Application of Random Forest Classifier in Loan Default Forecast

Huannan Zhang[1](✉), Yilin Bi[1], Wangdong Jiang[1], Chuntian Luo[1], Shengjia Cao[1], Peng Guo[1,2], and Jianjun Zhang[3]

[1] Hunan University of Finance and Economics, Changsha 410205, China
1136345443@qq.com
[2] University Malaysia Sabah, Kota Kinabalu, Malaysia
[3] Hunan Normal University, Changsha, China

Abstract. Calculating the possible default risk of borrowers before issuing loans is the cornerstone of risk management of financial institutions and the basis of industry development. This study uses the idea of non-equilibrium data classification to statistically analyze the loan data provided by Kaggle, and then uses Sklearn-ensemble-Random Forest Classifier in Python to establish a random forest model for loan default forecast. The experimental results show that the random forest algorithm exceeds the decision tree and logistic regression classification algorithm in predicting performance on this data set. By using random forest algorithm to sort the importance of features, we can calculate the important characteristics that affect the default, and provide an important basis for the judgment of lending risk in the financial field.

Keywords: Risk management · Random forest algorithm · Loan default forecast · Big data analysis

1 Introduction

Loans are an important way for companies and individuals to solve the problem of capital operation. It is this demand that the bank's various loan businesses are targeting [1]. The good operation of this mechanism must prevent loan defaults and calculate the possible default risk of borrowers before issuing loans. It is the cornerstone of risk management of financial institutions and the basis of industry development [2].

Based on the idea of non-equilibrium data classification, this study statistically analyzes the loan data provided by Kaggle, and then uses Sklearn-ensemble-Random Forest Classifier in Python to establish a random forest model for loan default forecast. The experimental results show that the random forest algorithm exceeds the decision tree and logistic regression classification algorithm in predicting performance on this data set. By using random forest algorithm to sort the importance of features, we can calculate the important characteristics that affect the default, and provide an important basis for the judgment of lending risk in the financial field [3]. The first section of this paper mainly introduces unbalanced data classification and random forest algorithm; the second section mainly performs data preprocessing and data analysis. The third section

X. Sun et al. (Eds.): ICAIS 2020, CCIS 1254, pp. 410–420, 2020.
https://doi.org/10.1007/978-981-15-8101-4_37

mainly constructs a random forest classification model for predicting loan defaults, and obtains the AUC value of the evaluation results of the model. By comparing the random forest algorithm with the decision tree and the logistic regression algorithm model, the conclusion that the random forest algorithm is better is obtained. Finally, by evaluating the importance of each feature, it is concluded which features have a greater impact on the outcome of the eventual default. The fourth section summarizes the full text.

2 Random Forest Classifier

2.1 Unbalanced Data Classification

Unbalanced data refers to one type (majority) of data far exceeds another type(minority), and is common in many fields such as network intrusion detection, financial fraud transaction detection, text classification, and the like. In many cases, we are only interested in the classification of a few classes [4]. The classification problem of dealing with unbalanced data can be solved by the penalty weight of positive and negative samples [5]. The idea is that in the process of algorithm implementation, different weights are assigned to the categories of different sample sizes in the classification. Generally, the small sample size has high weight and large sample. The quantity category has a low weight and is then calculated and modeled [6].

2.2 Introduction to Random Forest

Random forest belongs to the Bagging (short for Bootstrap AGgregation) method in integrated learning [7]. Random forests are made up of many decision trees, and there is no correlation between different decision trees. When we perform the classification task, the new input sample enters, and each decision tree in the forest is judged and classified separately. Each decision tree will get its own classification result, and which classification result of the decision tree Most, then random forest will use this result as the final result [8]. The process is shown in Fig. 1.

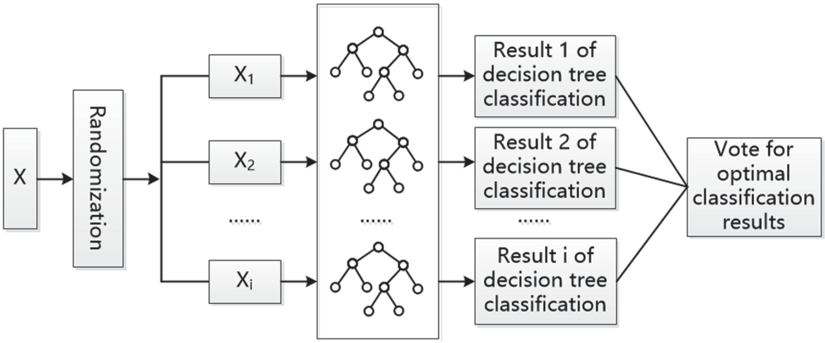
X
Randomization
X1
X2
......
Xi
......
Result 1 of decision tree classification
Result 2 of decision tree classification
......
Result i of decision tree classification
Vote for optimal classification results

