A Novel Credit Risk Assessment Model Based on LightGBM

Jinshu Gu

Aerospace Research Institute of Materials & Processing Technology, Beijing, China

Abstract: Credit risk assessment is a critical process in the bank's approval of loans and is of great significance in the bank's risk management. With the development of big data and artificial intelligence technology, it is an important research direction to comprehensively evaluate enterprise or individual credit based on multidimensional data. In this paper, we apply a new LightGBM model based on the decision tree algorithm promotion framework to implement credit risk assessment, which is a typical binary classification problem. The results show that compared with the classical SVM model, LightGBM could achieve higher prediction accuracy, so it is an accurate and effective credit risk assessment method.

Keywords: credit risk assessment; LightGBM; decision tree; classification

1. Introduction

With the development of Internet finance and information technology, as a classic and critical issue in the financial field, credit risk assessment has attracted great attention from academic researchers and financial institutions. The main task of credit risk assessment is to establish a model that could distinguish good creditors from bad creditors [1]. In simple terms, the goal is to take the credit data set of the past, consisting mainly of multidimensional data of individuals, and use them to learn the rules that are generally applicable in the future to distinguish between two creditor human beings, with as few false positives and false negatives as possible [2]. Good credit classification is not only conducive to financial institutions and credit enterprises to effectively control risks and increase profits, but also conducive to the long-term healthy development of relevant enterprises and the national economy.

Over the past few decades, many methods have been introduced to evaluate credit risk [3]. For example, discriminant analysis and mathematical programming have been widely used in credit classification research. However, due to the nonlinear relationship between the default probability and the characteristics of credit customers, these hard computing techniques may not achieve good classification results in credit classification tasks. Because of this, some emerging soft computing and machine learning models such as Nearest Neighbor Algorithm, Artificial Neural Network, Evolutionary Algorithm, and Support Vector Machine have been applied to credit classification tasks and achieved good classification results [4].

In recent years, a new method based on the decision tree promotion framework has been proposed by researchers, which can solve the classification problem well [5]. To further explore and improve the accuracy of the credit classification model, we apply this method to some different credit datasets and compare it with traditional machine learning methods, hoping to get some enlightenment.

2. Methodology

2.1. Light Gradient Boosting Machine

Light Gradient Boosting Machine (LightGBM) is a novel GBDT (Gradient Boosting Decision Tree) algorithm proposed in 2017, which has been widely used in several typical machine learning domains, such as energy, finance, and biology [5]. The LightGBM algorithm contains two novel steps, which are the gradient-based one-side sampling and the exclusive feature bundling, respectively. Therefore, LightGBM could effectively deal with a large number of data instances and features simultaneously.

Given the supervised training set $X = \{(x_i, y_i)\}_{i=1}^{n}$, LightGBM aims to find an approximation $\hat{F}(x)$ to a certain function $F^*(x)$ that minimizes the expected value of a specific loss function $L(y, F(x))$ as follows:

$$\hat{F} = \arg\min_F E_{y,X} L(y, F(x)) \tag{1}$$

LightGBM employs several T regression trees $\sum_{t=1}^{T} f_t(X)$ to approximate the final model, which is

$$F_T(X) = \sum_{t=1}^{T} f_t(X) \tag{2}$$

Different from the traditional GBDT based techniques, such as XGBoost and GBDT, the core issue for LightGBM is how to search for the optimal tree structure. LightGBM is the tree learning method and splitting point selection criterion. Most decision tree algorithms employ a level-wise tree learning method. In a level-wise tree learning algorithm, one feature is selected and placed at the root node, and this attribute is split based on several criteria (e.g., information gain or Gini index). Then, training samples are split into subsets (one for each branch that extends from the root node). Third, this step

© ACADEMIC PUBLISHING HOUSE

is repeated for a selected branch. However, LightGBM grows trees following a leaf-wise (or best-first) method. This leaf-wise approach expands nodes in a best-first order instead of a fixed, level-wise order [5].

