Hindawi
Complexity
Volume 2021, Article ID 6618841, 13 pages
https://doi.org/10.1155/2021/6618841

Research Article

Research on Credit Card Default Prediction Based on *k*-Means SMOTE and BP Neural Network

Ying Chen and Ruirui Zhang

School of Business, Sichuan Agricultural University, Chengdu 611830, China

Correspondence should be addressed to Ruirui Zhang; zhangruiruisw@163.com

Received 12 November 2020; Revised 28 January 2021; Accepted 17 February 2021; Published 13 March 2021

Academic Editor: Benjamin Miranda Tabak

Aiming at the problem that the credit card default data of a financial institution is unbalanced, which leads to unsatisfactory prediction results, this paper proposes a prediction model based on *k*-means SMOTE and BP neural network. In this model, *k*-means SMOTE algorithm is used to change the data distribution, and then the importance of data features is calculated by using random forest, and then it is substituted into the initial weights of BP neural network for prediction. The model effectively solves the problem of sample data imbalance. At the same time, this paper constructs five common machine learning models, KNN, logistics, SVM, random forest, and tree, and compares the classification performance of these six prediction models. The experimental results show that the proposed algorithm can greatly improve the prediction performance of the model, making its AUC value from 0.765 to 0.929. Moreover, when the importance of features is taken as the initial weight of BP neural network, the accuracy of model prediction is also slightly improved. In addition, compared with the other five prediction models, the comprehensive prediction effect of BP neural network is better.

1. Introduction

Recently, the state vigorously promotes the economic construction of large- and medium-sized cities, which not only improves people's living standards but also changes people's consumption concept and consumption mode. People are more and more inclined to spend ahead of time and mortgage their "credit" to the bank to enjoy certain things in advance. However, when consuming, people often lack rational thinking and overestimate their ability to repay loans to banks in time. On the one hand, it increases the loan risk of banks; on the other hand, it increases the credit crisis of consumers themselves [1]. With a large number of banks selling credit cards, the phenomenon of credit card default emerges one after another. It is very important for banks to effectively identify high-risk credit card default users. Generally speaking, compared with the credit card customers who have not paid their loans overdue, there are fewer overdue repayments [2, 3]. This variable feature of overdue and overdue loan repayment is called "two classifications" in machine learning prediction. In the prediction of "two classifications," a few categories are called positive examples (default), and most categories are called counterexamples (nondefault). However, most of the credit card loan data are unbalanced. In view of this situation, domestic and overseas scholars have taken up on a large scale a lot of researches. Khoshgoftaar et al. [4] proposed an evolutionary sampling method for unbalanced data, which uses genetic algorithms to selectively delete most types of samples and retain samples with a lot of feature information. Compared with other existing data sampling technologies, evolutionary sampling technology has better performance and is more conducive to empirical replication. The FN undersampling method used by Zhao et al. [5] regarded the minority class as a cluster, which was divided into multiple regions. And they calculated the distance from the negative class samples to the sample mean point in each region, reserving only one sample point in each region. Finally, the remaining negative class samples were used as new negative class samples and the original positive class samples for training and analysis. Zan et al. [6] used the generative countermeasure network (GAN) to synthesize a few samples to balance the data, then used AdaBoost to change the weight of the input samples, and established a prediction model based on the decision tree classifier. To a certain extent, the recognition rate of unbalanced data was improved. Hu et al. [7] used an improved version of oversampling and undersampling techniques to solve the problem of data imbalance and synthesized the new samples by assigning higher weights to adjacent minority samples through a weight vector. Based on the Euclidean distance standard undersampling most types of samples and keeping the number constant during the resampling process, they found that this method was superior to using a single data sampling technique. Han et al. [8] used an improved version of the smooth algorithm: borderline-smote, which essentially synthesizes new samples from minority samples. However, the original smooth algorithm selects a small number of samples around k nearest neighbors, while scholars use an improved version of the algorithm to find the minority class at the boundary line and use this method to synthesize new samples. Wang et al. [9] constructed a deep learning prediction model for imbalanced data. The model proposed a new loss function on the basis of the original neural network. This method does not need to balance the data in advance. Predictive analysis can be performed directly, and it can effectively reduce the classification error of positive and negative examples. Jiao et al. [10] proposed a reinforcement learning cumulative reward mechanism to improve the attribute selection of the classification regression tree, so as to improve the model's prediction probability for a small number of samples.

We can see that the problem of category imbalance is mainly solved from the following two perspectives: the first perspective is to balance the data by changing the number of samples. This method can also be divided into three aspects. On the one hand, it is to improve the oversampling method. On the other hand, it is based on the principle of undersampling to change the data distribution. On the third hand, it is the method of combining oversampling and undersampling. The second perspective is to improve the classifier algorithm to improve the prediction performance of the model and at the same time use relevant evaluation indicators to evaluate the prediction results. Under normal circumstances, since undersampling will lose information, oversampling is the most widely used technique, and smote is the more common method. However, we have found that most scholars cannot reduce the imbalance between and within the sample categories at the same time when using the improved version of the smooth method, and the applicability of the improved version of the classifier is also limited. Therefore, this paper proposes an improved version of the smooth algorithm with better applicability, which combines the k-means algorithm. This method clusters all samples using the k-means unsupervised learning algorithm, finds clusters with more samples in the minority class, and then uses the smote method that synthesizes new samples in the cluster to change the data distribution. It can not only reduce the imbalance between the categories but also reduce the imbalance within the categories. At the same time, it combines the BP neural network method to predict the credit card default situation to help the bank to identify credit card risks effectively.

2. Basic Theory

2.1. PCA. The main idea of the principal component analysis (PCA) method is to transform the n-dimensional feature variable through the coordinate axis and the origin to form a new m-dimensional feature (usually, m is less than n) [11]. This m-dimensional feature is also called principal component. Its essence is to replace a series of related sample features with newly generated comprehensive features that are irrelevant to each other. When analyzing the data, you can set the cumulative variance ratio determination factor in advance. The working steps of PCA are as follows:

> The first step is to standardize the original sample. This step is automatically executed by the software that analyzes the data.
>
> The second step is to determine the correlation between the sample features and calculate the correlation coefficient matrix.
>
> The third step is to determine the number m of principal components after dimensionality reduction, calculate the eigenvalues and their corresponding eigenvectors, and then synthesize these eigenvectors to obtain each principal component.
>
> The fourth step is to determine the comprehensive evaluation index, calculate the information contribution rate of each feature value and principal component, and then weight these values to obtain the final evaluation value.

