Loan Default Prediction based on Machine Learning (LightGBM Model)

Xingzhe Dong

International School, Beijing University of Posts and Telecommunications, Beijing, China.

dongxingzhe@bupt.edu.cn

Abstract. With the rapid development of Internet finance, the number of online loan platforms has been increasing, and the scale of loan business is gradually expanding. At the same time, the bad debt and non-performing loan ratios have also risen sharply, the reasons for which include incomplete information of lenders, imprudent management of capital chain, inaccurate default prediction, etc. Therefore, loan platforms need a set of efficient and accurate loan default prediction solutions to ensure the healthy operation of Internet finance and avoid huge losses for platforms and investors. Current loan default prediction models mainly contain traditional machine learning methods such as logistic regression and decision tree, as well as integrated models such as Random Forest and GBDT. In this paper, we use LightGBM machine learning model to process massive loan default information using the lender information dataset on Tianchi Platform, and the process includes data preprocessing, feature engineering, model training and evaluation. The experimental results show that the LightGBM algorithm has a strong ability to predict loan default, with a final AUC value reaching about 0.73. This method is helpful for banks and Internet loan platforms to conduct background investigation and default prediction of loan applicants, so as to strengthen the ability of loan risk management.

Keywords: Loan default prediction; LightGBM algorithm; Machine learning.

1. Introduction

Finance is the core of the modern economic system, whose existence and operation depend on good social credit. According to the People's Bank of China's Financial Statistics Report for the First Half of 2022 , China's RMB loans increased by 13.68 trillion yuan in the first half of 2022, 919.2 billion yuan over the previous year. Divided into different sectors, household loans increased by 2.18 trillion yuan. Loans to enterprises and institutions increased by 11.4 trillion yuan, including 2.99 trillion yuan for short-term loans, 6.22 trillion yuan for medium and long-term loans, and 2.11 trillion yuan for bill financing. Loans to non-banking financial institutions increased by 10.3 billion yuan.

However, in recent years, the lack of credit for bank loans has become increasingly serious. According to statistics, direct economic losses due to enterprises and individuals evading debts are about 180 billion yuan every year in China. The credit economy has become a "bad debt economy" with serious credit problems. In 2021, the non-performing loan ratio of Chinese banks was 1.33%, 0.13% less than that of the end of the previous year. In the first quarter of 2022, the balance of non-performing loans in the banking industry was 3.7 trillion yuan, and the ratio of non-performing loans was 1.79%, slightly down from the beginning of the year. Although the non-performing loan ratio is relatively low now, the existing non-performing loans will still cause large losses to banks and lending institutions. Therefore, it is particularly important to predict the default possibility of lenders.

Currently, the research on loan default prediction is mainly based on the traditional machine learning models of logistic regression and decision tree.

1.1 Logistic Regression Algorithm

Logistic regression is a widely used algorithm that derived from linear regression, but since the target variable of linear regression is continuous, it is not accurate enough in classification. Therefore, logistic regression is used to characterize the probability that a sample belongs to a certain class.

In 2015, Akwaa-Sekyis et al. used logistic regression on the data of 224 commercial customers from a national branch of a bank in Ghana to study and identify factors that significantly affect

commercial loan default, including loan price, loan purpose, loan duration, etc. . In 2017, Fangke Luo et al. used Logistic regression model to assess the credit risk of individual microloans. By constructing a binary classification logistic credit risk assessment model, the evaluation results showed that the gender, age, income, occupation and geographical location of the client were all related to the credit risk of individual microloans .

However, when the feature space of the dataset is large, logistic regression does not perform very well and is easy to underfit, and the accuracy is generally not high. In addition, logistic regression cannot process a large number of multi-class features or variables quickly. When there is a large amount of loan information, the processing efficiency of logistic regression algorithm is low. Also, logistic regression can only deal with binary classification problems, and must be linearly separable. For nonlinear features, transformation is required before regression.

1.2 Decision Tree Algorithm

In decision analysis, decision trees can be used to visually and explicitly represent decisions and their relationships. Decision trees, which use tree-like decision models, are a common tool in data mining for deriving strategies to achieve specific goals, including classification and regression. The key to building a tree is to select the best features. As feature selection methods, decision tree algorithms are generally divided into ID3, C4.5 and CART. In the decision tree algorithm, we can clearly see the importance of features and see the relationship between features.

Yuan Xin et al. used decision tree to predict the Titanic dataset in 2020. The processed data were introduced into the validated decision tree model to obtain the prediction results. This approach is often referred to as decision tree learning from data. Figure 1 shows the simple structure of the predictive decision tree.

