ISSN 2616-5775 Vol. 7, Issue 10: 60-66, DOI: 10.25236/AJCIS.2024.071009

Credit Default Probability Prediction Model Based on XGBoost Algorithm

Hong Rao*, Chenhao Wei

Sun Yueqi Honors College, China University of Mining and Technology, Xvzhou, 221116, China
**Corresponding author: 18968502317@163.com*

Abstract: *With the rapid economic development of China, the importance of credit consumption methods in China's economy and people's daily lives has become increasingly prominent. Based on the GiveMeSomeCredit dataset, this paper constructs an XGBoost model to conduct in-depth predictive analysis on credit default probability issues. This paper first performs meticulous data cleaning and missing value handling on the dataset, and divides the dataset to prepare for model training. Subsequently, the XGBoost model is constructed and trained, and parameter optimization is further carried out during this process. Finally, the model's performance is evaluated using key evaluation indicators such as Accuracy, Precision, Recall, and AUC, and it is compared with the random forest model and logistic regression model. The results show that the XGBoost model performs better. It can be seen that the XGBoost model has high application value in credit default probability prediction.*

Keywords: *Credit Default, Risk Prediction, XGBoost Model*

1. Introduction

As China's economic system continues to improve, its bond market has gradually become more mature and stable, transitioning from a rapid growth phase to a steady upward trend. Against this backdrop, it has become an urgent issue for financial professionals to accurately select reliable and trustworthy borrowers. Machine learning models are an excellent choice for studying such prediction problems.

In the current research landscape, there have been numerous achievements related to credit default probability prediction. For instance, Wang Jianhai used combined feature selection and LightGBM to construct a credit default prediction model [1], while Zheng Cheng conducted research on bond defaults based on the CNN-SVM model [2]. Previous researchers have proposed various solutions to this problem based on their own studies, but each has its own shortcomings and limitations.

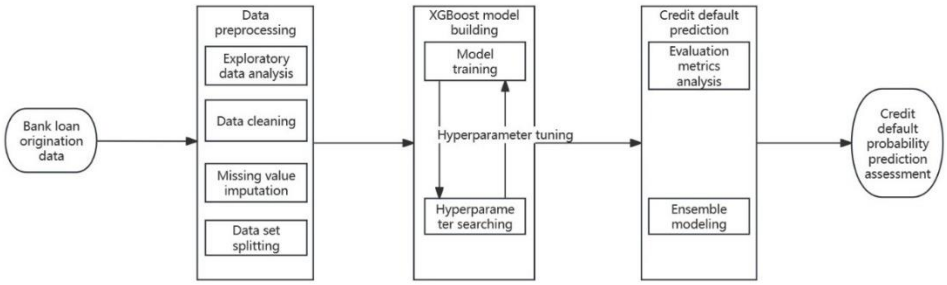

Figure 1: Credit Default Probability Prediction Method Based on XGBoost

Therefore, this paper aims to further delve into the issue of credit default probability prediction by utilizing the XGBoost (Extreme Gradient Boosting) model, aiming to build upon previous research and achieve further advancements. Employing the GiveMeSomeCredit dataset from Kaggle, this paper expects to achieve relatively accurate prediction results for credit default probability. Based on the overall construction using the XGBoost algorithm, this paper compares the prediction results with a series of other algorithmic models, including logistic regression, random forest, and support vector machines. Through quantitative results, the unique superiority of the XGBoost algorithm in addressing this problem is objectively and directly demonstrated.

ISSN 2616-5775 Vol. 7, Issue 10: 60-66, DOI: 10.25236/AJCIS.2024.071009

The credit default probability prediction method proposed in this paper based on XGBoost is illustrated in Figure 1.

2. Literature Review: Credit Default

Credit default is an increasingly common and vexing issue in the financial lending industry. Although credit ratings, which provide an overall evaluation of a borrower's ability and willingness to fulfill related contracts and economic commitments, are currently in use, cases of debt defaults among high-credit-rated borrowers still occur occasionally. This has not only caused lenders to doubt the effectiveness of such credit rating systems but has also spurred the demand for a more accurate credit default probability prediction system. Therefore, this paper aims to establish a more reliable credit default probability prediction system based on the XGBoost algorithm.