2.2. Feature Importance Calculation of Random Forest. Random forest is a relatively basic machine learning algorithm, which is widely used in predictive analysis [12], data labeling [13], tag ranking [14], feature importance calculation [15], and other fields. The principle of the algorithm is as follows: using bootstrap method to randomly construct n decision trees, each decision tree is split and pruned and finally combined to form a random forest. In this paper, random forest is used to calculate feature importance, which is used as the initial weight of BP neural network. The basic algorithm steps are as follows:

> The first step is to calculate the out-of-bag data error (error1) by using the sample data that has not been selected (out-of-bag data) when drawing samples to construct a decision tree.
>
> The second step is to randomly add noise interference to all the sample features of the data outside the bag and then calculate the error again and record it as error2.
>
> The third step is to calculate the importance of a feature = $\sum_i^n (\text{error2} - \text{error1})/n$ (n is the number of decision trees constructed).

2.3. BP Neural Network. The prediction model used in this paper is the BP neural network algorithm, which is a feedforward neural network for error backward update. It is often used for bank risk analysis [16], geological disaster monitoring [17], image and handwritten digit recognition [18, 19], and other fields. BP neural network consists of three parts: input layer, middle layer, and output layer. In the model, data samples enter the input layer through a weighted combination of different weights, then pass through the middle layer, and finally get the result from the output layer. Different weights and activation functions make the output of the model very different. In this experiment, the following steps were taken:

The first step is to assign some parameters and initialize some parameters. In the experiment, this paper takes the feature importance calculated by the random forest as the weight of the input layer X_i and sets the same value for the weight of one input variable corresponding to multiple hidden layers. In addition, the number of nodes in the input layer, hidden layer, and output layer is determined.

The second step is to calculate the output of the hidden layer Z_i:

$$Z_i = f\left(\sum_{i=1}^{l} W_{ij}X_i + a_j\right). \tag{1}$$

The third step is to calculate the output layer Y_i:

$$Y_k = \sum_{k}^{n} f_j W_{jk} + b_k. \tag{2}$$

Among them, both aj and bk in the second and third steps are offset.

The fourth step is to calculate the error E:

$$E = \frac{1}{2}\sum_{k=1}^{s}\left(y_k - Y_k\right)^2. \tag{3}$$

Among them, yk is the expected output value, and Yk is the actual output value.

The fifth step is to update the weights and biases in reverse.

3. *k*-Means SMOTE Algorithm

We know that smote is a method for synthesizing new samples and solving data imbalance proposed by Chawla et al. [20] and is widely used in various fields. Smote is an improved method of random oversampling technology. It is not a simple random sampling, repeating the original sample, but a new artificial sample generated by a formula. But the smote algorithm will also increase the imbalance between the positive and negative classes of the sample to a certain extent. Therefore, according to the problem of imbalance of credit card sample categories, this paper uses an improved smote algorithm called *k*-means SMOTE algorithm. This algorithm can reduce the imbalance between categories on the one hand and reduce the imbalance within categories on the other hand. In this experiment, we first cluster all samples (30,000), then use *k*-means method to filter clusters with more minority categories, select clusters with more minority categories after filtering, and finally perform smote oversampling in the filtered clusters. The detailed steps of the *k*-means SMOTE algorithm are as follows:

The first step is to randomly select k points among all samples $D = x_1, x_2, x_3, \ldots, x_{30000}$ and use them as the sample cluster centers $C_1, C_2, C_3, \ldots, C_k$.

The second step is to calculate the distance from each sample to the cluster center:

$$d = \sqrt{\sum \left(x_i - C_k\right)^2}. \tag{4}$$

Among them, $x_1, x_2, x_3, \ldots x_i \in D$; $C_1, C_2, C_3, \ldots, C_K \in C$.

The third step is to allocate the sample into the closest clusters:

$$x^i \in C_{\text{nearest}}. \tag{5}$$

The fourth step is to recalculate the cluster center:

$$\mu_i = \frac{1}{|C_i|}\sum_{x \epsilon C_i} x. \tag{6}$$

The fifth step is to repeat the above second, third, and fourth steps until the cluster center no longer changes.

The sixth step is to filter clusters with fewer minority classes and select clusters with more minority classes to synthesize new minority samples.

The seventh step is to perform smote oversampling of CK in each filtered cluster:

$$X_{\text{new}} = x_c + \text{rand}(0, 1) \times \left(\tilde{x} - x_c\right). \tag{7}$$

Among them, rand(0, 1) represents a random number between 0 and 1, X_{new} represents a new synthesized negative class sample, and xc represents a negative class randomly selected from m nearest neighbors in the filtered clusters. $\tilde{x}$ represents the negative samples in the filtered clusters except m neighbors. The *k*-means SMOTE algorithm flow is shown in Figure 1.

4. Experimental Data and Preliminary Analysis

4.1. Preliminary Analysis of Data. This paper uses data on credit card usage, which comes from the kaggle website (https://www.kaggle.com/uciml/default-of-credit-card-clients-dataset). The sample size of this data is 30,000, of which 6,636 are in the positive category (default) and 23,364 in the negative category (no default). The sample has a total of 25 variables. In this experiment, considering that the variable ID has no relationship with the target variable, the deletion process was