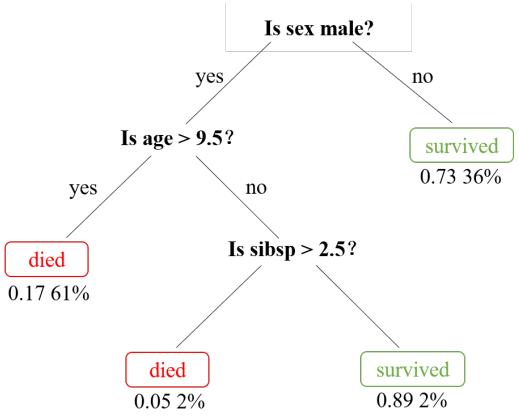

Figure 1. Titanic passenger survival prediction decision tree

However, the decision tree is not very effective for more complex data, which will cause overfitting. The greedy algorithm cannot guarantee the global optimality of decision trees. At the same time, the stability of decision trees is poor, and the small changes in data will lead to the generation of completely different trees. Therefore, we keep introducing new techniques to improve the algorithm capability, such as boosting.

1.3 Integration Model

Integration models are used to accomplish the learning task by combining multiple learners. Through ensemble method, multiple weak Learners can be combined into a strong Classifier, so the generalization ability of ensemble learning is generally better than that of a single classifier. Ensemble methods mainly include Bagging and Boosting, both of which combine existing classification or regression algorithms in a certain way to form a more powerful classification. Different ensemble methods lead to different results. Common integration models based on Bagging idea include random forest, and integration models based on Boosting idea include Adaboost, GBDT, XGBoost, LightGBM, etc.

In 2016, using the online personal loan data, Xiangdong Liu et al. used random forest algorithm to construct a personal loan default prediction model. The study found that the random forest model is more suitable for credit risk assessment, followed by CART, ANN, and C4.5. Information of users such as marriage, house/car property (loan) is of low importance, while credit file information such as company size, working hours, historical borrowing and credit score is particularly important in credit risk assessment. In 2020, Yiding Liu and Jing Xu proposed a financial early warning model based on Adaboost strong classifier, aiming to solve the defects of the complexity and low accuracy of the enterprise financial early warning system model. The experimental results show that compared with the weak classifier with BP neural network, the financial early warning model based on Adaboost strong classifier has higher classification accuracy.

The first chapter of this paper introduces the main methods used in previous studies. The second chapter focuses on the theory and application principle of LightGBM algorithm, and then introduces the calculation method of the evaluation index AUC. The third chapter is the detailed steps of the experiment, including the data processing method and the modeling process. Finally, it comes to conclusion and the effect of the model is evaluated.

2. Introduction to Algorithm Theory

2.1 LightGBM algorithm

Gradient Boosting Decision Tree (GBDT) belong to ensemble learning algorithms, which include XGBoost and LightGBM algorithms. LightGBM algorithm was proposed by Microsoft in 2017 due to the high computational complexity and time consuming of XGBoost algorithm when encountering massive data. This algorithm incorporates several basic algorithms to improve the accuracy and reduce the complexity of the algorithm.

Histogram-based Decision tree algorithm

LightGBM uses a histogram-based decision tree algorithm, which transforms traversal samples into traversal histograms. It reduces both the memory usage and the computational complexity by using the histogram difference.

Gradient-based One-side Sampling algorithm

LightGBM uses Gradient-based One-Side Sampling (GOSS) to filter out the samples with small gradients and retain the data with large gradients, which reduces the information gain and a large amount of computation. GOSS first sorts all the absolute values of the features to be split in descending order, selects $a \times 100\,\%$ data samples with the largest absolute value, and then randomly selects $b \times 100\,\%$ data samples from the remaining small gradient data samples. Then, these data are multiplied by a constant $\frac{1-a}{b}$ to generate a small gradient sample point set. The large gradient and the small gradient set are merged, and the small sample gradient is multiplied by a weight. Repeat these steps until reaching the goal. This algorithm can focus on undertrained samples without worrying about changing the distribution of the original dataset.

Leaf - wise algorithm

LightGBM uses a Leaf-count oriented (Leaf-Wise algorithm with depth restrictions) decision tree building algorithm instead of the decision tree depth-oriented (Level-wise grown by layers) growth strategy used by most GBDT tools.

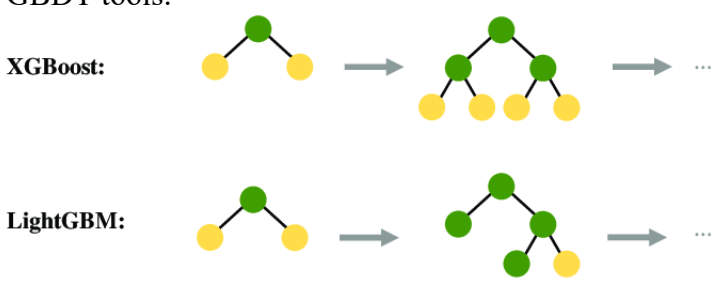

Figure 2. Comparison of XGBoost Level-wise growth tree and LightGBM Leaf-wise growth tree

As can be seen from Figure 2, Level-wise can split the leaves of the same layer simultaneously, which is convenient for multi-threaded optimization. It can effectively control the model complexity and avoid overfitting. However, the Level-Wise algorithm is relatively inefficient, because it discriminates the levels of leaves in the same layer, resulting in unnecessary overhead. Many leaves have low splitting gain, which does not require searching and splitting.