In the application of XGBoost in credit evaluation and other fields, there have been numerous research achievements. For instance, Ni Xu established a research system on credit evaluation of new agricultural business entities in China based on the XGBoost algorithm [3]. Zhou Qing'an conducted a study on personal online loan credit evaluation using a genetic XGBoost model [4]. Zhou Rongxi and his team also developed a credit default prediction model based on the XGBoost algorithm [5]. Compared to other models, their established models have demonstrated superior performance. The numerous model constructions and studies based on the XGBoost algorithm have shown the significant application value of this model in addressing this particular problem.

In summary, the XGBoost algorithm holds tremendous application prospects in credit default probability prediction. However, to fully leverage its potential, it is necessary to address challenges such as extensive parameter tuning and potential overfitting on certain datasets. Addressing these issues can further enhance the reliability of the algorithm.

3. Data preprocessing

3.1 An Introduction to Data

This article utilizes the 'givemesomecredit' dataset from Kaggle, which comprises information on a total of 250,000 borrowers.

The specific meanings of each field in the dataset are shown in Table 1:

Table 1: List of Fields and Their Corresponding Meanings

SeriousDIqin2yrs	Persons with overdue payments of 90 days or more
RevolvingUtilizationOfUnsecuredLines	Available credit line ratio
Age	Borrower's age
NumberOfTime30-59DaysPastDueNotWorse	Number of overdue payments for 30-59 days
DebtRatio	Debt-to-income ratio
MonthlyIncome	Monthly income
NumberOfOpenCreditLinesAndLoans	Number of credit facilities
NumberOfTimes90DaysLate	Number of times the borrower has defaulted for 90 days or more
NumberRealEstataLoansOrLines	Number of fixed asset loans
NumberOfTime60-89DaysPastDueNotWorse	Number of overdue payments for 60-89 days
NumberOfDependents	Number of dependents

3.2 Initial exploration of data

After reading in the data, in order to facilitate a clearer observation of the characteristics of the data, we draw distribution plots, box plots, and relationship heatmaps of the data fields.

The distribution chart of data fields is shown in Figure 2:

ISSN 2616-5775 Vol. 7, Issue 10: 60-66, DOI: 10.25236/AJCIS.2024.071009

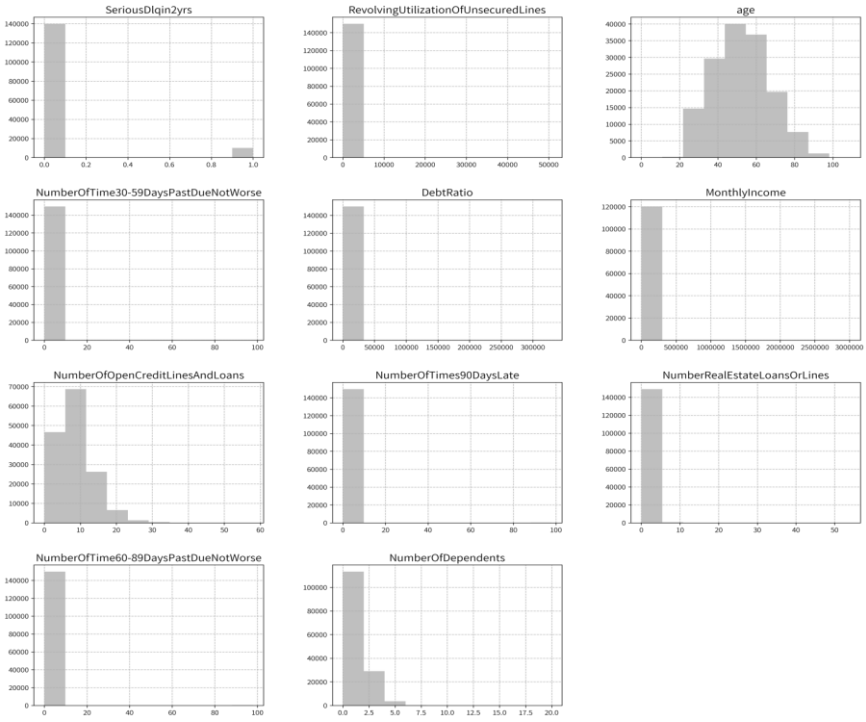

Figure 2: Distribution Chart of Data Fields

The box plot of data field is shown in Figure 3.

