**Loan Default Prediction with Artificial Intelligence Machine Learning**

Yehan Li

Artificial Intelligence, Coventry University

6072CEM: Individual Project (Artificial Intelligence)

Dr. Jeffrey Ting

March 28, 2024

**Text number: 4100**

**Abstract**

With the increasing demand for loans and the frequent default events in the bond market and credit market, the non-performing asset rate of commercial banks and the default risk of user personal loans have become the core issues concerned by the government and the banking industry, and the effective assessment and measurement of credit loan default risk has become the core task for commercial banks to improve their management level.

In this project, personal credit in financial risk control as the background, according to the data information of loan applicants to predict whether there is the possibility of default, we are committed to establish a variety of machine learning models, based on the data of loan default information, model training, used to predict whether the lender is likely to default.

In this experiment, We used some of the most popular algorithms of our time, which including Decision tree, Bagging, RandomForest, Xgboost, Lightgbm and Catboost, are used to train the loan default data by using the five-fold cross validation method. The AUC index is used as the evaluation index to evaluate the six models, and the best model is catboost model, which has the highest AUC of 74.88%, and the data visualization analysis report is generated.

**Key words:** Data analysis; Data imbalance; Machine learning; Cross validation; Ensemble learning

**Table of Contents**

[1. Problem Description 5](#_Toc162372414)

[1.1 Background of the problem 5](#_Toc162372415)

[1.2 Key technologies Required 5](#_Toc162372416)

[2. Data Analysis 7](#_Toc162372417)

[2.1 Data acquisition 7](#_Toc162372418)

[2.2 Statistical interpreation of data 7](#_Toc162372419)

[2.3 Data visualization 8](#_Toc162372420)

[3. Model Introduction 13](#_Toc162372421)

[3.1 Decision-making tree 13](#_Toc162372422)

[3.2 Bagging 14](#_Toc162372423)

[3.3 Random forest 14](#_Toc162372424)

[3.4 XGBoost 15](#_Toc162372425)

[3.5 LightGBM 15](#_Toc162372426)

[3.6 CatBoost 16](#_Toc162372427)

[4. Model Implementation 16](#_Toc162372428)

[4.1 Experimental introduction 16](#_Toc162372429)

[4.2 Decision tree 17](#_Toc162372430)

[4.3 Bagging 19](#_Toc162372431)

[4.4 Random forest 20](#_Toc162372432)

[4.5 XGBoost 21](#_Toc162372433)

[4.6 Lightgbm 23](#_Toc162372434)

[4.7 CatBoost 25](#_Toc162372435)

[5. Conclusion 27](#_Toc162372436)

[6. Bibliography 29](#_Toc162293949)

[7. Appendix 31](#_Toc162293949)

**Table of Figures**

[Figure 1 Viewing data dimensions 7](#_Toc162293927)

Figure 2 Data missing scale chart [8](#_Toc162293928)

[Firgue 3 Data category label ratio 8](#_Toc162293929)

[Figrue 4 Visualization of the distribution of each variable 10](#_Toc162293930)

[Figrue 5 Distribution of transaction amount values 11](#_Toc162293931)

[Figrue 6 Distribution in certain years 11](#_Toc162293932)

[Figrue 7 Proportion of category tags 0 and 1 in total data (left) and in total loan amount (right) 12](#_Toc162293933)

[Figrue 8 Graph of loan time distribution in training set data and test set data 12](#_Toc162293934)

[Figrue 9 Thermal maps of the correlation between the various features 13](#_Toc162293935)

[Figrue 10 Data set partitioning operation code 16](#_Toc162293936)

[Figrue 11 Decision tree implementation operation code 17](#_Toc162293937)

[Figrue 12 Decision tree run result confusion matrix 17](#_Toc162293938)

[Figrue 13 Decision tree model training effect AUC 18](#_Toc162293939)

[Figrue 14 The top ten feature and partial structure trees in the decision tree model 18](#_Toc162293940)

[Figrue 15 Bagging run result confusion matrix 19](#_Toc162293941)

[Figrue 16 Bagging model training effect AUC 19](#_Toc162293942)

[Figrue 17 Random forest run result confusion matrix 20](#_Toc162293943)

[Figrue 18 Random forest model training effect AUC 20](#_Toc162293944)

[Figrue 19 Rank the top ten important features of the output of the random forest model 21](#_Toc162293945)

[Figrue 20 XGBoost implements code operations 22](#_Toc162293946)

