Journal of
Innovations in Economics & Management
journal homepage: www.journal-iem.online

Paper Type: Original Article

Telecom bank card fraud prediction model based on machine learning

Wenyi Huang[1,a#], Yeyang Chen [2,b#], Yiheng Song[3,c#], Minghao Liu[4,d#], Zihan Jin[5,e#], Qiongya Tang [6,f#*]

[1] Department of education, Luoding Polytechnic, Yunfu, China

[2] School of Data Science and Artificial Intelligence, Wenzhou University of Technology, Wenzhou, China

[3] School of Electric Power, SHENYANG INSTITUTE OF ENGINEERING, Shenyang, China

[4] College of Water&Architectural Engineering, Shihezi University, Shihezi, China

[5] School of Computer Science, Shenyang Aerospace University, Liaoning Province, Shenyang,China

[6] Guangzhou, China

#These authors contributed equally to this work and should be considered co-first authors

**corresponding author*

Abstract

It is particularly important to identify and prevent telecom scams that trick victims into transferring funds through phone calls, Internet and text messages. Based on the collected data, a prediction model of telecom bank card fraud is established in this paper. In the analysis, we first checked the data missing through Pandas and missingno library, and conducted Pearson correlation analysis, and found that the ratio of transaction amount has a strong positive correlation with fraud. In terms of data preprocessing, outliers are defined and data are cleaned by box diagram, missing values are processed by KNN filling, and data is normalized by Yeo-Johnson transformation. Then, the importance of features is calculated by random forest and GBDT, and the features with greater influence are selected. In the model training, XGBoost, LightGBM and CatBoost integrated learning algorithms were selected, and the optimal model configuration was obtained through parameter optimization, and finally integrated into BaggingClassifier. The model performance evaluation shows that the prediction accuracy of the model established in this paper is up to 99.99%.

Keywords: Telecom bank card fraud, data preprocessing, feature engineering, integrated learning algorithms, model performance evaluation, security measures

1 | Introduction

Telecom bank card fraud is a kind of criminal behavior carried out by telephone, network and other ways, scammers often impersonate regular institutions or individuals, and trick victims to transfer or make money. Although the government has increased its crackdown and telecom fraud cases have been suppressed, the situation is still grim, especially the brush single rebate and false investment and financial fraud are the most common. In order to gain an in-depth understanding of the patterns and effects of telecom bank card fraud, this paper analyzes 1 million telecom bank card transaction data with the aim of building predictive models through machine learning methods to identify and prevent telecom bank card fraud. We expect that the model can not only effectively predict fraud, but also provide valuable preventive suggestions for public security departments, banks and the public to improve property security.

2 | Data preprocessing

2.1 | Missing value processing

The missing values are first viewed through the Pandas and missingno libraries[1], as shown in Figure 1.

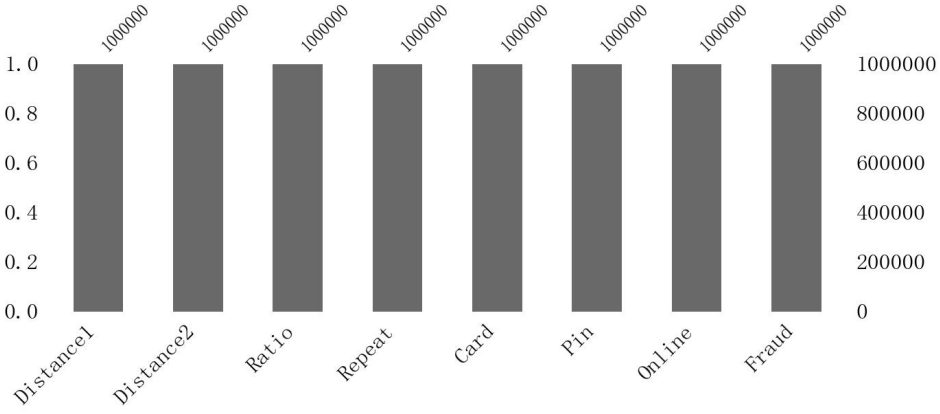


Figure 1. Missing value view

As can be seen from Figure 1, there are no missing values in the data set, and then we will conduct correlation analysis.