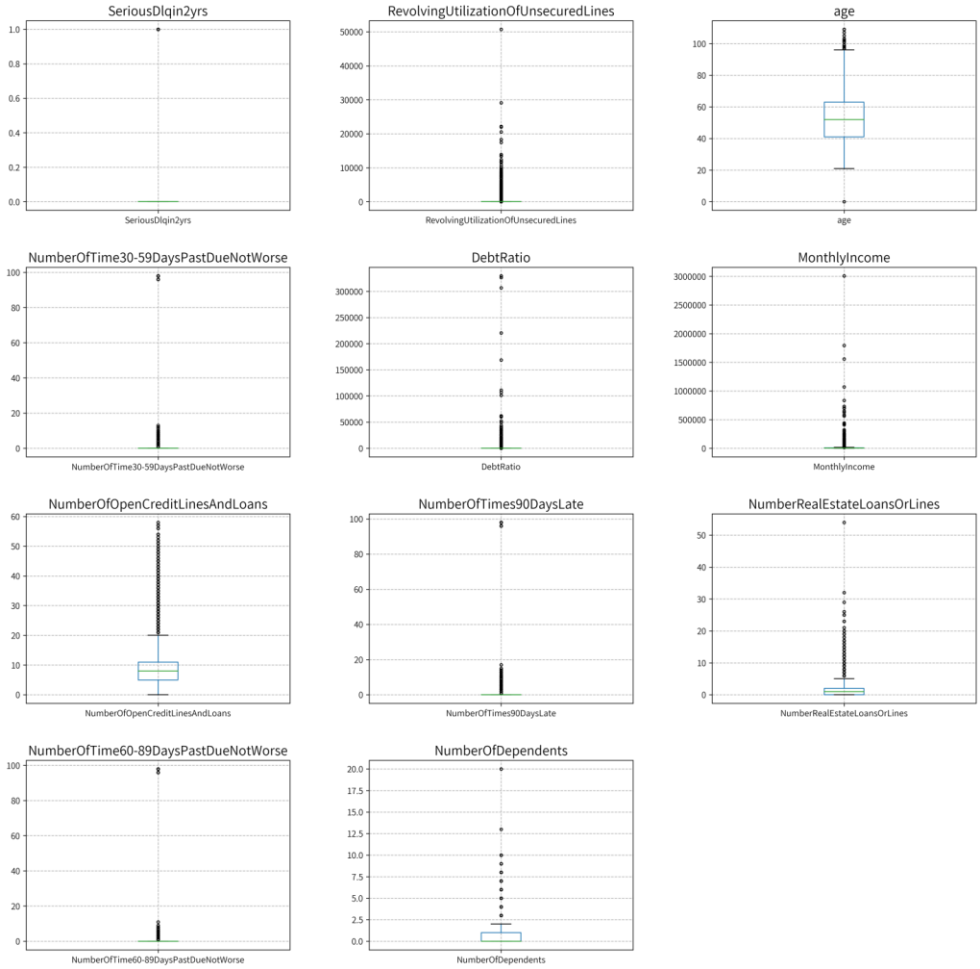

Figure 3: Box plot of data field

ISSN 2616-5775 Vol. 7, Issue 10: 60-66, DOI: 10.25236/AJCIS.2024.071009

The heatmap of data field is shown in Figure 4.