[Figrue 21 Changes of AUC during XGBoost model training 22](#_Toc162293947)

[Figrue 22 XGBoost model training effect AUC 23](#_Toc162293948)

[Figrue 23 Lightgbm implements code operations 23](#_Toc162293949)

[Figrue 24 Changes of AUC during Lightgbm model training 24](#_Toc162293944)

[Figrue 25 Lightgbm model training effect AUC 24](#_Toc162293945)

[Figrue 26 Structure tree of Lightgbm model 25](#_Toc162293946)

[Figrue 27 CatBoost implements operational code 25](#_Toc162293947)

[Figrue 28 Changes of AUC during CatBoost model training 26](#_Toc162293948)

[Figrue 29 catboost model training effect AUC 27](#_Toc162293949)

[Figrue 30 Each model evaluation renderings 27](#_Toc162293946)

|  |
| --- |
| **Academic Report** |

# 1. Problem Description

1.1 Background of the problem

In recent years, the loan scale for buying a house, a car and other consumption has increased rapidly. The demand for loans is constantly increasing, followed by frequent default events in the bond market and credit market of each country. The non-performing asset rate of commercial banks and the default risk of user's personal loans have become the core issues concerned by the government and the banking industry, and effective assessment and measurement of credit loan default risk has become the core task for commercial banks to improve their management level.

In this paper, by processing the user loan data and the lender's various characteristics data, analyzing the correlation between the characteristics and the default situation, the personal loan default prediction model is established to help the bank's business personnel to clarify more meaningful indicator variables of the customer, predict the user loan risk in response to the increasing loan review business, and find the potential loss of the loan early. Reduce manual audit costs, improve prediction accuracy, and better provide users with loan services and business.

The data set of this project comes from Kaggle data platform. With personal credit in financial risk control as the background, it predicts whether the loan applicant is likely to default according to its data information, and establishes a model to predict whether the loan is approved.

1.2 Key technologies Required

1. **Matplotlib Data Visualization:** Matplotlib is the main instrument for visualization in Python.It can be implemented in different platofrms including static, dynamic and interactively changeable graphs.

2. **Seaborn Data Visualization:** The extremely flexible library called Seaborn built Python matplotlib matplot extension is a package for making graphs.Because of its interactivity, users can intuitively create so many statistics, which will consequently catch the eye of the viewer.Seaborn is actually a wrapper that relies on matplotlib with a more authentic API, and offers the facility to plot more easily, so to speak Seaborn can be thought of in conjunction with matplotlib.

3. **pandas for Statistical Analysis:** pandas is a very huge Python package used in the analysis of statistical data structure and it is based on Numpy.It provides an out-of-the-box data structure based on python that is fast and easy to use for data analysis in Python.pandas has two data structures namely Series (1D array structure) and DataFrame (2D array structure) which makes it a smarter choice of data analysis.

4. **Machine learning in Scikit-learn:** An open source machine learning library sitting on top of NumPy, SciPy, and Matplotlib called Scikit-learn.Such course explains the most significant methods of classification, regression, clustering (applied algorithms include k-means, random forest, logistic regression, naive Bayes, SVM, and others).Optimized to build upon NumPy and SciPy, two extremely popular Python packages for doing number crunching work.

5. **PCA Data Dimensionality Reduction:** A high-dimensional data set can be reduced in dimension to a low-dimensional data set using the unsupervised dimensionality reduction technique of principal component analysis (PCA).This is achieved by transforming to a new set of variables, the principal components (PCs), which are uncorrelated, and which are ordered so that the first few retain most of the variation present in all of the original variables (Mishra et al., 2017).

6. **Ensemble learning:** Ensemble learning is a machine learning method that performs a learning task by building and combining multiple learners, mainly combining limited models with each other. The goal of ensemble learning is to create a stronger, more comprehensive strong supervised model by combining several weakly supervised models. Ensemble learning is based on the principle that weak classifiers can correct each other's incorrect predictions, even if one of them comes from a weak classifier. Three methods of ensemble learning include stacking, boosting, and bagging.

7. **GridSearch Method:** The GridSearch method is an exhaustive search method for parameter values, and the optimal learning algorithm is obtained by optimizing the estimated function's parameters using the cross-validation method. Grid search is used to find the model's ideal hyperparameters, and the algorithm delivers a good classifier after testing all parameter combinations for the fitting function and automatically adjusting to the best parameter combination.