2.2 | Pearson correlation analysis

In this paper, Pearson correlation coefficient is used to measure the correlation among indicators, and corr method in Pandas library is used to calculate the Pearson correlation coefficient among indicators in the data set. This analysis is often used to understand the strength of the linear relationship between different variables. The calculation formula is as follows[2].

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}} \quad (1)$$

Where: n is the number of samples, x and y are the two variables to be compared, ∑xy is the sum of the product of variables x and y, ∑x and ∑y are the sum of variables x and y, respectively, and $\sum x^2$ and $\sum y^2$ are the sum of squares of variables x and y, respectively.

The correlation coefficient is a number between -1 and 1 that quantifies the strength and direction of the linear relationship between two variables. The closer the absolute value of the coefficient is to 1, the stronger the linear relationship between the two variables. A positive value indicates a positive correlation, i.e. an increase in one variable is usually accompanied by an increase in another variable; A negative value indicates a negative correlation, where an increase in one variable is accompanied by a decrease in the other.

data.corr() calculates the Pearson correlation coefficient. python is used to generate correlation_matrix, which is a matrix containing the correlation coefficients between all variable pairs[3]. This is then visualized as a heatmap by seaborn's heatmap function, making correlation analysis more intuitive. As shown in Figure 2.

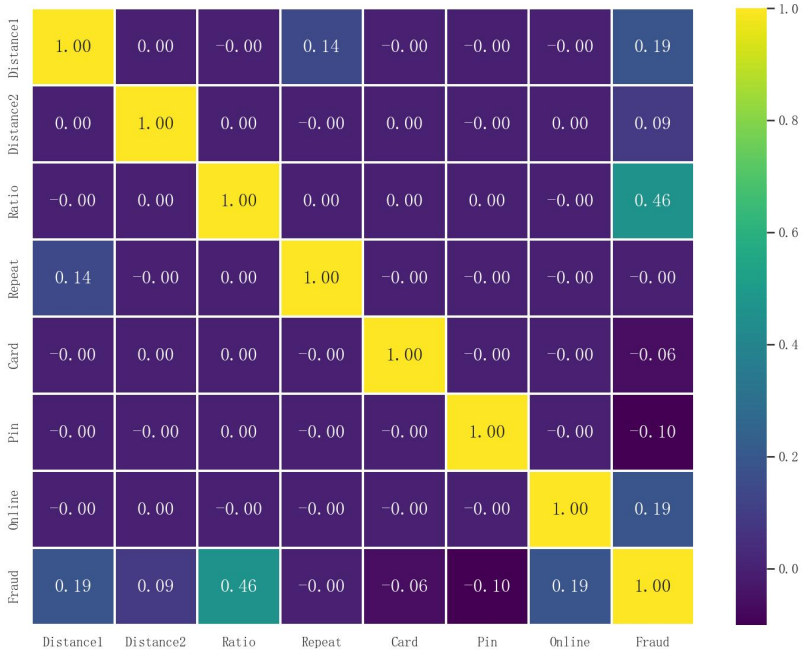


Figure 2 Pearson correlation heat map

As can be seen from Figure 2:

Ratio: shows a relatively strong positive correlation (0.46), which means that there is a certain correlation between the change ratio of transaction amount and telecom fraud. When this ratio changes significantly, it may be associated with fraudulent activity.

Distance1 and Online: Both have a correlation coefficient of 0.19 with telecom fraud, indicating that these variables may also be associated with fraudulent activity. Online transactions, in particular, can be exploited by fraudsters because of their anonymity and convenience.

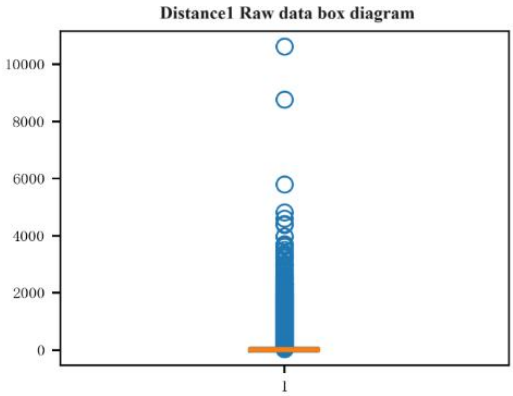
Repeat: The correlation coefficient is close to 0 (-0.00), indicating that there is little direct linear relationship between whether a card transfer is made at the same bank and telecom fraud. This means that this feature is not significant in predicting the occurrence of telecom fraud, regardless of whether the fraudulent transaction takes place at the same bank.