Fig. 1. Schematic diagram of random forest

2.3 Principles and Characteristics of Random Forest Algorithms

The Random Forest algorithm, which includes classification and regression problems, if there are N samples, there are N samples randomly selected (each time randomly selects one sample and then returns to continue selection). This selected N samples are used to train a decision tree as a sample at the root of the decision tree [9]. When each sample has M attributes, when each node of the decision tree needs to be split, m attributes are randomly selected from the M attributes, satisfying the condition m << M. Then use some strategy (such as information gain) from these m attributes to select 1 attribute as the split attribute of the node [10].

During the decision tree formation process, each node must be split according to the steps until it can no longer split. Note that no pruning is done during the formation of the entire decision tree [11]. Follow the steps to build a large number of decision trees, which constitutes a random forest. The algorithm steps are as follows (Table 1):

Table 1. Random forest algorithm

```
Random forest algorithm
Input：
T= Training set,
$N_{tree}$= The number of decision tree in forest,
M= Number of predictors in each sample,
$M_{try}$= The number of variables participating in the partition in each tree node,
$S_{sampsize}$= Size of Bootstrap samples
The process of Algorithmic：
for($i_{tree}$=0;1＜$i_{tree}$≤$N_{tree}$; $i_{tree}$++)
{
1.   Generate a Bootstrap data sample using the training set T, the size is $S_{sampsize}$
2.   Construct an untrimmed tree $i_{tree}$ using the generated Bootstrap data. In the
process of generating a tree $i_{tree}$, $M_{try}$ variables are randomly selected from M and the
best one is selected according to a certain standard (Gini value) for branching.
}
Output：
The problem of Regression: The average of the return values of all individual
numbers is used as the forecast result.
The problem of Classification: The classification results of most decision trees are
used as forecast results.
```

It can be seen from the above algorithm process that the randomness of the data space is implemented by Bagging (Bootstrap Aggregating), and the randomness of the feature space is implemented by a Random Subspace [12]. For the classification problem, each decision tree in the random forest classifies and predicts new samples, and then somehow aggregates the decision results of these trees to give the final classification results of the samples.

1. The introduction of two random factors in rows and columns in the data makes it difficult for random forests to fall into overfitting [13].
2. Random forests have good anti-noise ability [14].
3. When there are a large number of missing values in the data set, the random forest can effectively estimate and process the missing values [15].
4. Strong adaptability to the data set: can handle both discrete data and continuous data, the data set does not need to be standardized [16].
5. Can be ordered to the importance of variables, to facilitate the interpretation of variables [17]. There are two ways to calculate the importance of variables in random forests: one is based on the average drop accuracy of OOB (Out of Bag). That is, in the process of growing the decision tree, the OOB sample is first used to test and record the sample of the fault, and then the value of a column of the Bootstrap sample is randomly disordered, and the decision tree is used to predict and re-record. The number of wrong samples are recorded[18]. The number of two forecast errors divided by the total number of OOB samples is the error rate change of this decision tree. The average rate of error reduction is obtained by averaging the error rate changes of all trees in the random forest [19]. The other is based on the GINI drop method at the time of splitting. The random forest in the growth decision tree is splitting according to the decline of GINI impureness, and all the nodes in the forest that select a variable as a split variable are summarized. The amount of GINI dropped [20].

2.4 Random Forest Method for Unbalanced Data Classification

The random forest algorithm defaults to a weight of 1 for each class, which is to assume that the misclassification costs of all classes are equivalent. In scikit-learn, the random forest algorithm provides a class_weight parameter whose value can be a list or dict value, manually indicating the weight of different categories. If the parameter is "balanced", the random forest algorithm automatically adjusts the weight using the y value, and the various weights are inversely proportional to the class frequency in the input data.

The calculation formula is:

$$\frac{n_{samples}}{n_{classes} * np.bincount(y)} \tag{1}$$

"balanced_subsample" is similar to the "balanced" mode, which uses the number of samples in a sample with a return type instead of the total number of samples. Therefore, we can solve the problem of unbalanced data classification by this method.