# 2.Data Analysis

2.1 Data acquisition

The data in this research was downloaded from Kaggle. The loan default prediction dataset includes some anonymous features. The data is derived from loan records on a credit platform, with a total data volume of over 120w and 47 columns of variable information, 15 of which are labeled as anonymous variables. In this experiment, 800,000 pieces were used as the training set and 200,000 pieces as the test set A.

2.2 Statistical interpreation of data

Before pre-processing, a simple statistical analysis of the data can help us to get a preliminary understanding of the dimension and size, type and other descriptive information of the data.

图形用户界面, 文本, 应用程序

描述已自动生成

Fig.1 Viewing data dimensions

As shown in Figure 2-1, by viewing the data dimension of the data set, it can be concluded that the dimension of the training set in this data set is 800000\*47. The dimension of test set A is 200000\*46.

The function data\_train.info() can view the basic information of the data, understand the basic type of the data, and use the function data\_train.isnull().any() can simply view the missing value of the data. Data\_train.isnull ().sum()/len(data\_train) can obtain the missing data ratio, as shown in the figure below:

图表

描述已自动生成

Fig.2 Data missing scale chart

2.3 Data visualization

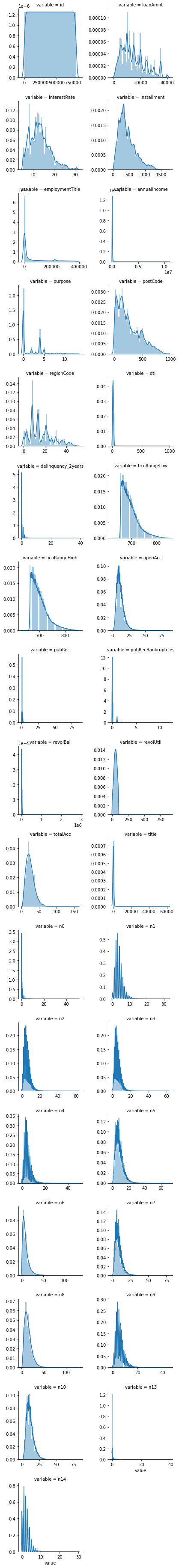
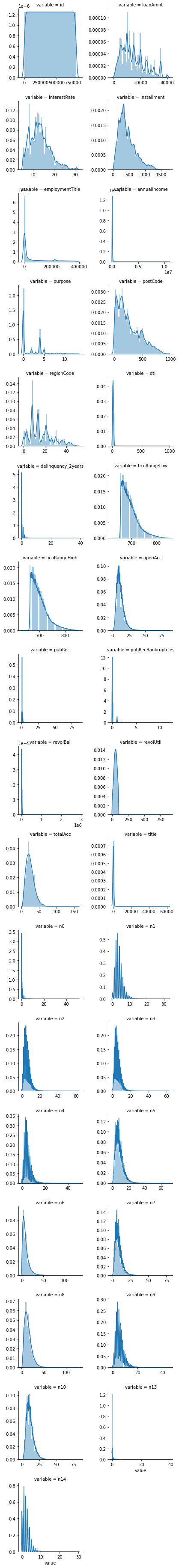
Data visualization is the graphical representation of data. Visualized information can help people quickly and easily obtain data and understand its hidden meaning. Therefore, data visualization is the most effective way to present data analysis results. Data visualization can make the data more intuitive, observe the changing trend of variables, and be easy to understand. Therefore, data visualization plays an important role in data analysis.

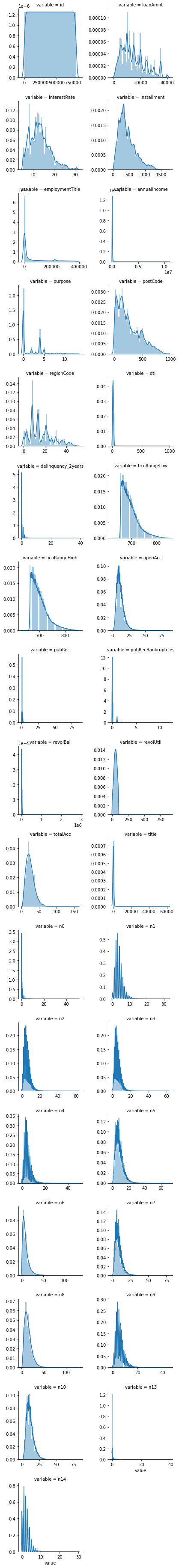
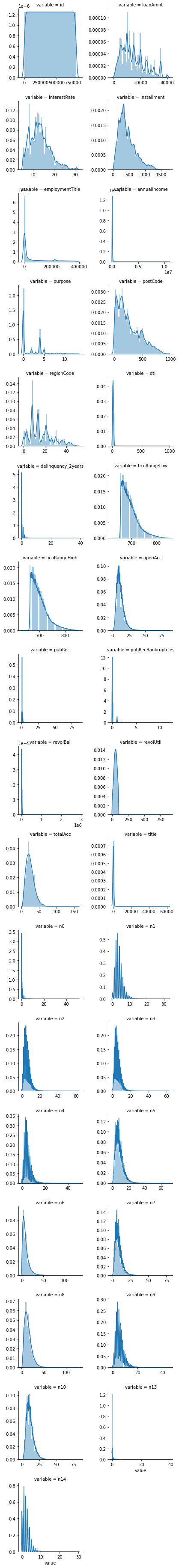
图表, 饼图