Enumerating all possible tree structures is costly. Thus, in LightGBM, a histogram-based approximation algorithm is used for selecting candidate splitting points when the dataset is comparatively large. Histogram algorithm is to discretize continuous floating-point eigenvalues into K integers and construct a histogram with width k. After traversing the data once, the histogram accumulates the required statistics. Then, according to the discrete value of the histogram, traverses to find the optimal segmentation point. The histogram only needs to calculate the information gain of the histogram statistics, which is much smaller than that of the presort algorithm which iterates through all the values each time. Besides, the histogram of the leaf node is obtained by using the subtraction of the histogram of the parent node and the adjacent node, to reduce the number of times of histogram construction and improve the efficiency. Finally, the memory used to store histogram statistics is much smaller than that of the presort algorithm.

2.2. Support Vector Machine

In recent years, the SVM method has been widely used in several different fields, due to its good generalization performance and strong theoretical foundations [6]. In general, SVM could adopt the principle of structural risk minimization (SRM), which could avoid the "dimension disaster" and has great generalization ability. The main objective of SVM is to estimate a relationship between input and output random variables under the assumption that the joint distribution of the variables is completely unknown.

The implementation of SVM model can be summarized by the following steps: (1) divide the training set and testing set; (2) choose the appropriate kernel function (Linear, Gauss, Polynomial, and Sigmoid); (3) select a hyper-parameter optimization method (Grid Search, Evolutionary Strategy, Particle Swarm Optimization, and Simulated Annealing); (4) model training and testing.

The SVM model could be formalized as a problem of inferring a function $y = f(x)$ based on the training data $X = \{(x_i, d_i), i = 1,2, \dots, m\}$, where $x_i \in R^n$is the i^{th} input vector for the i^{th} training data, $d_i \in R$ is the target value for the i^{th} training data and m is the number of training data. Furthermore, learning an SVM is equivalent to finding a regression function of the form:

$$f(x) = \sum_{i=1}^{m}(\alpha_i - \alpha_i^*)k(x_i, x) + b \tag{3}$$

Where $k(x_i, x)$ is a positive definite kernel function,$\alpha = (\alpha_1, \alpha_2, \dots, \alpha_m)^T$, $\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_m^*)^T$and b are parameters of the model.

In this paper, the Gauss kernel function which is a nonlinear kernel would be used. Its form is given by the following equation:

$$k(x_i, x) = \exp(-\frac{\|x_i - x\|^2}{\sigma^2}) \tag{4}$$

where $\sigma > 0$ is the width of the kernel.

The essence of SVM is to find an n-1 hyperplane with the best tolerance in the n-dimensional space with x samples to separate the two kinds of samples and make the distance from any sample to the hyperplane greater than or equal to 1. The establishment of hyperplane is only affected by the vectors on the decision boundary. When the SVM encounters linearly indivisible samples, it will project the samples to a higher dimensional space to make the samples linearly separable.

3. Experiment Results

3.1. Data Description

In this paper, the Australian credit dataset (ACD) and German credit dataset (GCD) from UCI Machine Learning Repository are used, which have been utilized by many scholars to implement credit risk assessment. In general, ACD has 690 total instances, which is composed of 383 good instances and 307 bad instances, and the number of attributes of ACD is 14. GCD has 1000 total instances, which is composed of 700 good instances and 300 bad instances, and the number of attributes of GCD is 24. Both of which are less enough than the number of instances. Besides, we also utilize another real-life credit dataset of a US commercial bank (AMCD) to compare the classification accuracy of different models. AMCD has 5000 total instances, which is composed of 4185 good instances and 815 bad instances, which means that AMCD is a typical unbalanced dataset and more like the actual situation. The structure of these three datasets is described in Table 1.

Table 1. Information of three credit datasets

	ACD	GCD	AMCD
Total Instances	690	1000	5000
Good Instances	383	700	4185
Bad Instances	307	300	815
Attributes	14	24	65
Classes	2	2	2

For model training, all the datasets are randomly partitioned into training sets and independent test sets, and the corresponding proportion is 80% and 20%. To make the evaluation more credible, the dataset division processes are repeated ten times and compute the average prediction accuracy of different models. Moreover, the min-max normalization processes are implemented on both datasets to eliminate the impact of data magnitude by transferring all the data to [0, 1].