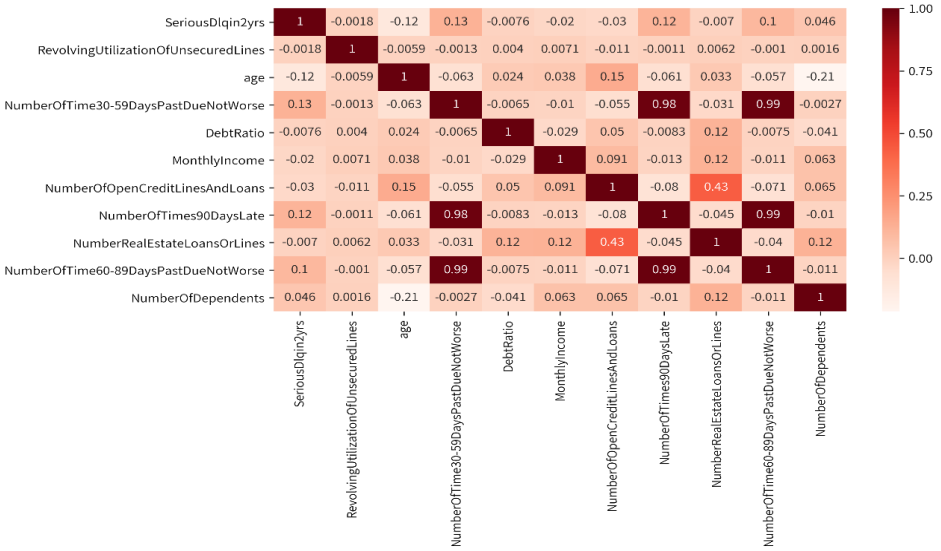

Figure 4: Heatmap of Data Field

Through Figure 2, we can see that most of the fields are obviously skewed, and skewness correction should be considered in subsequent modeling.

Through Figure 3, we can observe the outliers in each field, such as "Number of overdue loans for 30-59 days," "Debt ratio," "Monthly income," "Number of loans overdue for more than 90 days," "Number of fixed asset loans," and "Number of overdue loans for 60-89 days," which are numerous and difficult to observe the data distribution. Based on this, further preprocessing of the data will be conducted.

Through Figure 4, we can roughly understand the strength of the relationship between various data. We found that the three fields related to the number of overdue loans have a very high collinearity, so we will consider removing the collinearity in subsequent processing.

3.3 Data preprocessing

In the process of data preprocessing, this paper first constructs functions for handling outliers and obvious errors, followed by functions for removing collinearity, handling missing values, and resampling. After the initial processing of the data, the dataset used in this paper is divided into a training set and a validation set in an 8:2 ratio for model testing. Additionally, a stratified K-fold cross-splitter and grid search algorithm are constructed to optimize parameters in order to obtain the optimal prediction model. Finally, a function is constructed to evaluate the performance of the classification model.

4. Model establishment

4.1 Introduction to XGBoost Algorithm

XGBoost (Extreme Gradient Boosting) is an improved integrated model based on the boosting framework and CART regression trees. It transforms a batch of weak classifiers into strong classifiers through iterative computation. This algorithm adopts the integration idea, supports parallel computing with multiple threads, solves the minimum loss function through second-order Taylor expansion to determine the splitting nodes, and constructs the final model. It performs well in both classification and regression problems.

Here is a brief introduction to the principles of the XGBoost algorithm:

First, we define the XGBoost model as an additive model derived from the Boosting idea:

$$\hat{Y}_i = \sum_q^Q f_q(x_i) \tag{1}$$

ISSN 2616-5775 Vol. 7, Issue 10: 60-66, DOI: 10.25236/AJCIS.2024.071009

In this context, we regard f_q as a single tree, and the model consists of a total of q trees.

We further define the objective function of the model as follows:

$$L(\phi) = \sum L(y_i, \hat{y}_i) + \sum_n \Omega(f_q) \tag{2}$$

The first term is the loss function (Lost Function), where y_i represents the true value and $\hat{y}_i$ represents the predicted value, with i being the sample number. The second term is the regularization term $\Omega(f_q) = \gamma T + \frac{1}{2}\lambda\|\omega\|^2$, also known as the penalty term. In this term, T represents the number of leaf nodes, and ω represents the values on the leaf nodes. The purpose of adding the regularization term is to control the complexity of the model, striking a balance between the complexity of the model and its performance.

Based on the objective function, the forward stagewise algorithm is employed to solve for the decision tree in the current state f_t:

$$L^{(t)} = \sum_i L(y_I, \hat{y}_i^{(t-1)} + f_i(x_i)) + \Omega(f_t) \tag{3}$$

Since the results of the first t-1 rounds are known, we can optimize the above equation in the current state of the t-th round to obtain f_t.Among them, since the regularization term of the first t-1 rounds is a constant term and has no effect on the optimization result, it is removed.