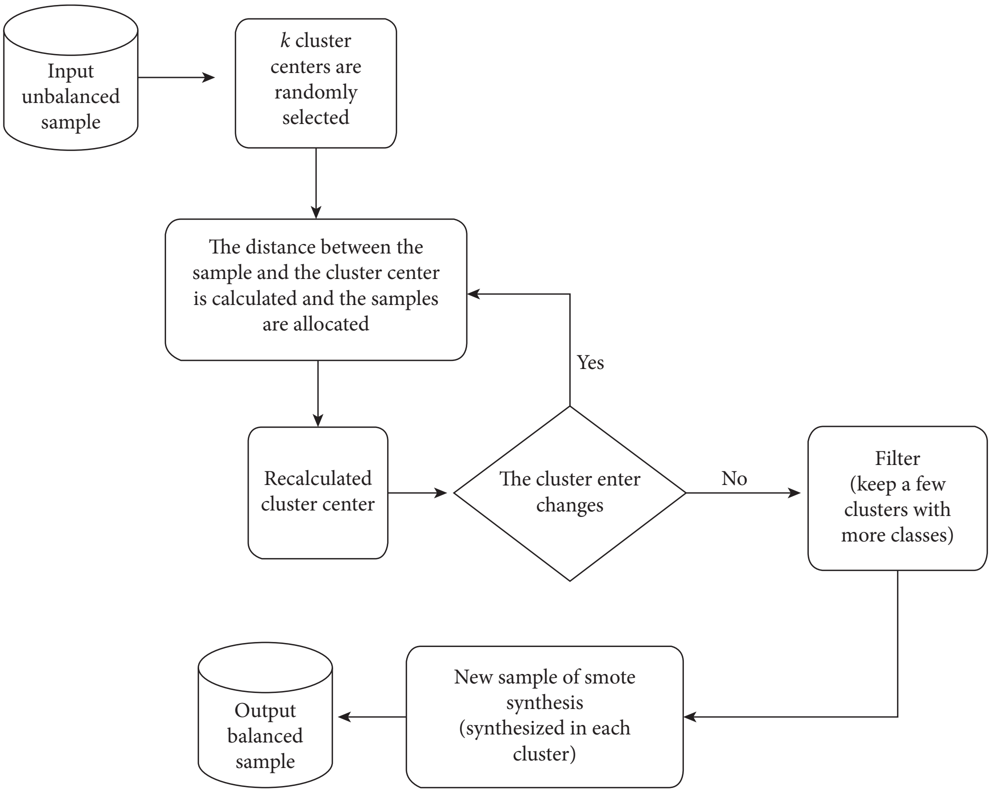

Input unbalanced sample
k cluster centers are randomly selected
The distance between the sample and the cluster center is calculated and the samples are allocated
Yes
Recalculated cluster center
The cluster enter changes
No
Filter (keep a few clusters with more classes)
New sample of smote synthesis (synthesized in each cluster)
Output balanced sample

Figure 1: *k*-means SMOTE algorithm flowchart.

performed. 23 characteristic variables and 1 target variable were selected. The variables are shown in Table 1:

Among these 23 features, each feature has been processed accordingly. For the feature limit_bal, we draw a density map according to the default type, and the result is shown in Figure 2.

It can be found from Figure 2 that when the given credit amount is approximately below 150,000, the probability of default is greater than that of nondefault. This shows that when the credit amount is low, there may be more defaulters. For the feature age, we also performed a visual analysis, as shown in Figure 3.

Figure 3 shows that the probability of nondefault of age between approximately 25 and 40 is higher, which indicates that consumers in this age group are more capable of repaying credit card loans. This may be because their work and family tend to be stable without too much pressure. For the feature sex, we draw a stacked histogram according to the target variable, as shown in Figure 4.

As shown in Figure 4, whether it is male or female, the proportion of default consumers is still relatively low, which is in line with the general situation. Conventionally, most of the default data such as credit card fraud are uneven, and we need to make some adjustments to the model based on the actual situation. For the feature education, we find that the feature has six attribute values, and the meanings of the numbers 5 and 6 are unknown, in order to avoid causing a "dimensional disaster" when processing data. We merge them into one meaning (unknown) and draw a stacked histogram to visualize this feature, as shown in Figure 5.

Table 1: Variable attributes.

Number	Variable	Type
1	limit_bal	Continuous
2	Sex	Category
3	Age	Continuous
4	Education	Category
5	Marriage	Category
6	pay_0	Category
7	pay_2	Category
8	pay_3	Category
9	pay_4	Category
10	pay_5	Category
11	pay_6	Category
12	bill_amt1	Continuous
13	bill_amt2	Continuous
14	bill_amt3	Continuous
15	bill_amt4	Continuous
16	bill_amt5	Continuous
17	bill_amt6	Continuous
18	pay_amt1	Continuous
19	pay_amt2	Continuous
20	pay_amt3	Continuous
21	pay_amt4	Continuous
22	pay_amt5	Continuous
23	pay_amt6	Continuous
24	Default.payment.next.month	Category

For the feature marriage, we draw the same graph as the feature sex and education. The default and nondefault conditions of this feature are shown in Figure 6.

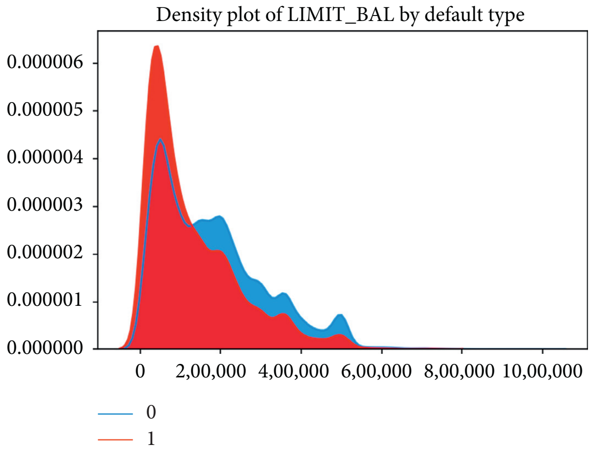


Figure 2: Density diagram of limit_bal.

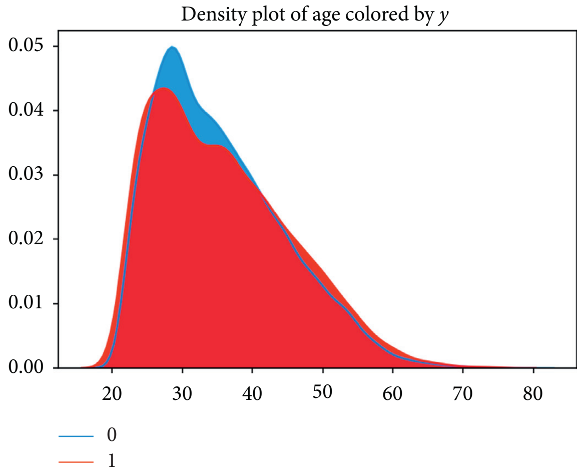


Figure 3: Density diagram of age.

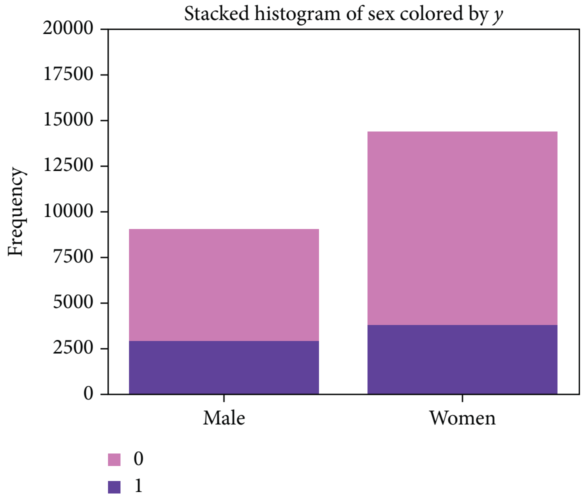


Figure 4: Stacked histogram of gender.