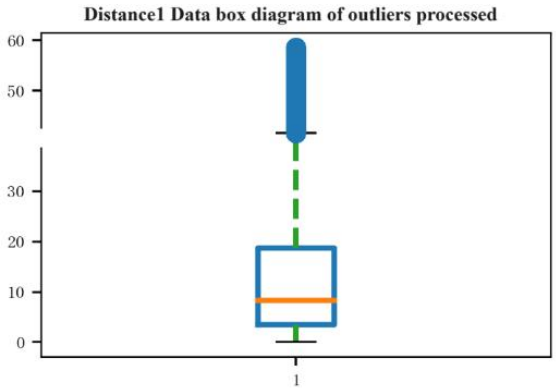
Online: As mentioned above, the correlation coefficient is 0.19. Although this correlation is not very strong, it still indicates that there is a correlation between online transactions and telecom fraud. The ease and reach of online transactions can make them a tool for fraudsters.

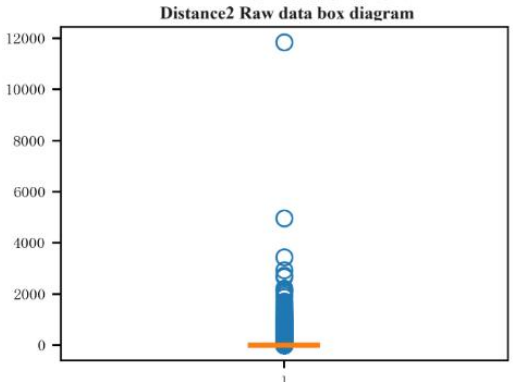
To sum up, Ratio and Online are more significantly correlated with telecom fraud than other indicators. The correlation of Repeat (whether the transaction is with the same bank or not) is almost negligible, suggesting that this feature is of little help in identifying fraud. Therefore, in the fourth question to establish the prediction model of telecom bank card fraud, we will calculate the importance of features again, combined with the correlation analysis of this question to further screen the features, and improve the prediction ability of the model.

2.3 | Outlier test

In this paper, the boxplot () function in the matplotlib library is used to draw the corresponding boxplot for each column data, and the number of outliers for each column data set is calculated according to the results of the boxplot. According to the box diagram principle, data less than Q1-1.5×IQR or greater than Q3+1.5×IQR in the data are defined as outliers, which need to be filtered out and only legitimate data is retained for subsequent analysis. Finally, use the filtered data to draw the box diagram again and save it to the output directory[4]. In this way, the processing of outliers can be automated, reducing manual intervention in data processing, and enabling more objective identification of outliers in the data and accurate calculation of the number of outliers. Figure 3 and Figure 4 show the original data box diagram of some features and the data box diagram after outlier processing.







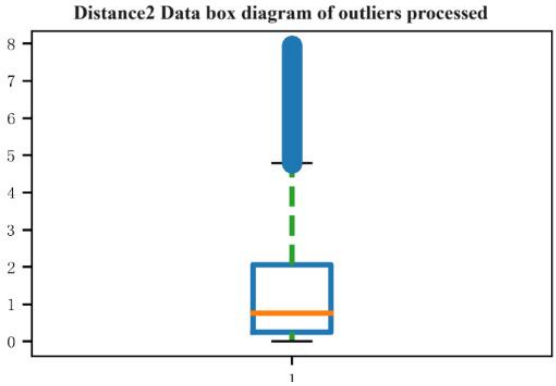


Figure 3. Raw data histogram distribution

Figure 4. Yeo-Johnson converted histogram

2.4 | KNN filling

After the outliers are removed, the missing values will be filled in. After careful analysis of the data set, it is decided to use the KNNImputer class to fill in the missing values. This class uses machine learning techniques based on the KNN algorithm to calculate missing values, and uses the similarity between "neighbors" to calculate relevant missing values. In this paper, the n_neighbors parameter in the KNN algorithm is set to 5, that is, for each missing value, we will fill it with the last 5 non-null values[5]. KNN algorithm uses K nearest neighbor non-missing samples similar to itself to calculate the missing value. Meanwhile, by considering the relationship between the distribution of other features and target features, the reliability and accuracy of missing value filling are improved. Therefore, compared with traditional methods such as fixed value filling and mean value filling, we believe that using KNN algorithm to fill missing values is more reliable and accurate.