Leaf-wise is more efficient, finding the leaf with the largest split gain from all the current leaves, and then it splits, and keeps repeating. Compared with Level-wise, Leaf-wise can reduce the error and get better accuracy with the same number of splits. The disadvantage of Leaf-wise is that it may lead to deep decision trees and overfitting. Therefore, LightGBM adds a maximum depth restriction on Leaf-wise to prevent overfitting while ensuring high efficiency.

Feature Bundling algorithm

LightGBM also applies the Exclusive Feature Bundling (EFB) algorithm. In the training process, it can bundle the features that are mutually exclusive, that is, the two features whose values will not take zero at the same time, or the conflict rate is low, into one feature for processing, which reduces the number of features and memory consumption.

2.2 Evaluation Index AUC-ROC

In logistic regression, for the definition of positive and negative cases, a threshold value is usually set. The cases greater than the threshold value are considered as positive class, and the cases less than the threshold value are considered as negative class. If we reduce this threshold, more samples will be identified as positive classes, which will improve the recognition rate of positive classes, but at the same time, more negative classes will be incorrectly identified as positive classes. In order to visually represent this phenomenon, ROC (Receiver Operating Characteristics) is introduced.

ROC is the probability curve, and AUC (Area Under the Curve) indicates the degree or measure of separability. It can reflect the level of model classification. When it comes to classification problems, we can rely on the AUC-ROC curve. When we need to examine or visualize the performance of a multi-class classification problem, we use the AUC-ROC Curve. It is a performance measure of classification problems under various threshold settings, and it is one of the most important evaluation metrics to check the performance of any classification model.

According to the definition of the Confuse Matrix,

(1) If an instance is of Positive class and is predicted to be of Positive class, it is TP (True Positive) class.

(2) If an instance is of Positive class but is predicted to be of Negative class, it is FN (False Negative) class.

(3) If an instance is of Negative class but is predicted to be of Positive class, it is FP (False Positive) class.

(4) If an instance is of Negative class and is predicted to be of Negative class, it is TN (True Negative) class.

Among all the actual positive examples, the ratio of the correctly judged positive samples is the true positive rate TPR:

$$TPR = \frac{TP}{TP+FN} \tag{1}$$

Among all the actual negative examples, the ratio of the falsely judged positive examples is the false positive rate FPR:

$$FPR = \frac{FP}{FP+TN} \tag{2}$$

ROC space defines FPR as the X-axis and TPR as the Y-axis.

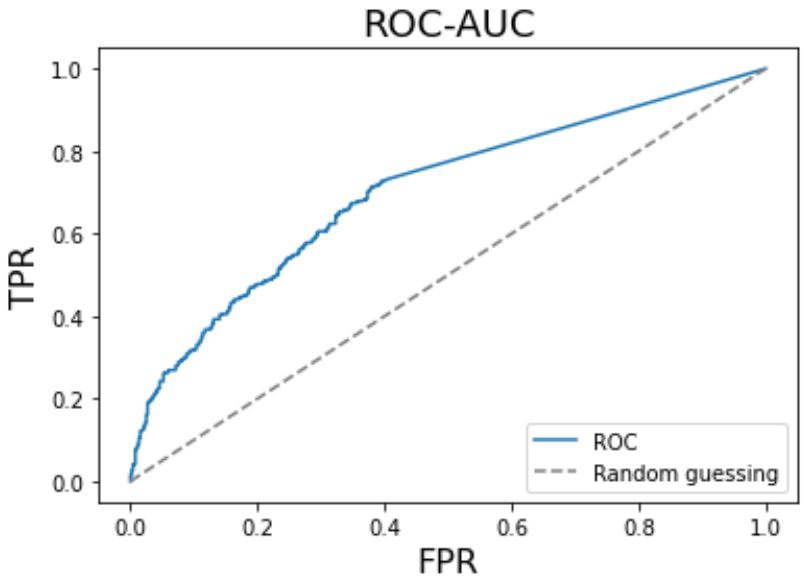

Figure 3. AUC-ROC curve

As shown in Figure 3, the dotted line represents the case of random classification, with the AUC value of 0.5. At this point, the model has the lowest authenticity and no application value. The closer the ROC curve is to the Y-axis, the higher the AUC value is, the better the model could predict class 0 as 0 and class 1 as 1. Therefore, the higher the AUC value, the better the model's ability to tell whether a lender is in default. We aim for a higher AUC indicator for the model, at which point the model has a better ability to predict loan defaults.