To solve equation (3), we perform a second-order Taylor expansion of $L^{(t)}$ and then carry out a series of substitution calculations to obtain the score function that evaluates the quality of the tree structure:

$$\text{Score} \cong -\frac{1}{2}\Sigma_{j=1}^{T} \frac{(\Sigma_{i\in I_j} g_i)^2}{\left(\sum_{i\in I_j} h_i + \lambda\right)^2} + \gamma T \tag{4}$$

The larger the value in equation (4), the better. Based on the resulting score function, we can obtain the criterion for dividing nodes, which is the difference between the score function after splitting a node and the score function before splitting the current node.

$$L_{split} = \frac{1}{2}\left[\frac{(\Sigma_{i\in I_L} g_i)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{(\Sigma_{i\in I_R} g_i)^2}{\sum_{i\in I_R} h_i + \lambda} - \frac{(\Sigma_{i\in I_j} g_i)^2}{\sum_{i\in I_j} h_i + \lambda}\right] - \gamma \tag{5}$$

In equation (5), I_L represents all the sample points that fall into the left node after splitting a certain node, and I_R represents all the sample points that fall into the right node. Therefore, the model is built by splitting at the node with the maximum L_{split} value. The XGboost algorithm can perform parallel computation when selecting the best splitting point and enumerating.

4.2 Basic parameter setting

(1) Objective: The objective function. Set to 'binary:logistic' to indicate the use of logistic regression for binary classification tasks as its objective function.

(2) n_jobs: The number of processors to use. Set to -1 to utilize all available processors.

(3) booster: The boosting algorithm to use. Set to 'gbtree' to indicate the use of gradient boosting decision trees.

(4) n_estimators: The number of boosting trees. Set to 1000 to indicate the use of 1000 boosting trees.

(5) learning_rate: The learning rate. Set to 0.01 to control the step size of model updates.

(6) max_depth: The maximum depth of the tree, controlling the complexity of the tree. Set to [6,9].

(7) subsample: The proportion of samples to use when training each tree, helping to prevent overfitting. Set to [0.6,0.9].

(8) colsample_bytree: The proportion of features to use when training each tree, helping to reduce correlation. Set to [0.5,0.6].

(9) reg_alpha: The penalty coefficient for the L1 regularization term, helping to prevent overfitting. Set to [0.05,0.1].

ISSN 2616-5775 Vol. 7, Issue 10: 60-66, DOI: 10.25236/AJCIS.2024.071009

4.3 Parameter Search

Using the GridSearchCV function to perform grid search, we train the XGBoost classifier and evaluate its performance for each parameter combination. Finally, we select the parameter combination with the highest ROCAUC score from the cross-validation folds and re-fit the model on the entire training dataset using the optimal parameter combination to achieve as accurate predictions as possible.

5. Solving the XGBoost Model

5.1 Selection of Evaluation Parameters

(1) Accuracy: The ratio of correctly predicted samples to the total number of samples.

(2) Precision: The proportion of actual positive samples among those predicted as positive.

(3) Recall: The proportion of predicted positive samples among the actual positive samples.

(4) AUC (Area Under the ROC Curve): The area under the Receiver Operating Characteristic (ROC) curve, which measures the model's ability to distinguish between positive and negative classes.

(5) Average ROC AUC Score.

5.2 XGBoost Model Solution Results

The evaluation index table for the XGBoost model is shown in Table 2.

Table 2: Evaluation Metrics for the XGBoost Model

AverageROCAUC=0.8686	Accuracy score	Precision score	Recall score	AUC
train	0.8258	0.8345	0.8104	0.8257
test	0.7853	0.7968	0.7777	0.7855

Overall, the model exhibits good performance on both the training and testing sets.

Accuracy: Both the training set and the testing set have high accuracy rates, at 81.93% and 78.78% respectively. This indicates that the model is able to correctly classify the majority of samples.

Precision and Recall: The precision and recall rates for both the training set and the testing set are also quite high, which suggests that the model performs well in identifying positive and negative samples.

AUC: The AUC values for both the training set and the testing set are close to 0.8, indicating that the model has good ability to distinguish between positive and negative samples.

Average ROC AUC Score: With an average score above 0.86, the model demonstrates a strong capability in distinguishing positive and negative samples, making it suitable for practical applications.