描述已自动生成We can view the comparison of the category labels of this data, as shown in the figure below. We can find that the data category label is unbalanced, so it is also a test for the model evaluation indicators selected for the subsequent model.

Fig.3 Data category label ratio

In the data visualization in this experiment, first of all, the distribution of a certain numerical variable is checked to see whether the variable conforms to the normal distribution. Generally, the normal data can make the model converge faster, while the variables that do not conform to the positive distribution can be logized. The distribution of each variable is visualized as follows:





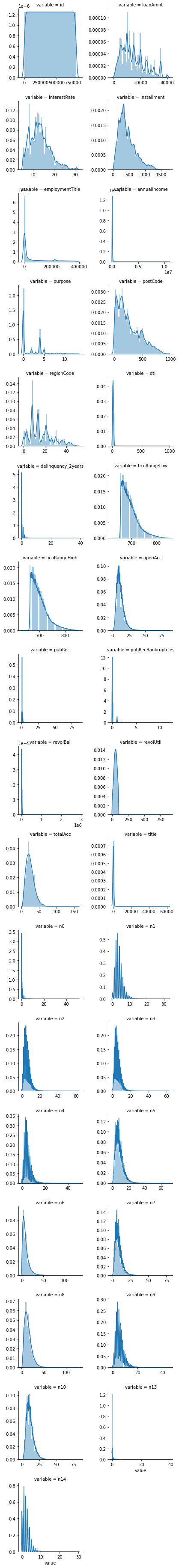
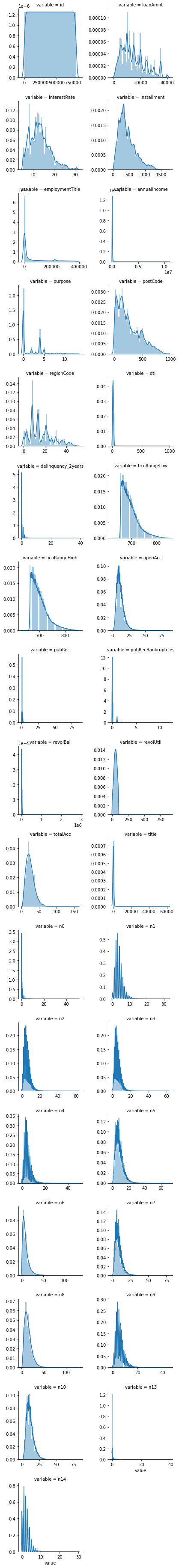


Fig.4 Visualization of the distribution of each variable

Plot the distribution of transaction amount values as shown in Figure 5, where the right graph is the distribution after log.

图片包含 徽标

描述已自动生成

图表, 直方图

描述已自动生成

图表, 条形图

描述已自动生成

Fig.5 Distribution of transaction amount values

We can look at the distribution of some years in the data, as shown in the figure below:

图表, 条形图

描述已自动生成

Fig.6 Distribution in certain years

View the proportion of category tags 0 and 1 in the total data (left) and in the total loan amount (right)

图表, 条形图, 瀑布图

描述已自动生成

Fig.7 Proportion of category tags 0 and 1 in total data (left) and in total loan amount (right)

Looking at the loan time distribution in the training set data and the test set data, we can find that there is a lot of overlap between the two.

图表, 直方图

描述已自动生成

Fig.8 Graph of loan time distribution in training set data and test set data

To view the correlation between the various features, see the following heat map. The darker the color, the stronger the correlation between the two features.