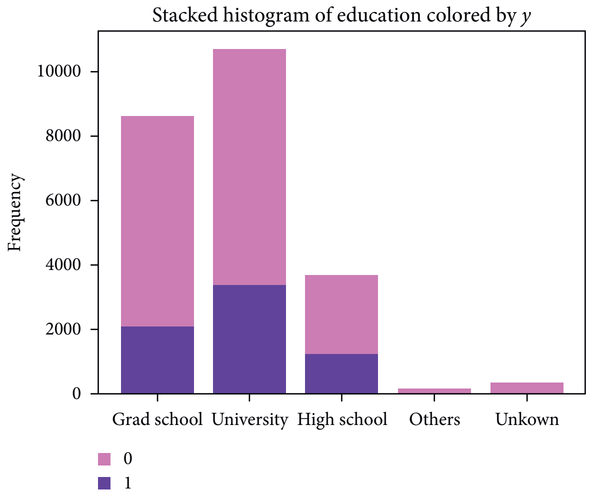


Figure 5: Stacked histogram of education.

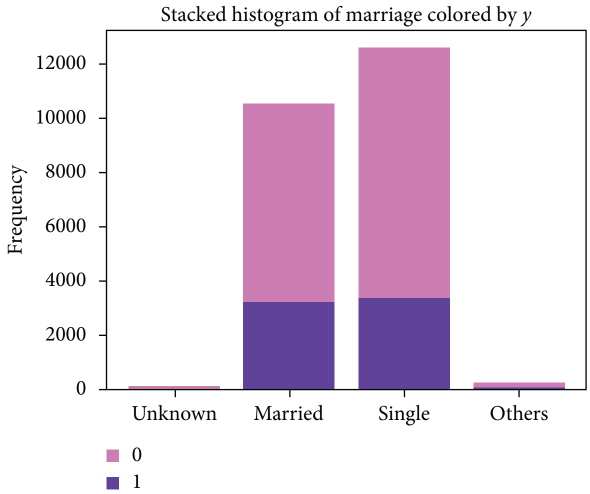
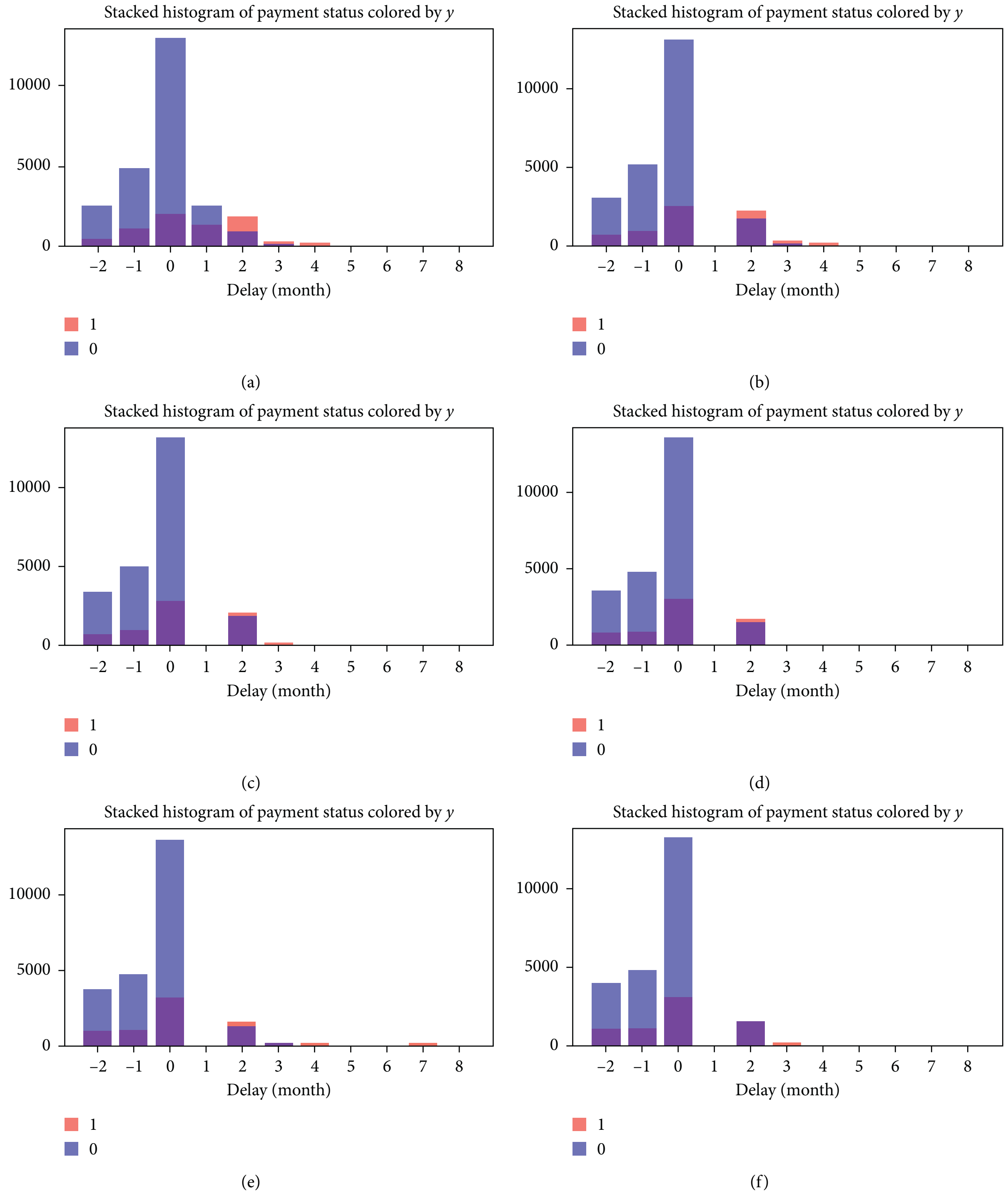


Figure 6: Stacked histogram of marriage.

It can be seen from the above three figures that the sample set is unbalanced in the corresponding attribute values of the three characteristics of gender, education, and marriage. For the feature series payment status, we draw different stacked histograms according to different months, and the results are shown in Figure 7.

It can be seen from Figure 7 that consumers who delay payment by one month or less have fewer credit card defaults and almost never happen. In the three months of May, August, and September, for consumers who delayed payment for more than 2 months, the greater the probability of their credit card default is, the more likely it is to increase the loan risk of financial institutions. For the feature series BillAMT and PayAMT, we also perform the corresponding analysis and draw a line graph to visualize the two features, as shown in Figures 8 and 9.