3 Data Preprocessing and Data Analysis

The random forest is an algorithm based on the idea of integrated learning, which integrates multiple trees. Its basic unit is the decision tree, and these decision trees are independent of each other. The random forest contains the following ideas:

(1) Random selection of data samples
(2) Construction of decision tree
(3) Random selection of candidate features
(4) Forest forecast strategy

3.1 Data Set

The loan default data set used in this article is from the Kaggle data science competition platform. The data set is named "Give Me Some Credit". The data set contains 25000 samples, of which 150,000 samples are used as training sets and 100,000 samples are used as test sets.

The training set has a total of 150,000 borrowers' historical data, including 10026 default samples, accounting for 6.684% of the total sample, loan default rate of 6.684%, and 139,974 non-default samples, accounting for 93.316% of the total sample. It can be seen that the data set is a typical highly unbalanced data. The data set includes the borrower's age, income, family, etc. and the loan situation, a total of 11 variables, of which SeriousDlqin2yrs is the label tag, and the other 10 variables are predictive features. The following table lists the variable names and data types (Table 2):

3.2 Data Preprocessing

A preliminary exploration of the data reveals that there are missing values in the two variables, Monthly Income and Number of Dependents, which are 29731 and 3924 respectively.

The outliers include: The minimum value in the age variable is 0, which is an outlier.

Among the three variables NumberOfTime30-59DaysPastDueNotWorse, Number OfTime30-59DaysPastDueNotWorse, NumberOfTimes90DaysLate, there are a few values of 96,98, which may be abnormal values or a certain behavior code.

Data preprocessing: When we use the pandas library in Python to read data, set the na_values parameter in the function pd.read_csv() to list, and treat the 0 in the age variable and 96,98 in the three overdue variables as NaN. Value, then use the sklearn-preprocessing-Imputer library to replace all NaN in the dataset with the average of the corresponding columns.

3.3 Data Analysis

The experimental environment used in the experiment was Anaconda3+Python3.8. First, an exploratory analysis of the data is performed to analyze the distribution of the default rate on each independent variable, and a frequency distribution table as shown in Table 3

It can be seen from Table 3 that the population below 25 years old and the population aged 26–35 years have a default rate of more than 10%. As the age increases, the default rate decreases.

It can be seen from Table 4 that the number of real estate and mortgage loans of 99.47% of borrowers is less than 5, and the default rate of borrowers with more than 5 credits has increased significantly, and the default rate of borrowers exceeding 10 is over 20%.

It can be seen from Table 5 that the default rate of borrowers who have not exceeded 30–59 days is only about 4%. As the number of overdue increases, the default rate increases significantly. The other two variables, the frequency distribution table of the borrower's 60–89 days overdue and the borrower's overdue frequency of 90 and above

Table 2. Data set variables

Variable name	Description of variable	Genre
SeriousDlqin2yrs	Whether to default	Y/N
Revolving Utilization Of Unsecured Lines	The total amount of credit card and personal credit loan (excluding mortgages, installment payments like car loans, etc.) divided by the sum of credit lines	Percentage
Age	Borrower's age	Integer
NumberOfTime30-59DaysPastDueNotWorse	The number of times the borrower has overdue 30–59 days in the past two years	Integer
Debt Ratio	Monthly debt payments	Percentage
Monthly Income	Monthly income	Real number
Number Of Open Credit Lines And Loans	The number of Open loans and Lines of credit	Integer
NumberOfTimes90DaysLate	The number of times the borrower has overdue 90 days or more in the past two years	Integer
Number Real Estate Loans Or Lines	Number of mortgage and real estate loans including housing mortgage credit loans	Integer
NumberOfTime60-89DaysPastDueNotWorse	The number of times the borrower has overdue 60–89 days in the past two years	Integer
Number Of Dependents	Number of people (spouses, children, etc.) who need to be supported in the family, excluding themselves	Integer

also showed the same trend as Table 5. Therefore, it can be seen that the more times the borrower has overdue, the higher the default rate.

The "Give Me Some Cerdit" dataset has 10 variables, statistical analysis of each variable and the frequency distribution table shown above, except that the variable NumberOfOpenCreditLinesAndLoans (the number of open loans and credit loans) has no

Table 3. Frequency distribution table of variable age

Age	Number	Proportion	Number of defaulters	The percentage of default in this interval
<25	3028	2.02%	338	11.16%
26–35	18458	12.3%	2053	11.12%
36–45	29819	19.9%	2628	8.8%
46–55	36690	24.5%	2786	7.6%
56–65	33406	22.3%	1531	4.6%
>65	28599	19.1%	690	2.4%

Table 4. Frequency distribution table of the variable number real estate loans or lines