图表, 散点图

描述已自动生成

Fig.9 Thermal maps of the correlation between the various features

# 3. Model Introduction

3.1 Decision-making tree

Decision tree learning algorithms often include selecting the best feature iteratively and segmenting the training data by feature to ensure that each subdata set has the best categorization. This procedure refers to the partitioning of the feature space and the creation of the decision tree. The steps for building a decision tree are as follows:

1. At first, the root node is built, all of the training data is placed on it, an ideal feature is chosen, and the training data set is divided into subsets based on this feature, ensuring that each subset has the best classification under the current conditions.

2. Create leaf nodes and assign these subsets to associated leaf nodes if these subsets can be roughly classified correctly.

3. In the event that certain subsets are incorrectly identified, new, optimal features are chosen for them, and the segmentation process is carried out to create the associated nodes. Recursion ensures that either all training data subsets are essentially accurately classified or that no relevant features exist.

4. In order to create a decision tree, each subset is allocated to a leaf node, meaning that there is an explicit class.

The main advantage of decision trees it that it does not require variables that must abide by a strict statistical assumption. Instead of predicting the length of survival time for each case, like what the Cox proportional hazards model does, decision trees can be applied to screen out the short-term default cases, the ones should be addressed with higher risk. This approach overcomes the drawbacks that the traditional binary credit risk model does not consider the time of default. The accuracy of prediction can also be better than simply using survival analysis (Chang et al., 2016).

3.2 Bagging

Bagging is a popular ensemble learning framework that reduces generalization mistakes by merging numerous models. The basic concept of ensemble learning is to create a group of "individual learners" first, then apply some sort of strategy to combine them. Ensemble learning is not a single machine learning algorithm in and of itself; rather, it is the process of generating and integrating several machine learning machines to fulfill the learning job. The bagging model's primary concepts are to lower the model's variance, increase accuracy, extract several training sets from the sample population, fit the model to each training set, and average each model's output.

Ensemble Bagging and Boosting methods improves performance by combining precictions output of multiple single weak learners. Bagging Meta-Estimator combines multiple base estimators (weak learners) to create a robust model by creating multiple bootstrap samples from the training data and train base models independently on each of these samples. For each iterations the boosting algorithm change the weight of the training data distributions based on miss-classification (Ashraf, n.d. 2024).

3.3 Random forest

Random forest belongs to supervised learning algorithm, is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or means prediction of the individual trees(Zhu et al., 2019).

The steps to construct a random forest are as follows:

1. If there are N samples, then N samples are chosen again at random (one sample at a time, chosen at random, and then chosen again). As the samples at the decision root node, the chosen N samples are utilized to train a decision tree.

2. M attributes are randomly chosen from the m attributes, and the condition m< be divided when each sample contains M attributes and each decision tree node has to be split. Keep in mind that pruning is not done at any point during the decision tree construction process.

3. Every node should be split in accordance with step 2 until it can no longer be split during the decision tree's development. Keep in mind that pruning is not done at any point during the decision tree construction process.

3.4 XGBoost

A kind of decision tree algorithm XGBoost is preordering approach based. This algorithm's core notion is as follows: first, each feature's value determines the feature's preordering. second, when traversing the split points, the optimal split point on a feature is found using the cost of O(#data). The data is split into left and right child nodes at the end, following the identification of the optimal split point for each characteristic.

One notable benefit of employing a pre-sorting algorithm is its ability to accurately identify the segmentation point. However, the drawbacks are also readily apparent: Initially, the spatial utilization is substantial. Such methods need to save twice as much memory as the training set of data because they must store both the eigenvalues of the data and the output of feature sorting (e.g., the sorted index for the next quick computation of the split points). Second, there's a significant time cost as well as a significant consumption cost because each division point requires the computation of split gain. It is not conducive to cache optimization, to sum up. The cache cannot be optimized after the presort because feature access to the gradient is random and feature access orders vary. Moreover, XGBoost will split the leaves in the same layer when building trees, which is not easy to over-fit. However, if there are too many leaf nodes, the splitting gain will be low, which will lead to extra overhead (Cheng, 2021).

3.5 LightGBM

The GBDT model, which has been around for a while in the field of machine learning, works by using decision tree iterative training to find the best model. Its benefits include a good training effect and resistance to overfitting. GBDT is frequently applied to data mining and multi-classification problems. With the help of LightGBM, you can put the GBDT algorithm into practice efficiently in parallel. Its benefits include faster training times, less memory usage, improved accuracy, distributed assistance, and quick handling of large amounts of data.