As shown in Figures 8 and 9, due to the imbalance of the data, the line of default only occupies the front part of the figure. Figure 8 shows the amount of the bill, and Figure 9 shows the amount previously paid. Comparing these two images, we find that the six subimages in Figure 9 have greater fluctuations and greater range than the six subimages in Figure 8. Moreover, the uncertainty of the previous



(a)

(b)

(c)

(d)

(e)

(f)

Figure 7: Stacked histogram of payment status. (a) Payment status in September. (b) Payment status in August. (c) Payment status in July. (d) Payment status in June. (e) Payment status in May. (f) Payment status in April.

payment amount has also increased the difficulty for banks to adjust the credit card loan limit.

4.2. Data Processing and Feature Importance. In this experiment, there are a total of 23 features and 1 target variable. After coding and data cleaning, 23 features become 89 input variables. This is a heavy load for model operation and is not conducive to the prediction results of this paper. For comparative analysis with other models, this paper uses PCA for dimensionality reduction, finally obtains 27 input variables, then uses random forest to calculate the importance of these 27 variables, and uses them as the initial weight of the BP neural network. The calculation results of the feature importance are shown in Table 2.

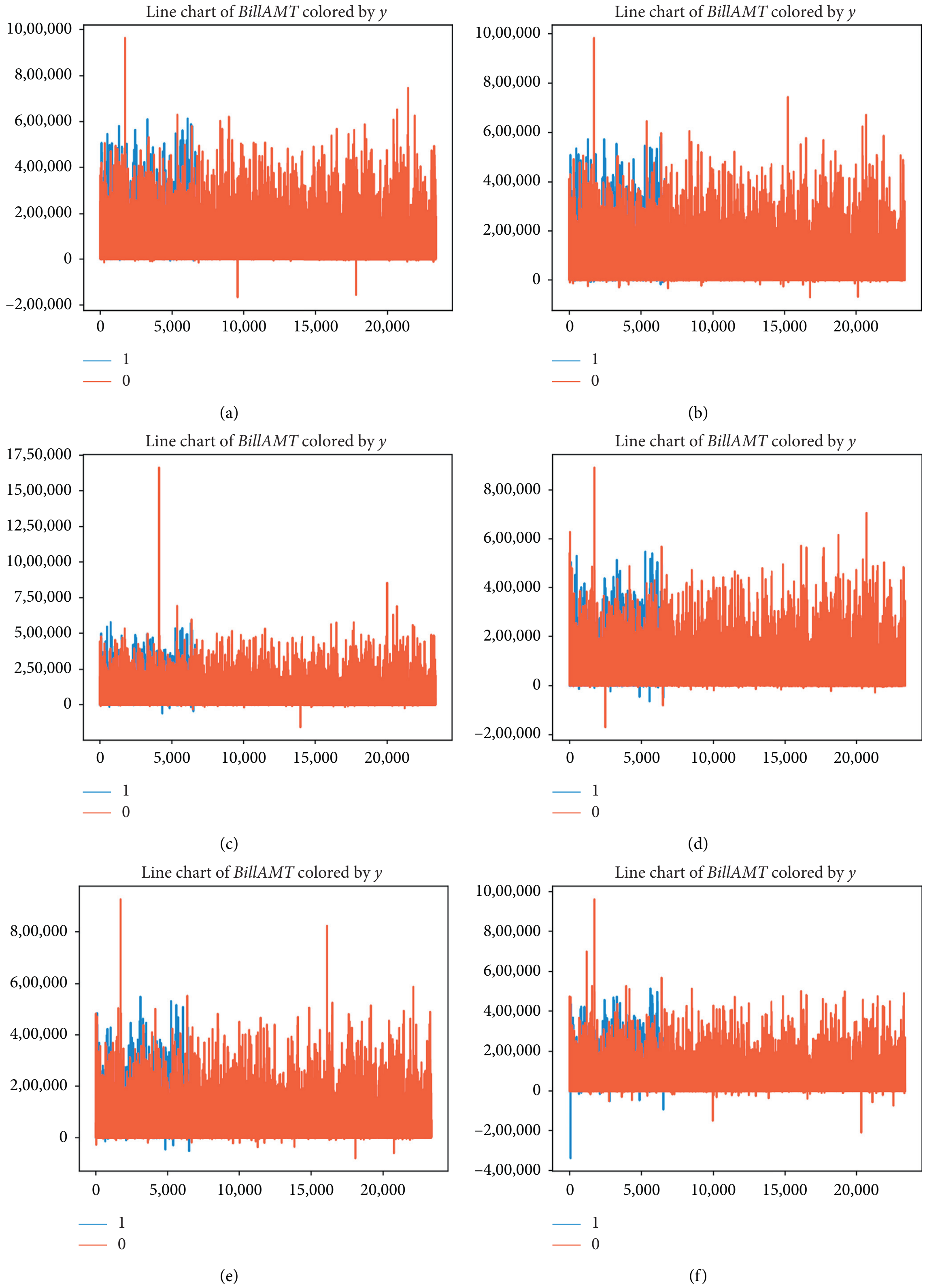
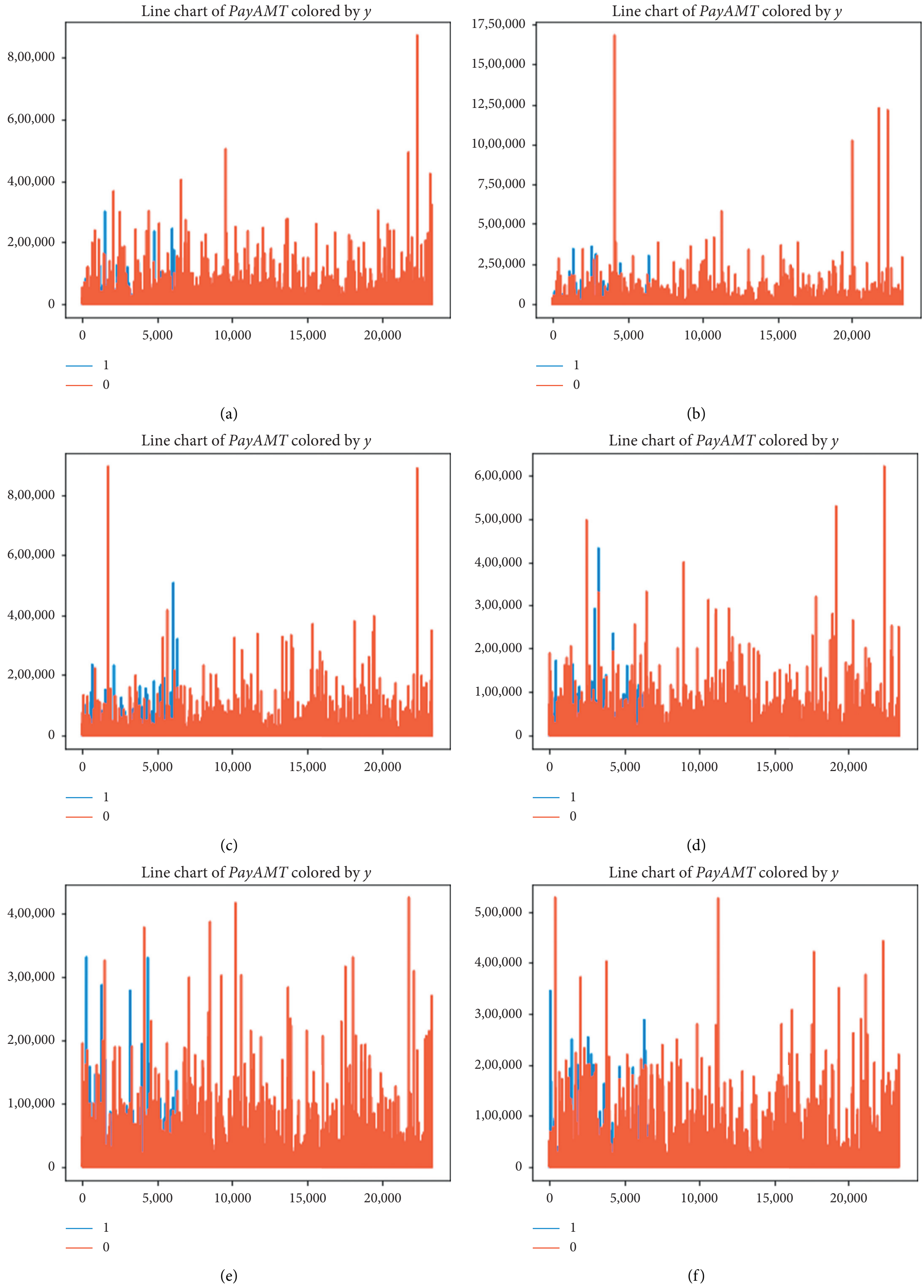