3. Empirical Analysis

This chapter introduces the sources and variable names of all lender information data, and describes the experimental process in detail, including data analysis and feature engineering.

3.1 Data Sources

The data used in the experiment come from the Financial Risk Control-Loan Default Prediction Challenge dataset of Tianchi Platform, containing 800,000 pieces of data as the training set, 200,000 pieces of data as the test set A, and 200,000 pieces of data as the test set B. It contains a total of 47 columns of variable information, of which 15 columns are anonymous variables. All the features are listed in the table below.

Table 1. Description of the original data set indicators

Variable Attributes	Variable Name	Variable Meaning	Variable Type
Loan Information	id	The unique credit identification assigned to the loan manifest	numeric
	loanAmnt	The loan amount	numeric
	term	Loan term (years)	numeric
	interestRate	The loan interest rate	numeric
	installment	Installment amount	numeric
	grade	Credit rating	types: A, B, C, D, E, F, G
	subGrade	Loan grade Subgrade	types: A1-A5, B1-B5, C1-C5, D1-D5, E1-E5
	issueDate	The month in which the loan is issued	date variable

Basic Borrower information	employmentTitle	Job title	numeric
	employmentLength	Years of employment (years)	date variable
	homeOwnership	Home ownership status provided by the borrower at the time of enrollment	types
	annualIncome	Annual income	numeric
	purpose	Borrower's loan purpose category at the time of loan application	type
	postcode	The first 3 digits of the zip code provided by the borrower in the loan application	type
	regionCode	Area code	type
	dti	Debt-to-income ratio	numeric
Borrower credit information	delinquency_2years	Number of delinquencies in a borrower's credit file that were more than 30 days past due in the past 2 years	numeric
	ficoRangeLow	The lower limit range to which the borrower's FICO falls at the time the loan is issued	numeric
	ficoRangeHigh	The upper limit range to which the borrower's FICO falls at the time the loan is issued	numeric
	openAcc	Number of open lines of credit in the borrower's credit file	numeric
	pubRec	The number of derogatory public records	numeric
	pubRecBankruptcies	Number of public record removals	numeric
	revolBal	Total credit turnover balance	numeric
	revolUtil	Revolving line utilization ratio, or the amount of credit used by the borrower relative to all available revolving credit	numeric
	totalAcc	Total number of current credit lines in borrower's credit file	numeric
	initialListStatus	Initial list status of the loan	type 0/1
	applicationType	Indicate whether the loan is an individual application or a joint application with two co-borrowers	numeric
	earliesCreditLine	The borrower the earliest report of credit open a month	date variable
	title	The name of the loan provided by the borrower	numeric
	policyCode	Publicly available policyCode =1; New products not publicly available policiesCode =2	type 1/2
N-series Anonymous Features	n0-n14	Treatment of some lender behavior count characteristics	numeric

3.2 Data Analysis

Search for features with missing values

After reading the training set and test set, the overall understanding of the data set is first carried out, including the number of samples and the original feature dimensions of the data set. The training

set has 800,000 pieces of information and 47 variables, and the test set A has 200,000 pieces of information and 48 variables. With the info() function, we can read the number of non-null values of each column of variables and the type of data.

Then check the missing values of features in the dataset, and count the number of missing values and the missing rate of each feature. The missing rate of features is shown by a bar graph. The features with missing values and their missing rates are shown in Figure 4.

Figure 4 indicates the number of null values vertically. When a feature has too many null values, we need to consider whether to retain this feature. For example, n11 has 13033 missing values, and the missing rate reaches about 0.09. When a feature has few missing values, we can fill it.

Numerical and objective classification of features

After finding the missing values, we classify the features and perform subsequent processing.

For all the features, we can broadly classify them into categorical features and numerical features, and put them into category_fea feature set and numerical_fea feature set, respectively. Categorical features sometimes have non-numerical relations, and sometimes have numerical relations. Different relationships require to take different approaches to assign values to different feature attributes. For numerical features, it is necessary to reduce the complexity of variables and lower the influence of noise through steps of binning and coding before constructing the model. In this experiment, the categorical features are grade, subGrade, employmentLength, issueDate, earliesCreditLine, and all other variables are numerical features.

As for the numerical characteristics, we also divide them into continuous variables and discrete variables. For discrete features, we list each of the attribute values and the number of that value. For continuous numerical variables, the distribution of their digital features can be visualized, and the distribution of their digital features is shown in Figure 5. At the same time, we try to make their distribution conform to the normal distribution, and for those that do not, we take the logarithm of them before observing their distribution. For example, we normalize the loanAmnt variable, as shown in Figure 6.

After normalization, the range of its numerical distribution is narrowed, and the probability is also distributed between 0 and 1, which is convenient for observation. In addition, for some special features, we can visualize their distribution according to whether they are in default or not, as shown in Figure 7.