3 | Feature engineering

3.1 | Yeo-Johnson

In this paper, if the absolute value of the skewness value of the data is greater than some very small threshold value of 0.05, it means that the data distribution type is not normal distribution

and needs to be transformed by Yeo-Johnson[6]. Figure 5 and Figure 6 show the original density histograms and the density histograms converted by Yeo-Johnson for seven features.

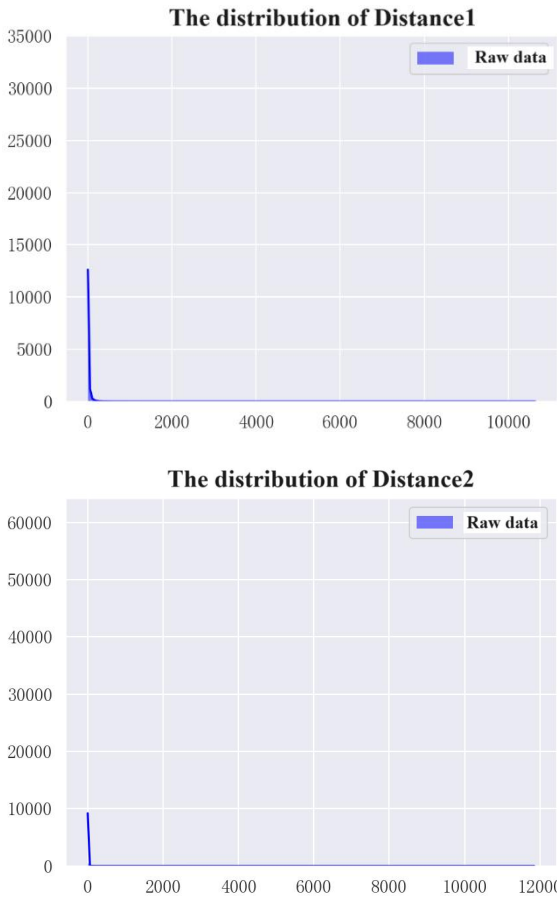


Figure 5. Raw data histogram distribution

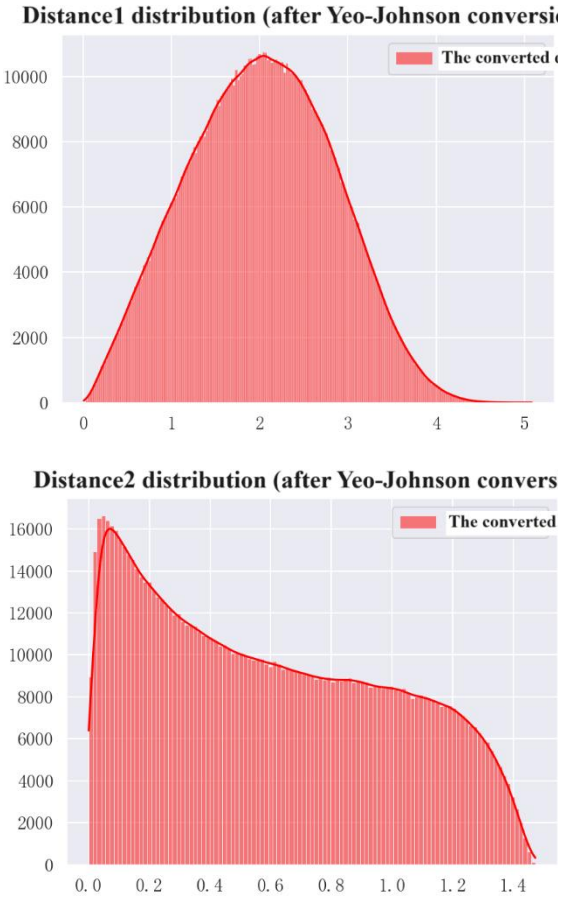


Figure 6. Yeo-Johnson converted histogram

3.2 | Random forests and the importance of GBDT features

In order to improve the prediction accuracy of the prediction model, it is necessary to screen out the features with large contribution to the target variables[7]. In this paper, random forest and GBDT are used to obtain the average feature importance to screen the features. The result is shown in Figure 7.

