric Wilcoxon matched-pairs signed-ranks test. The null hypothesis is 'Model A's mean of average accuracy/type I error/type II error = Model B's mean average accuracy/type I error/type II error'. The alternative hypothesis is 'Model A's mean average accuracy/type I error/type II error $\neq$ Model B's mean average accuracy/type I error/type II error'. The column 'improvement' gives the relative improvement in mean average accuracy (type I error or type II error) that Model B gives over Model A. The results are summarized in Table 8.

Note from this table that ensemble learning yields a significant improvement for the four base learners in terms of average accuracy indicator. For type I error, Bagging DT, Bagging ANN and Stacking get more than 20% improvement. For type II error, Boosting SVM and Stacking get more than 18% improvement. Like previous studies (Dietterich, 1997), the stability base learner, LRA, has not improved significantly in terms of three indicators. Especially for Boosting LRA, the statistical results show there is no difference between Boosting LRA and base learner, LRA. For SVM, with the improvements of average accuracy and type II error, type I error gets worse.

According to the above experimental results, we can draw the following conclusions:

(1) The application of ensemble learning has brought significant improvements as shown in Figs. 5–7 and Table 8. These results reveal that ensemble methods are dominated than individual base learner in credit scoring and are keeping with prior literatures.
(2) Bagging gets better results than Boosting except that Bagging SVM (48.07%) gets worse result than Boosting SVM (41.68%) in terms of type II error. The reason may be, just like Freund and Schapire (1996) suggested, that sometimes the poor performance of Boosting results from overfitting the training set since later training sets may be over-emphasizing examples that are noise. Thus Bagging is relative better choice for credit scoring when the noise is not removed.
(3) Among the three ensemble methods, Stacking (80.38%, 12.39%, 34.35%) and Bagging DT (80.76%, 13.37%, 31.71%) are two best methods. In addition, we tested the significance of Stacking and Bagging DT results by means of the non-parametric Wilcoxon matched-pairs signed-ranks test in terms of three indicators. The p-values are 0.392, 1.814E−5, 8.398E−9. The results reveal that the Stacking and Bagging have the same average accuracy while Stacking has the better type I error and Bagging DT has the better type II error.

Table 8
Outcomes of Wilcoxon matched-pairs signed-ranks test.

Model A	Model B	Average accuracy		Type I error		Type II error	
		Improvement (%)	p	Improvement (%)	p	Improvement (%)	p
LRA	Bagging (LRA)	0.91	4.264E−7**	5.39	2.022E−5**	1.68	0.051
	Boosting (LRA)	0.02	0.655	0.08	0.655	0.05	0.317
	Stacking	2.71	2.809E−9**	13.28	8.770E−9**	5.90	0.242
DT	Bagging (DT)	3.39	1.169E−23**	22.28	1.725E−30**	3.05	0.011*
	Boosting (DT)	1.78	5.817E−9**	10.64	1.297E−7**	2.84	0.009**
	Stacking	2.90	2.008E−26**	27.98	4.909E−37**	−5.02	6.395E−6**
ANN	Bagging (ANN)	5.00	9.260E−38**	26.35	1.164E−28**	4.47	0.059
	Boosting (ANN)	1.33	3.806E−11**	6.49	9.639E−11**	1.69	0.560
	Stacking	6.78	8.948E−41**	33.86	6.254E−30**	8.02	0.121
SVM	Bagging (SVM)	1.36	1.241E−5**	−6.68	0.061	6.47	3.933E−5**
	Boosting (SVM)	0.56	0.613	−92.23	5.702E−27**	18.92	2.758E−27**
	Stacking	5.03	1.355E−19**	−76.91	2.524E−34**	33.17	2.173E−34**

* p-values significant at $\alpha = 0.05$.
** p-values significant at $\alpha = 0.01$.

(4) From the base learner perspective, in this study we use four base learners, i.e., LRA, DT, ANN and SVM. For LRA, the application of Boosting has no influence for the performance indicators. For DT, the application of ensemble learning has brought a substantial improvement in all three credit datasets. For ANN, the application of ensemble learning gets better results except type II error. These findings further support ensemble learning work better when the base learner is unstable, e.g., DT and ANN. In our experiments we also found an interesting phenomenon that the application of ensemble learning has brought worse improvement for SVM in terms of type I error. The reason may be that the application of ensemble learning to SVM overfits the training dataset especial to good instances.