Figure 8: Line chart of billamt. (a) Amount of bill statement in September. (b) Amount of bill statement in August. (c) Amount of bill statement in July. (d) Amountof bill statement in June. (e) Amount of bill statement in May. (f) Amount of bill statement in April.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 9: Line chart of payment. (a) Amount of previous payment in September. (b) Amount of previous payment in August. (c) Amount of previous payment in July. (d) Amount of previous payment in June. (e) Amount of previous payment in May. (f) Amount of previous payment in April.

Table 2: Feature importance.

Number	Importance
1	0.06085758
2	0.06335107
3	0.0198413
4	0.13245482
5	0.10467543
6	0.07704057
7	0.02595485
8	0.0214903
9	0.05299085
10	0.02292236
11	0.0207296
12	0.02803694
13	0.02700749
14	0.02271594
15	0.03059199
16	0.02795678
17	0.02408269
18	0.01549738
19	0.01641339
20	0.04804207
21	0.01622095
22	0.0165196
23	0.052694
24	0.02696936
25	0.01606919
26	0.01431037
27	0.01456314

5. Model Prediction and Comparative Analysis

5.1. Model Evaluation Method. According to the actual situation, for unbalanced data, we should use the evaluation index of unbalanced data [21], but because at the beginning of the experiment, we have balanced the number of positive and negative classes in the sample. And we are still using the two-class evaluation indicators commonly used in the past: hybrid matrix, recall, precision, f1-score, AUC value, and so on.

5.2. BP Neural Network Prediction Model. This paper constructs a BP neural network prediction model based on credit card default data. Since this paper has 27 input variables, 55 neurons in the hidden layer, and 2 output layers, the BP neural network model used is shown in Figure 10.

Then, we use the 27 features after principal component dimensionality reduction as input variables $X_1, X_2, \ldots, X_{27}$ and use the feature importance calculated by the random forest as the initial weight of BP neural network. For example, the calculation formula for the weight W of the hidden layer is as follows:

$$\begin{gathered}
\begin{bmatrix} W_{11} & W_{12} & W_{13} & W_{14}\cdots & W_{155} \\ W_{21} & W_{22} & W_{23} & W_{24}\cdots & W_{255} \\ W_{31} & W_{32} & W_{33} & W_{34}\cdots & W_{355} \\ & & \cdots & & \\ W_{271} & W_{272} & W_{273} & W_{274}\cdots & W_{2755} \end{bmatrix} \\
\Downarrow \\
\begin{bmatrix} 0.06085758 & 0.06085758 & 0.06085758 & 0.06085758\ldots & 0.06085758 \\ 0.06335107 & 0.06335107 & 0.06335107 & 0.06335107\ldots & 0.06335107 \\ 0.0198413 & 0.0198413 & 0.0198413 & 0.0198413\ldots & 0.0198413 \\ & & \cdots & & \\ 0.01456314 & 0.01456314 & 0.01456314 & 0.01456314\ldots & 0.01456314 \end{bmatrix}.
\end{gathered} \tag{8}$$

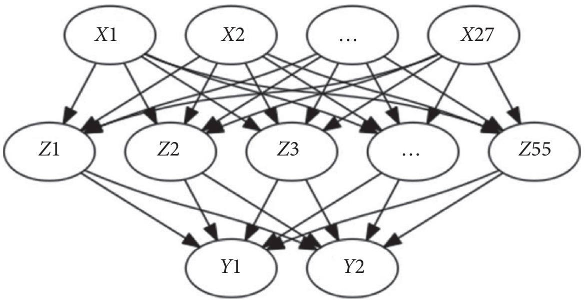


Figure 10: BP neural network model.

In formula (8), there are 27 rows and 55 columns. 27 rows are the number of input variables, and 55 columns are the number of hidden layer neurons. In this experiment, we set each row in the matrix to be the corresponding feature importance (as in the above formula matrix 2) and substitute the result into the model for prediction. We find that when the weights are initialized, the accuracy of the model prediction is 0.8796, and when the feature importance is assigned to the weights, the accuracy of the model prediction is 0.8811. In terms of amount, the accuracy of the second case is slightly higher.

When building the model, we used a three-layer BP neural network to build a credit card default prediction model. The input layer has 27 neurons, the hidden layer has 55 neurons, and the output layer has 2 neurons. The hidden layer is calculated using the following empirical formula:

$$n = 2 \times n1 + 2, \text{ ($n1$ is the number of input layers).} \tag{9}$$

In addition to the initial weight of the hidden layer and the number of neurons in the hidden layer, we have performed a simple process, and the other parameters are default values.