Figure 7 and 8 respectively show the distributions of discrete variables grade and employment Length and continuous variable Log LoanAmnt in cases of default and non-default.

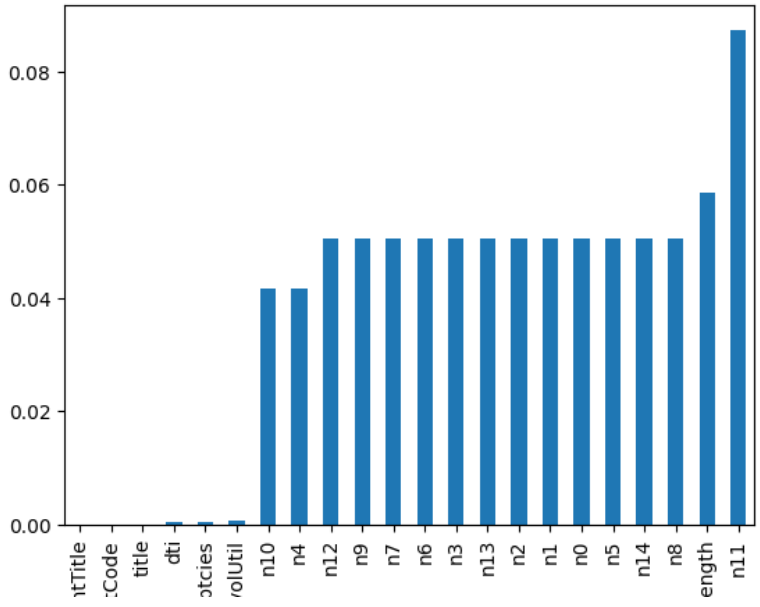

Figure 4. Features with missing values and their missing rates

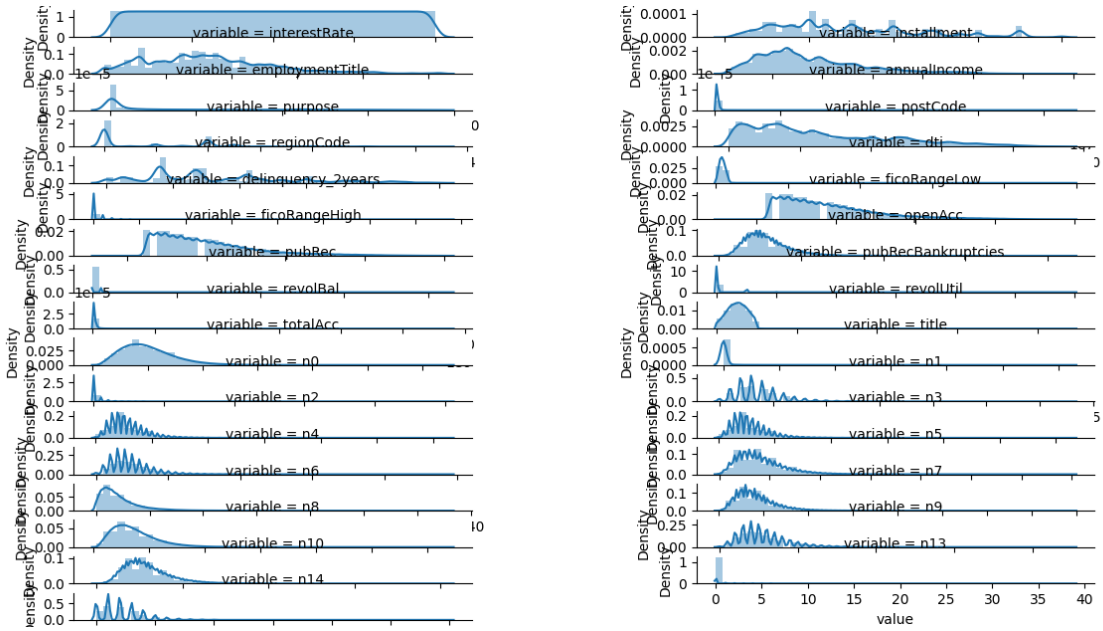

Figure 5. Numerical distribution of continuous variables

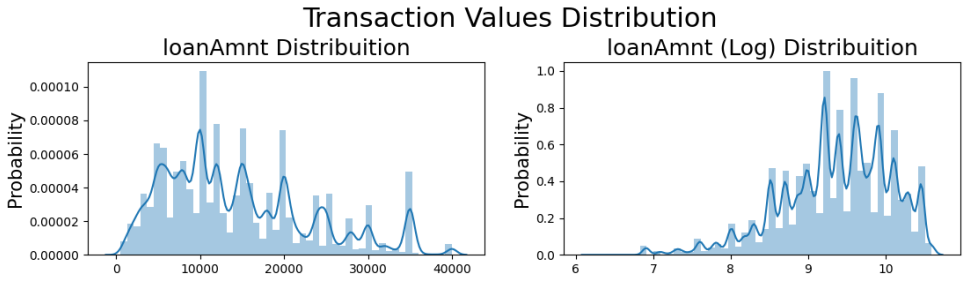

Figure 6. Distribution of loanAmnt before and after normalization

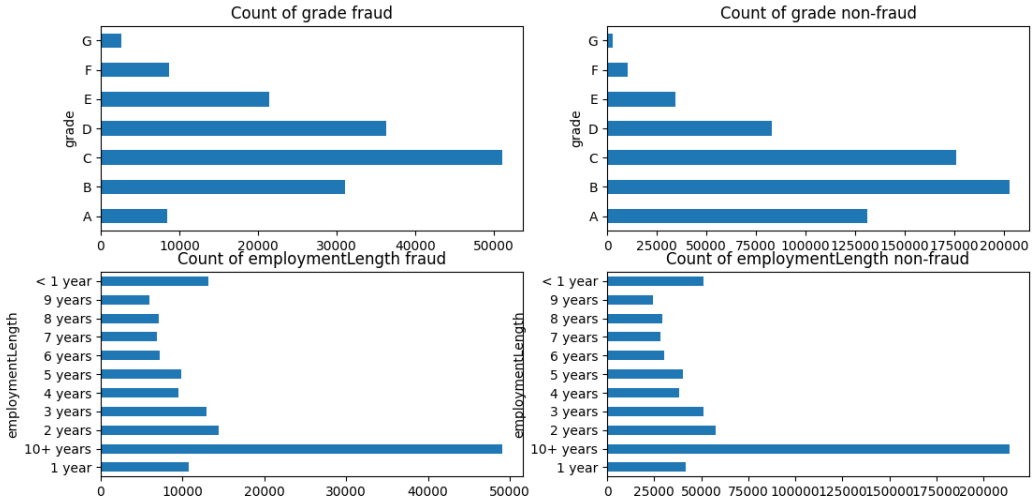

Figure 7. Distribution of discrete variables grade and employmentLength in cases of default and non-default

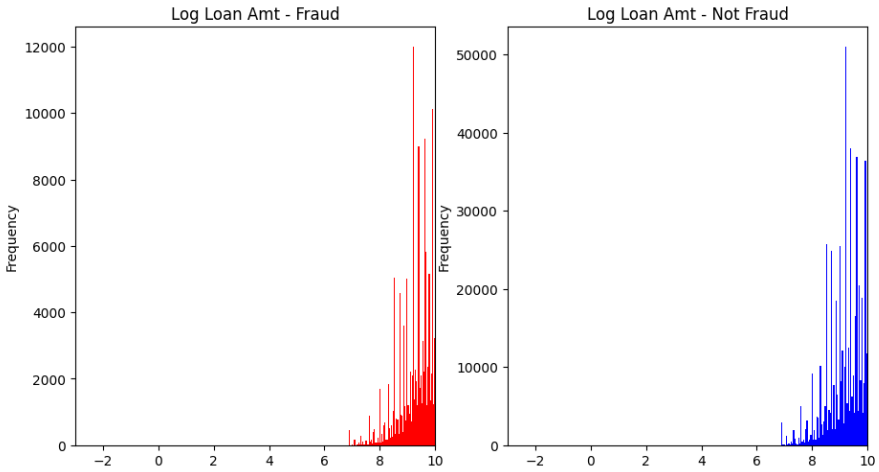

Figure 8. Distribution of the continuous variable *Log LoanAmnt* in cases of default and non-default

3.3 Feature Engineering

The feature engineering part mainly includes data preprocessing, data bucketing (i.e. data binning), feature interaction and feature coding.

3.3.1 Data Preprocessing

In this experiment, on the basis of the above data analysis, data preprocessing is carried out, which mainly includes the conversion of date variables, the deletion of redundant features, and the conversion of text type feature data to numerical values.

Date conversion

For the time feature issueDate, it should first be converted into time format (e.g. yyyy-mm-dd). Then, set the start date 2007-06-01 and subtract the start date from each issueDate data. The number of days difference dateLength indicates the length of the loan period. The larger the value is, the later the loan is issued, which can be used for modeling.