5. Conclusions and future directions

Ensemble learning is a powerful machine learning paradigm which has exhibited apparent advantages in many applications. In this study, a comparative assessment of three popular ensemble methods, i.e. Bagging, Boosting, and Stacking, based on four base learners, i.e., LRA, DT, ANN and SVM, is carried out. All these ensemble methods have been applied to three real world credit datasets, i.e. Australian and German credit datasets, which are from UCI machine learning repository, and China credit dataset, which is from Industrial and Commercial Bank of China. Experimental results reveal that the application of ensemble learning has been brought substantial improvement for individual base learner. Especially in our experiments, Bagging performs better than Boosting. In addition, Stacking and Bagging DT get best results in terms of three performance indicators, i.e., average accuracy, type I error and type II error. And among four base learners, DT gets best improvement in terms of three performance indicators after the application of ensemble learning.

Several future research directions also emerge. Firstly, large datasets for experiments and applications, particularly with more exploration of credit scoring data structures, should be collected to further valid the conclusions of this study. Secondly, further analyses are encouraged to explore the reasons why the application of ensemble learning has worsened the performance of SVM in terms of type I error. Thirdly, a major limitation of ensemble learning methods is the lack of interpretability of the results, i.e., the knowledge learned by ensembles is difficult to understand by humans. Therefore improving the interpretability of ensembles is another important yet largely understudied research direction.

Acknowledgements

The authors would like to thank the Editor-in-Chief and reviewers for their recommendation and comments. This work is partially supported by the grants from the Innovation and Technology Fund (ITF) of HK (GHP/006/07, InP/007/08).

References

Asuncion, A. & Newman, D. J. (2007). *UCI machine learning repository*. Irvine, CA: University of California, School of Information and Computer Science. <http://www.ics.uci.edu/~mlearn/MLRepository.html>

Baesens, B., van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society, 54*(6), 1082–1088.

Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*(2), 123–140.

Buta, P. (1994). Mining for financial knowledge with CBR. *AI Expert, 9*(2), 34–41.

Dasarathy, B. V., & Sheela, B. V. (1979). Composite classifier system design: Concepts and methodology. *Proceedings of the IEEE, 67*(5), 708–713.

Desai, V., Crook, J., & Overstreet, G. (1996). A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operations Research, 95*(1), 24–37.

Dietterich, T. G. (1997). Machine learning research: Four current directions. *AI Magazine, 18*(4), 97–136.

Freund, Y., & Schapire, R. (1996). Experiments with a new boosting algorithm. In *Proceedings of the thirteenth international conference on machine learning, Bari, Italy* (pp. 148–156).

Friedman, J. H. (1991). Multivariate adaptive regression splines. *The Annals of Statistics, 19*(1), 1–141.

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

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 12*(10), 993–1001.

Huang, Z., Chen, H., Hsu, C. J., Chen, W. H., & Wu, S. S. (2004). Credit rating analysis with support vector machines and neural networks: A market comparative study. *Decision Support Systems, 37*(4), 543–558.

Huang, C. L., Chen, M. C., & Wang, C. J. (2007). Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications, 33*(4), 847–856.

Hung, C., & Chen, J. H. (2009). A selective ensemble based on expected probabilities for bankruptcy prediction. *Expert Systems with Applications, 36*(3), 5297–5303.

Karels, G., & Prakash, A. (1987). Multivariate normality and forecasting of business bankruptcy. *Journal of Business Finance Accounting, 14*(4), 573–593.

Makowski, P. (1985). Credit scoring branches out. *Credit World, 74*(2), 30–37.

Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine, 6*(3), 21–45.

Reichert, A. K., Cho, C. C., & Wagner, G. M. (1983). An examination of the conceptual issues involved in developing credit-scoring models. *Journal of Business and Economic Statistics, 1*(2), 101–114.

Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning, 5*(2), 197–227.

Schebesch, K. B., & Stecking, R. (2005). Support vector machines for classifying and describing credit applicants: Detecting typical and critical regions. *Journal of the Operational Research Society, 56*(9), 1082–1088.

Shin, K. S., & Han, I. (2001). A case-based approach using inductive indexing for corporate bond rating. *Decision Support Systems, 32*(1), 41–52.

Thomas, L. C. (2000). A survey of credit and behavioral scoring: Forecasting financial risks of lending to customers. *International Journal of Forecasting, 16*(2), 149–172.

West, D. (2000). Neural network credit scoring models. *Computers and Operations Research, 27*(11-12), 1131–1152.

Windeatt, T., & Ardeshir, G. (2004). Decision tree simplification for classifier ensembles. *International Journal of Pattern Recognition, 18*(5), 749–776.

Witten, I. H., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques*. Boston: Morgan Kaufmann Publishers.

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks, 5*(2), 241–259.

Yu, L. A., Wang, S. Y., & Lai, K. K. (2008). Credit risk assessment with a multistage neural network ensemble learning approach. *Expert Systems with Applications, 34*(2), 1434–1444.

Zhou, Z. H. (2009). Ensemble. In L. Liu & T. Özsu (Eds.), *Encyclopedia of database systems*. Berlin: Springer.