Like XGBoost, LightGBM uses the Taylor expansion of the loss function to approximate the residual (which includes information from the first and second derivatives) and uses regularization terms to control the complexity of the model. Based on the traditional GBDT algorithm, LightGBM is optimized to avoid XGBoost's shortcomings and speed up GBDT model training without compromising accuracy.

3.6 CatBoost

CatBoost is a kind of Boosting algorithm and an improved implementation under the GBDT algorithm framework. Gradient Boosted Decision Tree (GBDT) implementation for Supervised bringing two innovations: Ordered Target Statistics and Ordered Boosting (Hancock & Khoshgoftaar, 2020).

The primary challenge is to handle categorical features in an efficient and rational manner. In addition, CatBoost addresses the issues of gradient bias and prediction shift, which lowers the likelihood of overfitting and enhances the algorithm's accuracy and capacity for generalization.

In contrast to XGBoost and LightGBM, CatBoost incorporates a novel technique that converts categorical information into numerical characteristics automatically: To create a new numerical feature, first create some data on categorical features, determine their frequency, and then adjust hyperparameters. In order to use the interactions between features and significantly enhance the feature dimension, Catboost additionally employs composite class features.

# 4. Model Implementation

4.1 Experimental introduction

All the data used by the model has been preprocessed by the previous data, and the data used by each model is consistent, and the size of the data partition is consistent. The cross-validation methods used in the model are all the 50-fold cross-validation methods. The data sets are divided into 80% training set and 20% verification set. See the following figure.

图片包含 图示

描述已自动生成

Fig.10 Data set partitioning operation code

Seven models including decision tree, AdaBoost, Bagging, RandomForest, xgboost, lightgbm and catboost will be used to train the same data, and AUC index will be used to evaluate each trained model. The parameter selection method of each model will use the grid search method to select the optimal parameters.

4.2 Decision tree

The decision tree is a predictive model used in machine learning that illustrates the mapping relationship between object attributes and object values. We use sklearn API to call the tree method to realize the decision tree, and use grid search method to select the optimal parameters. The grid search method can be seen in the following figure. The optimal parameters of the model can be selected, but the disadvantage is that the computer performance is very tested and the running time will be very long.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

Fig.11 Decision tree implementation operation code

The running result of the decision tree can be seen in the figure below. We can check the training report of the model, and we can find that the accuracy rate of the model is 80.14%, which has achieved a good effect.

表格

描述已自动生成

Fig.12 Decision tree run result confusion matrix

Since the label of this dataset is unbalanced, the AUC index needs to be used to evaluate the model. As shown in the figure below, the AUC of the model training effect of the basic decision tree is 70.66%.

图表

描述已自动生成

Fig.13 Decision tree model training effect AUC

In this way, we can view the top ten features ranked by importance in the decision tree model, as shown in the figure below. They are subGrade, grade, issueDateDT, term, homeOwnership, dti, annualIncome, installment, ficoRangeLow, interestRate.

图片包含 图表

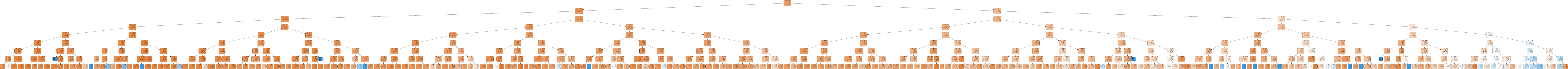
描述已自动生成

Fig.14 The top ten feature and partial structure trees in the decision tree model

4.3 Bagging

Bagging is a method that combines many models to minimize generalization mistakes. The primary concept involves training many models independently, after which each model casts a vote on the test sample's outcome. This is an illustration of model-averaging, a popular machine learning technique. An integrated approach is the process for implementing this strategy.

We take the above decision tree as the bottom tree and use the grid search method to search for the best model parameters of Bagging bagging method. The model report can be seen in the figure below. We can find the decision tree model based on Bagging. The accuracy is higher than that of the basic decision tree model, which is 80.239%.

表格

描述已自动生成

Fig.15 Bagging run result confusion matrix

Looking at the AUC index of this model, we can find that the AUC index of this model is also higher than that of the basic decision tree model, which is 71.46%.

图表

描述已自动生成

Fig.16 Bagging model training effect AUC

4.4 Random forest

Multiple decision trees are used by the random forest technique to train, categorize, and predict samples. In addition to classifying data, it may assess each variable's significance and determine how each one fits into the overall categorization. One popular ensemble learning technique that does not rely on weak learners is random forest, which is a component of the ensemble learning algorithm.