Number Real Estate Loans Or Lines	Number	Proportion	Number of defaulters	The percentage of default in this interval
<5	149207	99.47%	9884	6.6%
6–10	699	0.47%	121	17.3%
11–15	70	0.05%	16	22.8%
16–20	14	0.009%	3	21.4%
>20	10	0.007%	2	20%

Table 5. Frequency distribution table of the variable Number Of Time 30-59Days Past Due Not Worse

Number Of Time 30-59 Days Past Due Not Worse	Number	Proportion	Number of defaulters	The percentage of default in this interval
0	126018	84%	5041	4%
1	16032	10.7%	2409	15%
2	4598	3.1%	1219	26.5%
3	1754	1.2%	618	35.2%
4	747	0.5%	318	42.6%
5	342	0.23%	154	45%
6	140	0.09%	74	52.9%
≥7	104	0.07%	50	48.07%

significant correlation with the default rate. Other variables are related to whether the borrower ultimately defaults.

4 Modeling and Experimental Results

4.1 Random Forest Model

This experiment uses the sklearn-ensemble-Random Forest Classifier in Python to build a random forest model.

The parameter is set to:

N_estimators: The number of decision trees is set to 100.
Oob_score: Whether to use out-of-bag data, set to True,
Min_samples_split: When dividing nodes according to attributes, the number of samples per partition is set to 2,
Min_samples_leaf: The minimum number of samples with leaf nodes, set to 50,
N_jobs: Parallel number, set to −1 how many cores the computer CPU has, how many jobs are started
Class_weight: set to 'balanced_subsample', using y value to automatically adjust the weight, the various weights are inversely proportional to the category frequency in the input data.
Bootstrap: Whether to use the bootstrap sample sample, set to True.

4.2 Model Evaluation

The model evaluation index used in this experiment is the AUC (Area under the ROC curve) value. AUC is defined as the area under the ROC (Receiver Operating Characteristic) curve [21]. The horizontal axis of the ROC curve is False Positive Rate (FPR), the vertical axis is True Positive Rate (TPR), and since the ROC curve is generally above the line y = x, AUC The value ranges between 0.5 and 1 [22]. The AUC value is used as the evaluation criterion because many times the ROC curve does not clearly indicate which classifier works better, and as a numerical value, the classifier corresponding to the larger AUC is better.

The random forest model is compared with the logistic regression classification model and the decision tree classification model. The results are shown in the following table (Table 6).

Table 6. Comparison of random forests and other algorithms

Algorithms	AUC value
Random forest	0.86
Decision tree	0.80
Logistic regression	0.80

It can be seen from the table that the random forest algorithm has higher AUC values than the decision tree and the logistic regression algorithm, so the algorithm forecast performance of the random forest is better than the other two algorithms.

4.3 Feature Importance Metrics

This experiment uses the feature_ importance_ method of sklearn-ensemble-Random Forest Classifier to get the importance of each feature as shown in the following table (Table 7).

Table 7. Variable importance

Variables	feature_ importance
Revolving Utilization Of Unsecured Lines	0.3411
NumberOfTime30-59DaysPastDueNotWorse	0.1694
NumberOfTimes90DaysLate	0.1594
NumberOfTime60-89DaysPastDueNotWorse	0.0727
age	0.0677
Debt Ratio	0.0625
Monthly Income	0.0488
Number Of Open Credit Lines And Loans	0.0442
Number Real Estate Loans Or Lines	0.0223
Number Of Dependents	0.0117

As can be seen from the above table, the three characteristics of the borrower's total loan-to-credit ratio, the number of overdue 30–59 days in the past two years and the number of overdue over 90 days in the past two years are in the top three. There is a greater impact on whether the default is ultimately breached, so you can pay special attention to these characteristics of the borrower when processing the loan application.

5 Summary

This paper studies the random forest algorithm to predict loan defaults in the financial sector, using unbalanced data classification. In the process of constructing a single tree, randomly select some variables or features to participate in the tree node division, repeat multiple times and ensure the independence between the established trees. For the unbalanced data, the random forest method can be based on the parameter adjustment. The value automatically adjusts the weight to effectively solve the classification problem of unbalanced data. Experiments show that the random forest algorithm has better classification performance than the decision tree and logistic regression model, and has important reference significance for the loan default forecast problem in the financial field. In addition, by measuring the importance of each feature, in this experiment, the three characteristics of the borrower's age, debt ratio, and the number of real estate and mortgage loans can be greatly affected. It also has important reference significance for feature selection in other data mining.