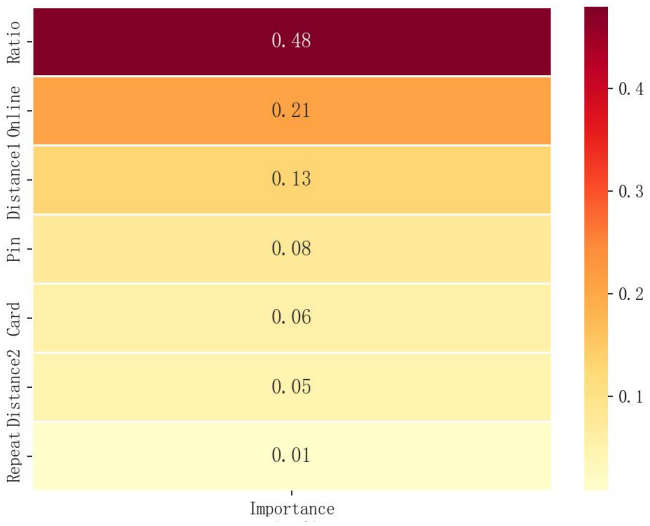


Figure 7. Feature importance

In summary, combined with the results of correlation analysis and feature importance analysis, we eliminate the feature "Repeat" and retain the remaining features to build a classification prediction model.

3.3 | Training set and test set partition

When processing the data set to prepare the prediction model of telecom bank card fraud, the feature normalization process is firstly carried out by MinMaxScaler. Normalization is scaling the range of data to a given minimum and maximum value, usually 0 to 1.

Next, with reference to relevant literature, there are two kinds of segmentation ratio of training set and test set for classification prediction model: 8:2 and 7:3, among which 8:2 is the most common, so this paper considers partitioning the data set into 80% training set and 20% test set. This can be done in Python with the train_test_split function, where test_size=0.2 is set to specify the proportion of the test set. In order to ensure that the segmented data can still maintain the proportion of various categories in the original data, a stratify sampling strategy (stratify=y) is adopted in this paper[8]. Stratified sampling ensures that the proportions of classes in the training and test sets are similar to those in the original data set, which is especially important for dealing with unbalanced data sets to avoid bias when training models due to too few samples in one class. Stratified sampling is shown in Figure 8.

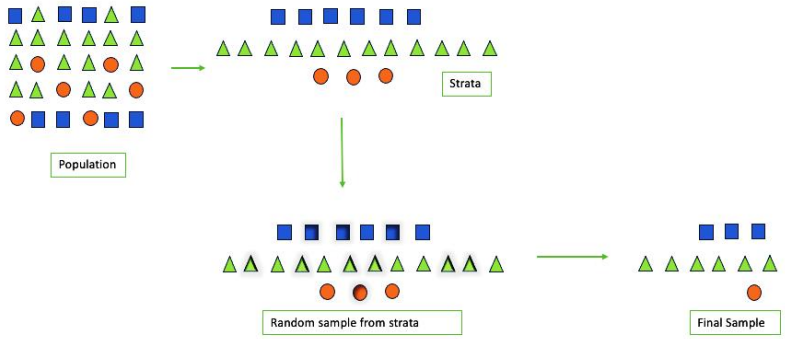


Figure 8. Stratified sampling flow chart

4 | A predictive integrated model of telecom bank card fraud

In order to make the model we established have a wide range of applicability and can be applied to real situations, Stacking is used in this paper to improve the prediction accuracy and robustness of the final model. Stacking is an integration technique that works by training another model (called a metamodel) on the output of multiple base models in the first layer. In this approach, you first train several different models (in this case, based on a different ensemble BaggingClassifier), and then train the metamodels with the output of those models (typically probabilistic predictions of the class) as a new set of features. In this article, the metamodel is a logistic regression, which makes a final decision by considering the probability of the base model's output. Using probability rather than class labels allows the metamodel to capture more information about uncertainty and make more refined decisions[9]. When implementing stacking, you first need to ensure that the different base models are diverse enough, because model diversity is the key to improving the stacking effect. The metamodel is then trained by using the predictions of the base model as input. This allows the metamodel to learn which base models are more reliable in a given situation, thereby optimizing the overall prediction.