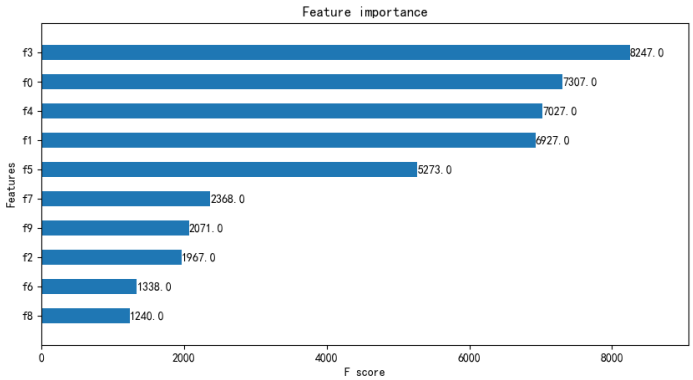

Figure 5: The Importance of Different Fields in the XGBoost Model

It is worth noting that we observe a slight decrease in performance on the testing set compared to the training set. This is primarily due to the fact that the testing set contains new data that was not used for training the model, so a certain degree of performance decline is normal. However, in this model, the

ISSN 2616-5775 Vol. 7, Issue 10: 60-66, DOI: 10.25236/AJCIS.2024.071009

decline in testing set performance is not significant, indicating good generalization ability of the model. Figure 5 demonstrates the importance of different fields in the XGBoost model.

5.3 Comparison of XGBoost Model with Other Models

5.3.1 Logistic Regression Model

The evaluation metrics table for the logistic regression model is shown in Table 3.

Table 3: Evaluation Metrics for the Logistic Regression Model

AverageROCAUC=0.8564	Accuracyscore	Precisionscore	Recallscore	AUC
train	0.7757	0.8022	0.7298	0.7755
test	0.7691	0.8108	0.7103	0.7700

In general, the logistic regression model does not perform well on either the training set or the testing set, and its performance is inferior compared to the XGBoost model.

5.3.2 Random Forest Model

The evaluation metrics table for the Random Forest model is shown in Table 4.

Table 4: Evaluation Metrics for the Random Forest Model

AverageROCAUC=0.8630	Accuracyscore	Precisionscore	Recallscore	AUC
train	0.8122	0.8145	0.8068	0.8122
test	0.7806	0.7893	0.7737	0.7807

As can be seen from the above figure, the prediction performance of the Random Forest model is relatively close to that of the XGBoost model, but there is still a certain gap in various aspects.

5.4 Experiment Conclusion

Overall, the model based on XGBoost outperforms the other two models, and its prediction results are relatively accurate. However, we found that although the model performs well overall, the average ROC AUC value cannot be further improved and remains stable at around 0.86. This indicates that there is still considerable room for optimization in the model.

6. Conclusion and Future Prospects

Based on the selected dataset, this paper constructed a model and, through data analysis and model comparison throughout the experimental process, we found that the XGBoost model is an excellent choice for studying credit default probability issues. It can achieve accurate predictions and inferences, demonstrating excellent practical application value. Credit industry professionals can greatly enhance the accuracy of credit limit predictions for borrowers by constructing credit default probability prediction models using XGBoost algorithms. This, in turn, improves the accuracy of lending decisions and helps reduce the probability of credit default incidents.

In the future, we plan to further optimize the model parameters and integrate multiple models to jointly address this issue in order to obtain better prediction results.

References

[1] Wang Jianhai. Design and Implementation of Credit Default Prediction Model Based on Combined Feature Selection and LightGBM [D]. Southeast University, 2022.
[2] Zheng Cheng. Research on Bond Default Prediction Based on CNN-SVM Model [D]. Zhejiang University of Finance and Economics, 2024.
[3] Ni Xu. Research on Credit Evaluation of New Agricultural Business Entities in China [D]. Chinese Academy of Agricultural Sciences, 2019.
[4] Zhou Qing'an. Research on Personal Online Loan Credit Evaluation Based on Genetic XGBoost Model [D]. Jiangxi University of Finance and Economics, 2020.
[5] Zhou Rongxi, Peng Hang, Li Xinyu, et al. Credit Bond Default Prediction Model Based on XGBoost Algorithm [J]. Bonds, 2019.