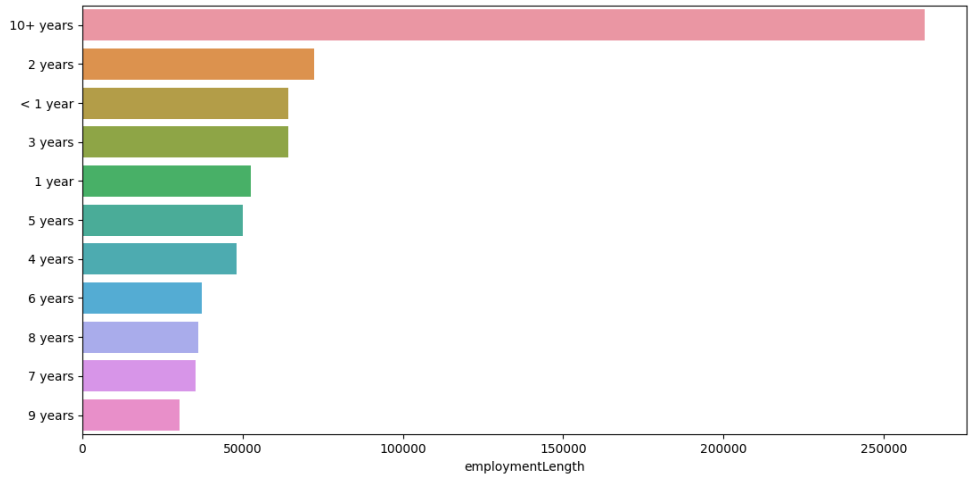

Figure 9. Visualization of employmentLength distribution

Redundant features removel

During the data analysis step, we find that some features are not useful. The variable issueDate is no longer needed after the time format transformation, so it can be deleted. In addition, we find the feature policyCode with only unique value through nunique() function, which has no use. The anonymous variables n2 is the same as n3, the redundant feature n2 can be deleted.

Conversion of text type features to numeric values

For the feature employmentLength, visualize it as shown in Figure 9. As is shown, the representation of the attribute value is “<1 year”, “1 year”, ..., “10+ years”, etc., which are not in

numerical formats and cannot be recognized by the model. Therefore, we convert "<1 year" to "0 year", "10+ years" to "10 years", and convert the employmentLength variable to int type to facilitate model identification.

3.3.2 Data Bucketing

For some variables, their value distribution span is relatively large. We have to group data according to specific rules and then use the quantified results. Data binning enables discretization of data, enhances data stability and reduces the risk of over-fitting.

In this experiment, the value span of loanAmnt variable is from 0 to 40000, so the fixed-width binning is used, with 1000 as the fixed width. The value range of each binning is loanAmnt/1000. For annualIncome and dti, the logarithmic function is used to map the values to the exponential-width bin.

3.3.3 Feature Interaction

Feature interactions can represent results formed by a pair of conditions, similar to the logical operator "AND". In the experiment, grade and subgrade are interacted, and then the anonymous variables n0-n14 are interacted to create connections between them. The interaction of grade and subgrade can represent the result when both variables are in effect, while the interaction of anonymous variables can effectively extract information from anonymous variables when the feature meaning is unknown.

3.3.4 Feature Encoding

For categorical variable grade, which, according to the analysis, contains seven levels "A, B, ..., G ", and they have priority. Therefore, we can use LabelEncoder to encode them. The seven levels correspond to seven numbers from 1 to 7.

3.4 Model Training and Evaluation

After processing the dataset, the LightGBM model should be constructed. The basic model is introduced, and the parameters are shown in Table 2. Using 5-fold cross-validation, a total of 10 AUC values are obtained in 5 groups, in which the maximum value is 0.723898, the minimum value is 0.719017, and the average value is 0.7222943. The AUC value is ideal, which indicates that the model has good learning ability and prediction ability.

At the same time of model training, the importance of all features is ranked using the analysis function of LightGBM. As can be seen from Figure 10, the variable dateLength obtained from 3.3.1, i.e., the time of loan issuance, has the greatest influence. The second is rest_Revol, which is a self-defined feature, representing the loan amount minus the balance of credit turnover. In addition, factors such as borrower area code, debt-to-income ratio, and the self-defined feature closeAcc (total number of credit lines – open number of credit lines) all have a great impact on the probability of loan default.

Table 2. Partial parameters of the model

LightGBM Parameter Name	Parameter values
num_leaves	64
min_child_weight	10
max_depth	- 1
learning_rate	0.05
min_child_samples	10
reg_alpha	0
reg_lambda	0.01
seed	2020
n_estimators	2000
subsample	0.7
colsample_bytree	0.07
subsample_freq	1