4.1 | Model training and parameter optimization

In order to find the optimal configuration of each model, the RandomizedSearchCV method is used to optimize the random search of hyperparameters. Different from Grid Search, it randomly selects parameter combinations in the specified hyperparameter space for model training and evaluation. This randomness helps explore a wider parameter space while reducing computational costs. In order to ensure the validity of the evaluation and reduce the risk of overfitting the model, StratifiedKFold was used for cross-validation. This method divides the data into k subsets, each of which takes turns as a test set in model training and the rest as a training set, while keeping the proportion of categories in each subset the same as in the full data set, thus ensuring balanced training and validation.

In this paper, XGBoost, LightGBM and CatBoost all use the best parameters found by RandomizedSearchCV to train several independent models, and integrate these models by BaggingClassifier[10]. The final prediction result of the integrated model is based on the average or majority vote of all individual model predictions, which can significantly improve the generalization ability of the model to new data and reduce the risk of overfitting the model on a specific sample. FIG. 9 is a schematic diagram of the optimal hyperparameters of the three models corresponding to different seed models.

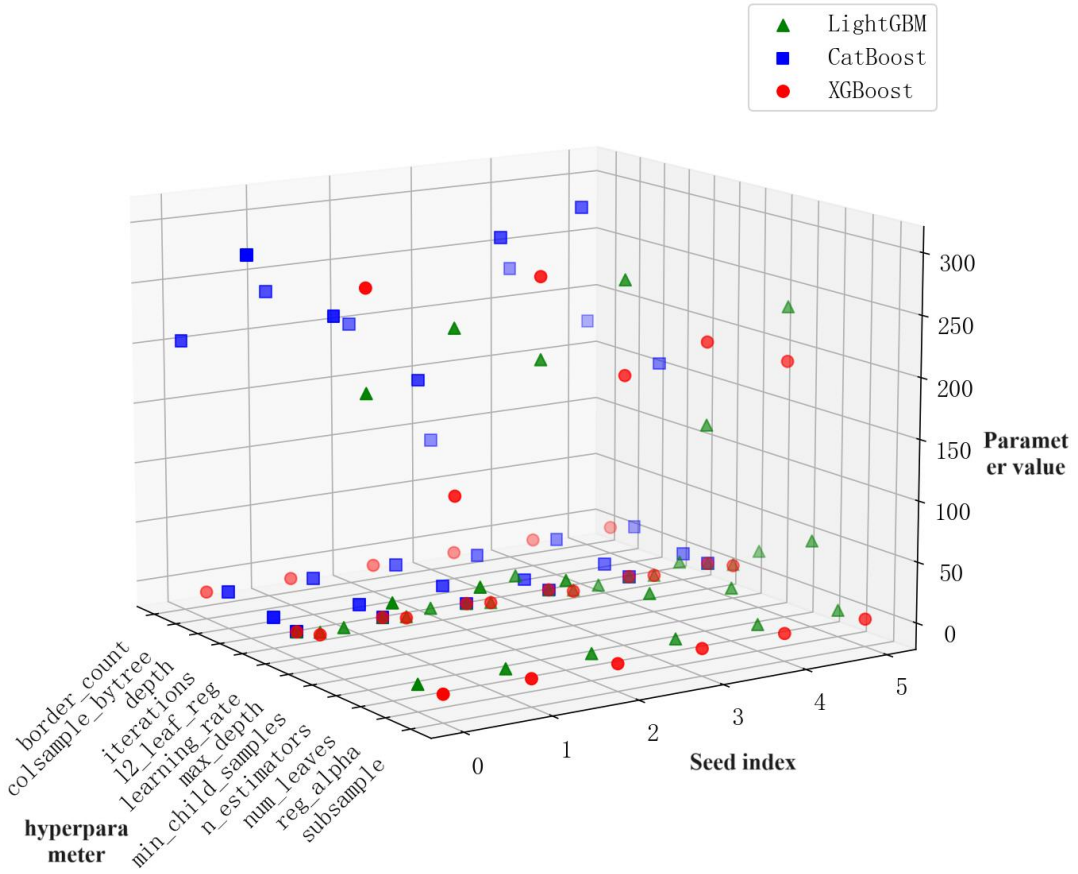


Figure 9. Optimal hyperparameters of gradient lifting algorithm

4.2 | Model training and parameter optimization

In this article, the performance evaluation phase focuses on using multiple metrics to comprehensively evaluate the model's performance, ensuring that the model's performance is understood from different perspectives.

First, the following four commonly used performance evaluation metrics are used:

Accuracy is the proportion of all correctly classified predictions (true cases and true negative cases) to the total sample. The formula is:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (1)$$

Accuracy is the proportion of observations correctly predicted to be positive to the total number of observations predicted to be positive. The formula is:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2)$$

The recall rate is the proportion of observations correctly predicted as positive classes to the total number of actual positive classes. The formula is:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

The F1 score is the conciliatory average of accuracy and recall and is a balanced representation of these two metrics. The formula is:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

The calculation results are shown in Table 1.