This paper mainly studies the loan default forecast from the perspective of random forest algorithm, and adopts the parameter adjustment method to solve the data non-equilibrium problem in data processing. However, it still needs to be improved in data processing and model optimization. There are still many futures jobs. First, explore more and more efficient unbalanced data processing methods and optimize data in data processing. Secondly, in the algorithm selection, learning from other algorithm models, try to combine the optimization model to improve performance. Finally, in terms of rendering, try to use visualizations to present the results in a chart that is easier to understand.

Acknowledgments. This research work is implemented at the 2011 Collaborative Innovation Center for Development and Utilization of Finance and Economics Big Data Property, Universities of Hunan Province; Hunan Provincial Key Laboratory of Big Data Science and Technology, Finance and Economics; Key Laboratory of Information Technology and Security, Hunan Provincial Higher Education. This research is funded by the Open Foundation for the University Innovation Platform in the Hunan Province grant number 18K103; Open project, grant number 20181901CRP03, 20181901CRP04, 20181901CRP05; Hunan Provincial Education Science 13th Five-Year Plan (Grant No. XJK016BXX001), Social Science Foundation of Hunan Province (Grant No. 17YBA049).

References

1. Torvekar, N., Game, P.S.: Predictive analysis of credit score for credit card defaulters. Int. J. Recent Technol. Eng. **7**(1), 4 (2019)
2. Kurapati, N., Bhansali, P.K.: Predicting the credit defaulters using machine learning techniques. Int. J. Manag. Technol. Eng. **8**(11), 6 (2018)
3. Jinwang, W., Zhouyi, G.: Customer credit risk assessment of commercial banks based on unbalanced samples – a case study of bank A. Finan. Theory Pract. **07**, 51–57 (2018)
4. Zhao, J., Lu, H.: An over sampling random forest algorithm for unbalanced data classification. Comput. Appl. Softw. **36**(04), 255–261+316 (2019)
5. Wei, Z.: Research on stochastic forest algorithm based on unbalanced data (2017)
6. Dong, L., Wang, Y.: Adaptive random sampling algorithm based on maximum equilibrium degree. J. Northeast Univ. (Nat. Sci. Ed.) **39**(06), 792–796 (2018)
7. Alabdulkarim, A., Al-Rodhaan, M., Tian, Y., Al-Dhelaan, A.: A privacy-preserving algorithm for clinical decision-support systems using random forest. Comput. Mater. Continua **58**(3), 585–601 (2019)
8. Fang, K., Jianbin, W.: A review of random forest methods. Stat. Inf. Forum **26**(03), 32–38 (2011)
9. Shan, T., Zhang, M.: Risk analysis of P2P network loan default based on random forest. China Science and Technology Paper Online (2019)
10. Zhu, Y.: Credit bond default risk measurement based on KMV stochastic forest model (2019)
11. Ma, X., Sha, J., Wang, D.: Study on a forecast of P2P network loan default based on the machine learning LightGBM and XGboost algorithms according to different high dimensional data cleaning. Electron. Commer. Res. Appl. **31**, 24–39 (2018)
12. Li, W.: Research on P2P network loan default forecast model based on integrated classification algorithm (2019)

13. Yan, T., Wang, X.: Research on the early warning of P2P network loan default risk based on machine learning–evidence of loan transaction from ‘auction loan’. Stat. Inf. Forum **33**(06), 69–76 (2018)
14. Ma, C., Zhao, H.: Research on the credit risk factors of P2P network loan subject based on random forest classification model. Jilin Univ. J. Soc. Sc. Ed. **59**(03), 39–48+231–232 (2019)
15. Dong, X.: Application of random forest in credit evaluation of P2P network borrowers (2019)
16. Zhou, L.: Research on loan default forecast based on unbalanced data classification (2013)
17. Li, L.: Research on enterprise credit risk evaluation based on stochastic forest algorithm (2012)
18. Qu, Y.: Stochastic forest forecast model of P2P network loan default (2018)
19. Xiaohong, Yu., Lou, W.: Credit risk evaluation, early warning and empirical research of P2P network loan based on random forest. Finan. Theory Pract. **02**, 53–58 (2016)
20. Cao, W., Li, C.: A comparative study of credit risk early warning model of P2P network lending in China based on Integrated Learning. Data Anal. Knowl. Discov. **2**(10), 65–76 (2018)
21. Zheng, J.: Research on credit evaluation of P2P borrowers based on stochastic forest model (2017)
22. Wang, S.: Comparative study on credit risk control of P2P network loan borrowers based on several common models (2019)