In this experiment, sklearn API is used to call the method to realize the decision tree, and grid search method is used to search for the best model parameters. The specific model training results can be seen in the figure below. We can find that the Bagging method mentioned above and the integration method have similar effects on this data set. The accuracy of the random forest model is 80.295%.

表格

描述已自动生成

Fig.17 Random forest run result confusion matrix

Look at the AUC of this random forest, which is 70.91%. We can find that the model accuracy of random forest is higher than that of Bagging method, but the AUC index of the model is lower than that of Bagging method. This is why sometimes the model evaluation should not only look at the model accuracy, but also select the model evaluation index based on the actual situation of the data.

图表

描述已自动生成

Fig.18 Random forest model training effect AUC

It can view the ranking of important features output by the model and select the top 10 important features, as shown in the figure below. Their decibels are dti, interestRate, revolBal, employmentTitle, revolUtil, postCode, annualIncome, installment, issueDateDT, subGrade, We can find that the top 10 important feature indicators of the model are partly the same as those of the decision tree.

背景图案

描述已自动生成

Fig.19 Rank the top ten important features of the output of the random forest model

4.5 XGBoost

With a speed increase of over 10 times over the usual toolkit, xgboost is the fastest and best open source boosted tree toolkit currently on the market. It is a massively parallel boosted tree tool. eXtreme Gradient Boosting, or xgboost, is a distributed gradient boosting library that has been tuned for efficiency, adaptability, and portability.

In this experiment, xgboost toolkit is called directly to construct the function to implement the model. The Xgboost model has a lot of parameters, so using grid search to find the best parameters can be very, very slow. The specific parameters of the Xgboost model can be seen in the following figure

散点图

低可信度描述已自动生成

Fig.20 XGBoost implements code operations

The model is used to train the same data, and the training effect of the model is output. The details can be seen in the following figure, which shows the change process of AUC during the training process of the model.

一些文字和图案

描述已自动生成

Fig.21 Changes of AUC during XGBoost model training

We can view the AUC of the model training results of a certain fold, which is 73.41%. We can find that the AUC obtained by this model is better than that of the previous models.

图表, 折线图

描述已自动生成

Fig.22 XGBoost model training effect AUC

4.6 Lightgbm

The GBDT algorithm can be implemented efficiently in parallel with the help of the LightGBM (Light Gradient Boosting Machine) framework. This algorithm has several benefits, including distributed assistance, faster training, less memory usage, improved accuracy, and quick processing of large amounts of data.

This model is implemented by directly calling the API of lightgbm toolkit and using this method. The parameters of this model are also very large, which can be seen in the following figure.

散点图

描述已自动生成

Fig.23 Lightgbm implements code operations

The process diagram of the model's half-fold cross-validation training can be seen as follows.

文本

描述已自动生成

Fig.24 Changes of AUC during Lightgbm model training

The AUC obtained by Lgb model training is 73.33%, and the effect is slightly worse than that of lgb model, partly because the parameters are not adjusted to the optimal. But both are better than the previous model such as the decision tree.

图表, 折线图

描述已自动生成

Fig.25 Lightgbm model training effect AUC

We can view the structure tree of the model, as shown in the figure below.

图示, 示意图

描述已自动生成

Fig. 26 Structure tree of Lightgbm model

4.7 CatBoost

A member of the Boosting algorithm family, CatBoost is a machine learning library that was made publicly available in 2017 by Yandex, a Russian company. The three popular artifacts of GBDT, CatBoost and XGBoost, are enhanced versions of the GBDT algorithm. They are part of LightGBM. A symmetric decision tree-based learner with fewer parameters, support for class-based variables, and good accuracy is the foundation of the GBDT framework CatBoost.

This experiment uses the API of catboost toolkit to implement catboost model, which has relatively few parameters, as shown in the following figure

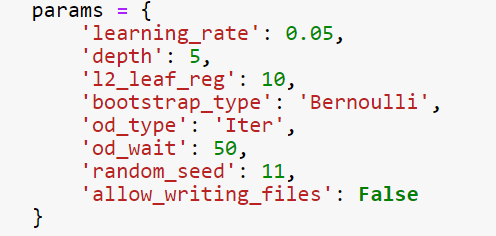


Fig.27 CatBoost implements operational code



文本, 表格

描述已自动生成

Fig.28 Changes of AUC during CatBoost model training

The total AUC of CatBoost five-fold cross-validation training process is 74.70%, 74.77%, 74.88, 74.68%, 74.68%, respectively. We can find that the maximum AUC of this model is 74.88%, which is the highest among the seven models tested at present, and the average AUC of this model is 74.76%, which is higher than the AUC of the previous model.