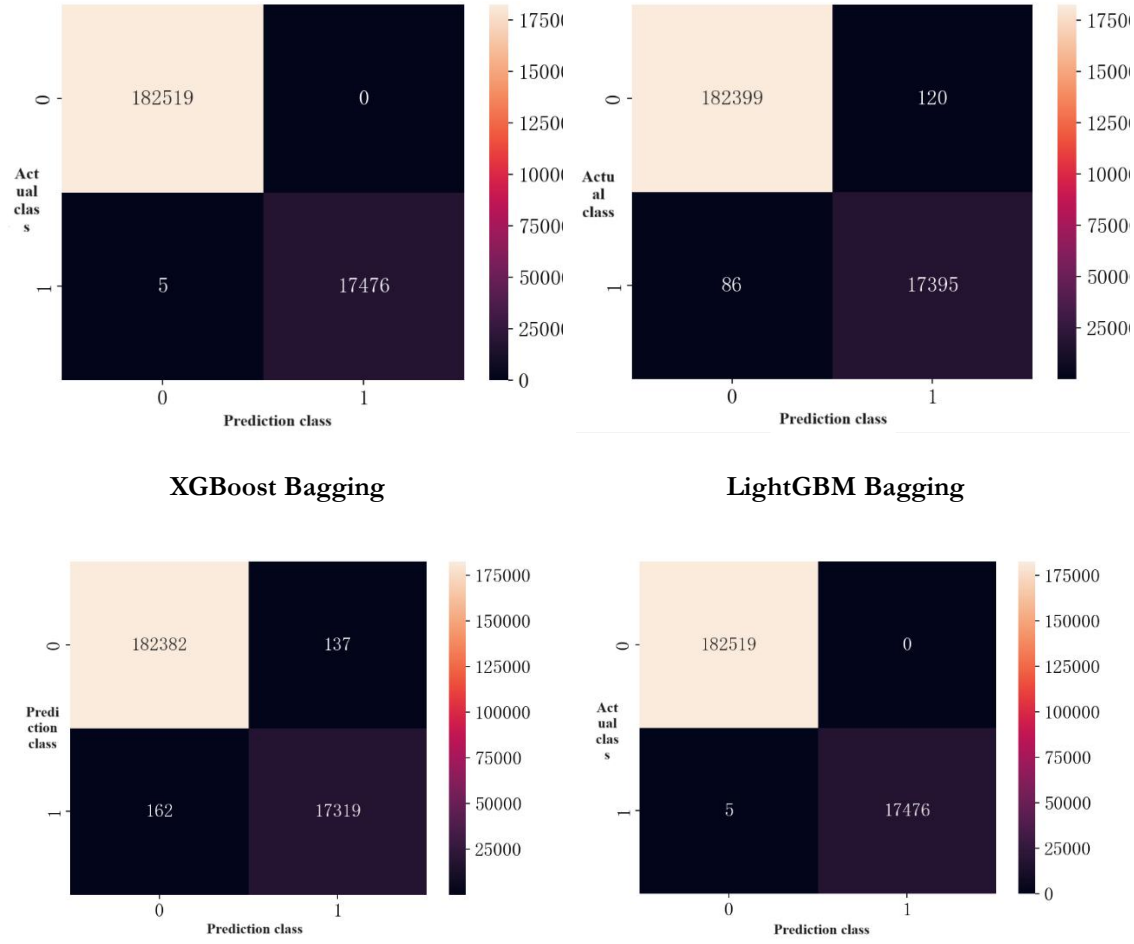
Table 1. Model comparison

	Accuracy	**Precision**	**Recall**	**F1 Score**
XGBoost Bagging	0.999975	0.999986	0.999857	0.999922
LightGBM Bagging	0.998970	0.996339	0.997211	0.996775
CatBoost Bagging	0.998505	0.995632	0.994991	0.995311
Stacking	0.999975	0.999986	0.999857	0.999922

It can be seen from Table 1 that XGBoost Bagging and Stacking models have the highest and equal precision, while LightGBM Bagging and CatBoost Bagging have lower precision. Considering that the data provided by the organizing committee is of high quality and the collected data sets in reality are often of poor quality. Although XGBoost Bagging combines the optimal models of different seeds, the algorithms of the models are too simple and not suitable for the real situation. In addition, different models can be Stacking. Make full use of the advantages of each model to predict the data set, more suitable for the real situation, so the combination model is not unnecessary, but in order to be better applicable to the reality of fraud prediction.

In order to visualize the performance of the model, ROC curves and confusion matrices were plotted for each model, and corresponding AUC values were calculated, which helped to evaluate

the classification ability of the model for different categories. By plotting the ROC curve and calculating the AUC separately for each class, you can gain a detailed understanding of how the model performs on each class, providing guidance for further optimization of the model. The confusion matrix and ROC curve of each model are shown in Figure 10.



XGBoost Bagging **LightGBM Bagging**

CatBoost Bagging **Stacking**

Figure 10. Confusion matrix and ROC curves of each model

5 | Conclusion