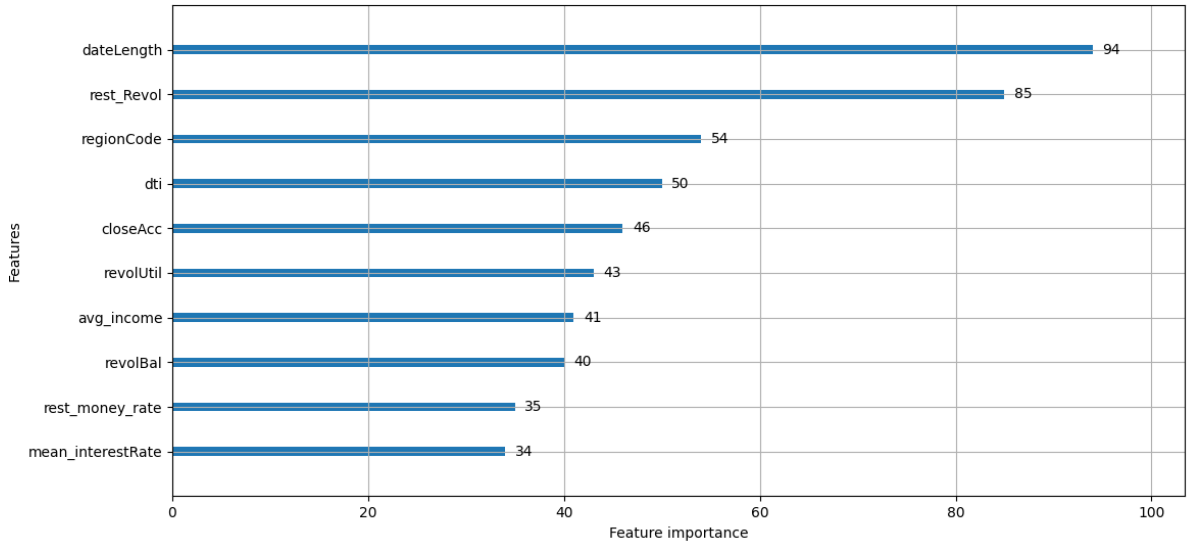

Figure 10. Rank of feature importance

4. Conclusion

Through the experiment, it can be found that the time of loan issuance, the debt-to-income ratio of the borrower, the region and other factors have a great impact on the probability prediction results. The AUC value of the model we constructed in the experiment is relatively ideal. Through the analysis of the effect of the traditional model in the literature review section and the data processing, model construction and the evaluation of the AUC value in the experimental section, it can be shown that LightGBM algorithm has high processing efficiency and strong prediction ability in the face of massive data. Therefore, LightGBM algorithm can be used as loan default prediction models in large banks and Internet loan platforms.

Financial institutions should improve the comprehensiveness of the information of lenders when dealing with loan business, so as to obtain more accurate data used in the prediction of the probability of loan default and establish a more three-dimensional risk assessment system. As for the lenders, they need to pay attention to the quality of all aspects of their own information, especially the information strongly related to their credit degree, such as debt-to-income ratio, the difference between loan amount and working balance, average income and so on. These important factors will determine whether the lender can obtain the loan qualification.

References

[1] Financial Statistics Report for the first half of 2022 (PBC.gov.cn), 2022, 7.

[2] Zhang Hanping. Research on Default prediction of personal Loan based on LightGBM Model [D]. Central China normal university, 2021. DOI: 10.27159 /, dc nki. Ghzsu. 2021.000706.

[3] Akwaa Sekyi, Ellis Kofi; Bosompra, Portia. (2015) . Determinants of business loan default in Ghana. Junior Scientific Researcher, 2015, vol. 1, Num. 1, p. 10-26. http://hdl.handle.net/10459.1/65930.

[4] Luo Fangke, Chen Xiaohong. [LUO F K, Chen X H. Credit risk assessment of individual small loan based on Logistic regression model and its application. Theory and practice of finance and economics, 2017, 38 (01) : 30-35, DOI: 10.16339 / j.carol carroll nki HDXBCJB. 2017.01.005.

[5] Yuan Xin, Duan Huaqiong. Based on decision tree algorithm of Titanic data prediction [J]. Computer knowledge and technology, 2020 (22) : 185-186 + 199. DOI: 10.14004 / j.carol carroll nki CKT. 2020.2622.

[6] https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052

[7] Liu X D, Li F, Wang J, et al. credit risk assessment of online lending under the background of big data: a case study of renrendai [J]. Statistics and information forum,2016,31(05):41-48.]

[8] Liu Yiding, Xu Jing. Financial early warning model based on strong Adaboost classifiers [J]. Journal of modern business, 2020 (31) : 187-188. The DOI: 10.14097 / j.carol carroll nki. 5392/2020.31.075.

[9] Rezazadeh, Alireza. (2020). A Generalized Flow for B2B Sales Predictive Modeling: An Azure Machine-Learning Approach. Forecasting. 2. 267-283. 10.3390/ Forecast2030015.

[10] Nathan Aim. (2020). Mathematics behind ROC-AUc Interpretation, from https://medium.com/analytics-vidhya/mathematics-behind-roc-auc-interpretation-e4e6f202a015

[11] Tang Yifeng. Research on Loan default prediction model based on XGBoost algorithm and LightGBM algorithm [J]. Modern computer,2021,27(32):33-37.

[12] Zhang Hanping. Based on the research of personal loan default LightGBM model [D]. Central China normal university, 2021. The DOI: 10.27159 /, dc nki. Ghzsu. 2021.000706.