图表, 折线图

描述已自动生成

图表, 折线图

描述已自动生成

Fig.29 catboost model training effect AUC

# 5. Conclusion

Based on this analysis, we used the training data of 6 models, Decision tree, Bagging, RandomForest, XGBoost, Lightgbm and CatBoost, and obtained the best results of each model as shown in the figure below. We got the best results using the CatBoost model with a maximum AUC of 74.88%.

Fig.30 Each model evaluation renderings

Because there are numerous possible financial hazards, predicting the frequency of loan defaults is a crucial task. Machine learning techniques for credit rating can assist borrowers and banks in efficiently managing these risks. By combining information about the lender's assets, credit history, use of social media platforms like Facebook and Whats App, online payments like Apple Pay and Ebay, personal, cultural, and religious background, social network, and demographic traits like education experience, among other things, it enables banks to create a credit score model of the lender. It might offer information that could be utilized to forecast a borrower's creditworthiness and capacity for repayment.

This study successfully addresses three previously identified gaps in mortgage risk analysis. Firstly, the traditional machine learning model is applied to summarize some characteristics that affect loan risk. It is worth noting that the influence of the anonymity characteristics of lender behavior on loan risk is also revealed, providing valuable insights. Second, we have demonstrated the feasibility of training a mortgage risk prediction model with the same precision using a limited data set. This finding is important because it shows that future studies can efficiently use subsampled data for model training, saving a lot of time. Therefore, a combination of prediction and explainable models can be used in other forecasting problems([Wakjira et al., 2022](https://www.sciencedirect.com/science/article/pii/S2666764923000218" \l "bib23)). In the future, other explainable models can also be applied to estimate the overall influence of the features used ([Lim et al., 2021](https://www.sciencedirect.com/science/article/pii/S2666764923000218" \l "bib13)).

|  |
| --- |
| **Bibliography** |

Chang, Y., Chang, K., Chu, H., & Tong, L. (2016). Establishing decision tree-based short-term default credit risk assessment models. *Communications in Statistics - Theory and Methods*, *45*(23), 6803–6815. https://doi.org/10.1080/03610926.2014.968730

Ashraf, R. (2024). Bank Customer Churn Prediction Using Machine Learning Framework. *SSRN*. https://doi.org/10.2139/ssrn.4757728

Zhu, L., Qiu, D., Ergu, D., Cai, Y., & Liu, K. (2019b). A study on predicting loan default based on the random forest algorithm. *Procedia Computer Science*, *162*, 503–513. https://doi.org/10.1016/j.procs.2019.12.017

Cheng, Y. (2021b). Research on Credit Strategy based on XGBOOST Algorithm and Optimization Problem. *Journal of Physics: Conference Series*, *1865*(4), 042137. https://doi.org/10.1088/1742-6596/1865/4/042137

Mishra, S. P., Sarkar, U. K., Taraphder, S., Datta, S., Swain, D. P., Saikhom, R., Panda, S., & Laishram, M. (2017b). Multivariate statistical data analysis- Principal Component Analysis (PCA) -. *International Journal of Livestock Research*, *7*(5), 60–78. https://www.bibliomed.org/?mno=261590

Hancock, J., & Khoshgoftaar, T. M. (2020b). CatBoost for Big Data: an Interdisciplinary Review. *Research Square (Research Square)*. https://doi.org/10.21203/rs.3.rs-54646/v2

Wakjira, T. G., Ibrahim, M., Ebead, U., & Alam, M. S. (2022). Explainable machine learning model and reliability analysis for flexural capacity prediction of RC beams strengthened in flexure with FRCM. *Engineering Structures*, *255*, 113903. https://doi.org/10.1016/j.engstruct.2022.113903

Zhu, X., Chu, Q., Song, X., Hu, P., & Peng, L. (2023). Explainable prediction of loan default based on machine learning models. *Data Science and Management*, *6*(3), 123–133. https://doi.org/10.1016/j.dsm.2023.04.003

|  |
| --- |
| **Appendix** |

Here are the code and dataset urls stored on Github:

https://github.coventry.ac.uk/liy377/Individual-Project-of-AI---14384024---Yehan-Li