By analyzing the transaction data of one million telecom bank cards, this paper establishes a prediction model of telecom bank card fraud based on machine learning. By analyzing the importance of features, key indicators such as transaction amount ratio are identified, and the performance of the model is optimized by data preprocessing and feature engineering. The results show that the integrated learning algorithm model has excellent performance in predicting telecom bank card fraud with a prediction accuracy of 99.99%. Finally, the model can not only effectively predict fraud behavior, but also provide valuable preventive suggestions for relevant departments to improve property security.

References

[1] Wang Wei. A Credit Card Fraud Prediction Model Based on Improved Focal Loss Function XGBoost [J] Information Record Materials, 2022, 23 (12): 192-196.

[2] Yi Deyan Analysis and Research on Telecom Fraud Prevention Based on Support Vector Machine [D] University of International Business and Economics, 2024.

[3] Xiao Wenqin Research on Telecom Fraud Identification Based on BP Neural Network [D] Central China Normal University, 2023 .

[4] Sun Yujia Research on Fraud Phone Identification Based on User Communication Behavior Data [D] Capital University of Economics and Trade, 2023.

[5] Sun Yue, Ding Jianli A Stacking Integrated Prediction Model for Flight Delays in Adverse Weather Conditions [J/OL] Big data: 1-18 [2024-06-08].

[6] Chen Xiaoling, Zhang Cong, Huang Xiaoyu Research on Grain Yield Prediction Based on Bayesian LightGBM Model [J] China Journal of Agricultural Machinery Chemistry, 2024, 45 (06): 163-169.

[7] Pang Songling, Fan Kaidi, Chen Chao, etc A multi time scale prediction model for electric vehicle charging load based on LightGBM algorithm and travel chain theory [J/OL] Automotive Technology: 1-8 [2024-06-08].

[8] Jin Wanying Research on 5G Telecom User Prediction Based on Data Mining [D] Dalian University of Technology, 2022.

[9] Liu Bofei 5G potential user identification based on ensemble learning [D] Dalian University of Technology, 2022.

[10] Yu Jiang The research and application of data mining technology in the telecommunications field [D] Xiangtan University, 2022.