Due to the uneven distribution of the experimental data, we use the k-means SMOTE algorithm to solve this problem. For the parameter k in the k-means SMOTE algorithm, we use the following empirical formula to calculate:

$$k = \sqrt[2]{\frac{N}{2}}, \text{ (N is the total number of samples).} \tag{10}$$

Then we substitute the sample size of 30000 (N) into the above formula, can calculate the value of k to be about 122, substitute it into the k-means SMOTE algorithm, and draw the ROC curve graph to intuitively compare the prediction performance of the model before and after k-means SMOTE. And we find that k-means SMOTE greatly improves the prediction performance of the model. The result is shown in Figure 11.

In Figure 11, we find that after the sample is processed by the k-means SMOTE algorithm, the prediction of the model has been greatly improved. The AUC value has been increased from 0.765 to 0.930, the ROC curve of the model is closer to the straight line 1 above the coordinate axis, and the accuracy rate has changed from 0.8252 to 0.8796.

Normally, the BP neural network model with more parameters is prone to overfitting. Because of the high fitting degree of the model, it is possible to learn the noise. We compare the performance of the prediction model in the training set and the testing set, and the results are as follows.

It can be seen from the above table that the values of performance indexes of the prediction model in these two groups of data set have little difference, so we judge that the possibility of overfitting the model in this experiment is relatively low. And the performance of the model can achieve the desired results.

5.3. Comparative Analysis with Other Models. In order to verify the effectiveness of the method used in this experiment, we also establish five other common machine learning models for predictive analysis under the same conditions. We have compared and analyzed the prediction results of these five models in the same situation and used several common performance indicators to evaluate the model. Since the confusion matrix is used to show the prediction results according to different situations, it is not easy to compare the performance of these five models. We adjust it slightly (e.g., the accuracy rate is approximately equal to the average of the accuracy of model positive and negative examples) as shown in Table 3.

It can be seen from Table 4 that the F1 values of these six models have reached above 0.8, indicating that these six models can effectively predict the credit imbalance data in this paper, but the comprehensive prediction performance of the BP neural network is slightly better. The AUC value is the highest among the six models, and the accuracy rate is higher for SVM. But the running time of the SVM model is too long, close to 6 minutes; compared to other models, the running efficiency of SVM is very low. If the amount of data is very large, it is not a wise choice for us to use SVM for prediction. In addition, we can find that except the lower AUC value of the decision tree, the difference in the AUC value of other models is not particularly large. This situation can also be intuitively seen through the ROC curve. The result is shown in Figure 12.

In Figure 12, we can find that if we do not look at the numbers in Table 3, we cannot see the obvious difference in the ROC curves of the first five models from Figure 12. In the above figure, the sixth image is the ROC curve of the decision tree, which is obviously different from the previous five images. This also shows that the tree has the worst performance among the six prediction models.

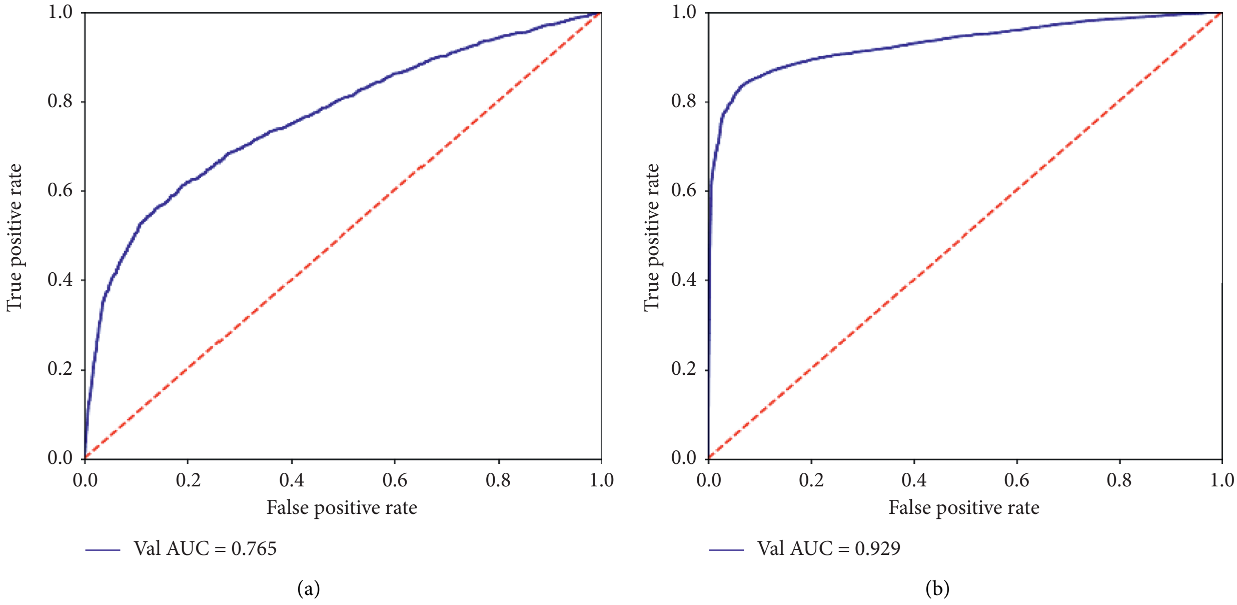


FIGURE 11: Comparison of ROC curves (a) before and (b) after *k*-means SMOTE.

TABLE 3: Comparison results of different data sets.

Data set	Accuracy	Precision	Recall	F1_score
Training set	0.881	0.923	0.840	0.880
Testing set	0.884	0.924	0.837	0.874

TABLE 4: Comparison results of six models.

Model	AUC	Accuracy	Precision	Recall	F1	Time (s)
BPnn	0.929	0.881	0.923	0.84	0.880	23.89
SVM	0.92	0.884	0.885	0.885	0.885	5 m 55.17
Logistic	0.92	0.876	0.875	0.88	0.877	52
RandomForest	0.92	0.817	0.875	0.875	0.875	35.62
KNN	0.91	0.869	0.87	0.87	0.87	4.75
Tree	0.82	0.871	0.82	0.815	0.817	3.62