3.2. Hyper-parameter Optimization

In this paper, we compare the classification accuracy between LightGBM and the traditional SVM model. In general, the prediction accuracy of machine learning methods would be significantly influenced by hyper-

© ACADEMIC PUBLISHING HOUSE

parameter. Therefore, we first determine the number and the range of variation of the hyper-parameters for both models.

Then, 5-fold cross-validation is used in our experiments when training models for choosing the hyper-parameters. For LightGBM, Grid Search method is implemented to conduct a parametric space search process, through which the optimal combination of different hyper-parameters could be obtained. As a result, the number of leaves is 80, and the feature fraction is 0.5. Moreover, we use the default values for the other hyper-parameters. For SVM, the Grid Search method is implemented and the optimal combination of different hyper-parameters could be obtained. According to the results, the kernel function is Gaussian, the penalty factor is 128, and the kernel function coefficient is 64.

Finally, Python 3.6 is used to implement all the experiments.

3.3. Classification Accuracy

For this typical classification task, the performance is measured by Type 1 accuracy (T1), Type 2 accuracy (T2) and Total accuracy (T), which stand for the percent of correctly classified good samples, the percent of correctly classified bad samples and the percent of correctly classified in total, respectively. After the training and testing process, the evaluation results of the two models are obtained. The average prediction accuracy of different models is shown in Table 2.

Table 2. The average classification accuracy of different models

ACD			
	T1	T2	T
LightGBM	90.1%	89.67%	89.89%
SVM	73.44%	77.87%	75.03%
GCD			
	T1	T2	T
LightGBM	92.5%	88.76%	90.63%
SVM	66.22%	62.67%	64.99%
AMCD			
	T1	T2	T
LightGBM	85.26%	91.2%	88.23%
SVM	60.33%	71.15%	65.00%

According to the results, we could find that the LightGBM model is superior to the SVM model under all the datasets. In the set of ACD, the overall accuracy is improved by more than 14%. In the set of GCD, the overall accuracy is improved by more than 25%. In the set of AMCD, the overall accuracy is improved by more than 23%. Besides, the results show that each method has different precision for type I and type II tasks, while LightGBM performs well in both categories.

4. Conclusion

In this study, we apply a new LightGBM model based on the decision tree algorithm promotion framework to implement credit risk assessment. Through the experiment results, the following major conclusions could be obtained. Firstly, LightGBM could obtain relatively high prediction accuracy, which is a very effective method to make a credit risk assessment. Secondly, although the credit risk assessment model based on the LightGBM algorithm has some improvement in the effect of classification prediction, there is still room for further improvement.

In the future, we will choose more real datasets to further verify the effectiveness of LightGBM, and we will also try to use the ensemble learning framework to make the classification. The optimization of hyper-parameters is also an important research direction.

References

[1] Arya, S., et al. Anatomy of the Credit Score. *Journal of Economic Behavior & Organization*, **2013**, 95: 175-185.

[2] Berger, S.C., and Gleisner, F. Emergence of Financial Intermediaries in Electronic Markets: The Case of Online P2P Lending. *Business Research*, **2009**, 2: 39-65.

[3] Marqués, A.I., et al. A literature review on the application of evolutionary computing to credit scoring. *Journal of the Operational Research Society*, **2013**, 64: 1384-1399.

[4] Xia, Y., et al. A Rejection Inference Technique Based on Contrastive Pessimistic Likelihood Estimation for P2P Lending. *Electronic Commerce Research and Applications*, **2018**: 111-124.

[5] Ke, G., et al. LightGBM: a highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, **2017**: 3146-3154.

[6] Suykens, J.A.K., and Vandewalle, J. Least Squares Support Vector Machine Classifiers. *Neural Processing Letters*, **1999**: 293-300.

© ACADEMIC PUBLISHING HOUSE