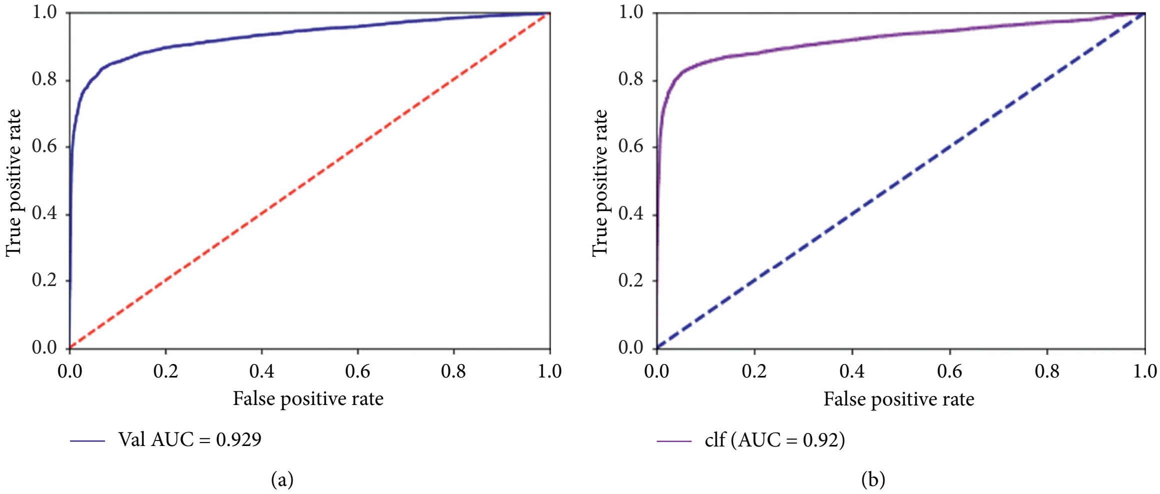
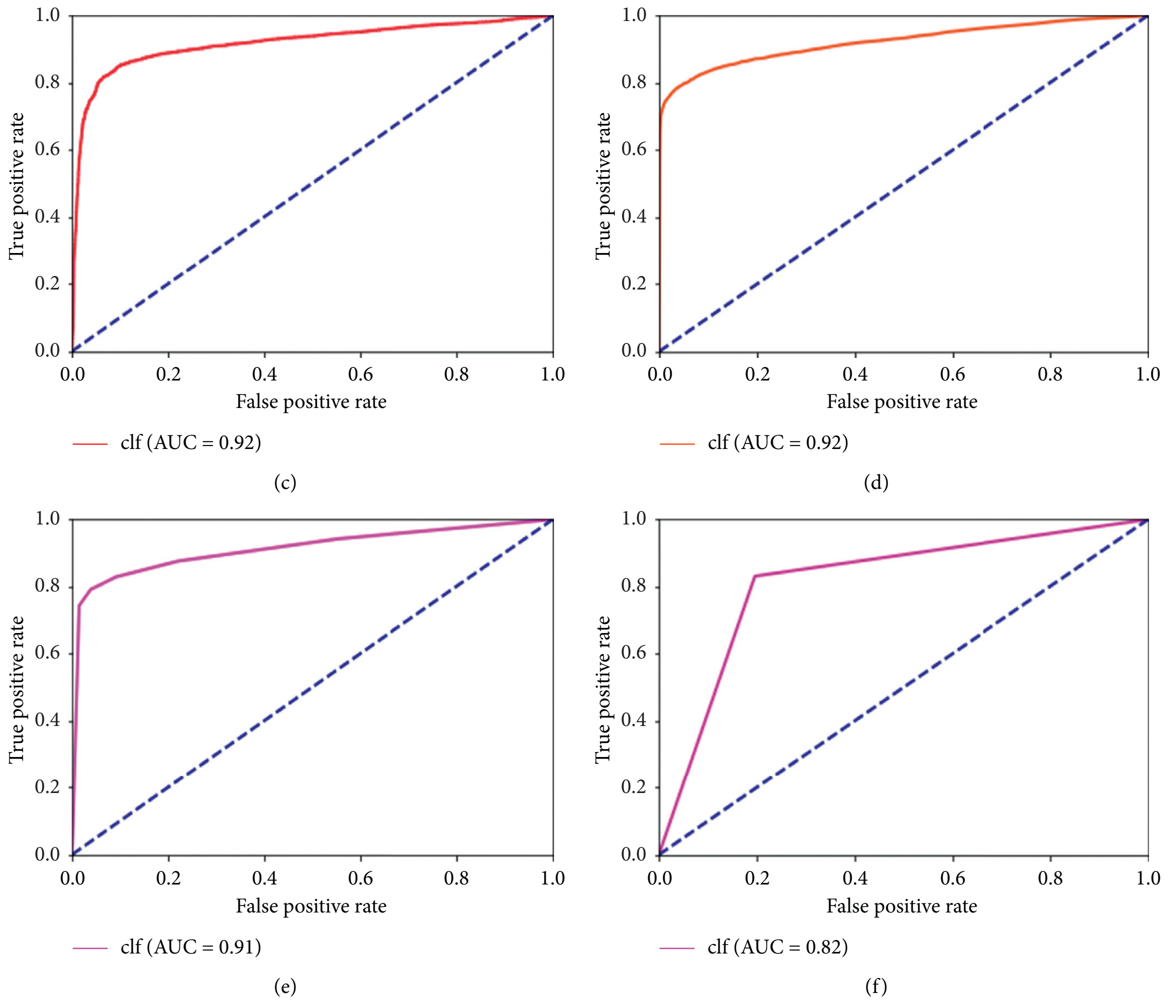


FIGURE 12: Continued.



(c)

(d)

(e)

(f)

Figure 12: Comparison of six models' ROC curves. (a) BPnn ROC. (b) SVM ROC. (c) Logistic ROC. (d) RandomForest ROC. (e) KNN ROC. (f) Tree ROC.

6. Summary

This paper proposes a comprehensive way by using k-means SMOTE and BP neural network algorithms for data imbalance. We find that the improved version of the smote algorithm (k-means SMOTE) not only effectively solves the problem of data imbalance but also improves the prediction performance of the model. In addition, we also find that using the feature importance calculated by the random forest as the initial weight of the hidden layer of the BP neural network can slightly improve the prediction performance of the model to a certain extent. However, this change is not obvious. On the one hand, it may be because the credit card default data has many influencing factors and is more complicated. We cannot take all such influencing factors into account, which may indirectly affect the calculation results of feature importance. On the other hand, we think that the amount of sample data may not be enough, the model of BP neural network is relatively simple, and there is no better interpretation of these data for predictive analysis.

In addition, with the gradual increase in the penetration rate of credit cards in our country, the research on its default risk has the following suggestions. On the one hand, we should further improve the construction of the credit indicator system. A good credit index system is conducive to better assessment of personal credit, and a risk prediction model with better classification performance can be established. Specifically, methods such as Delphi expert method, analytic hierarchy process, and regression analysis can be used to find the most representative individual credit indicators, then determine the weight of each indicator, and finally dynamically manage the evaluation system. On the other hand, we should strengthen risk management and control. Since credit card loan default involves personal moral issues, it is highly subjective and uncontrollable. Although major financial institutions are committed to developing the best methods for credit card loan risk avoidance, they have not been able to completely resolve the problem of credit defaults. Therefore, financial institutions should focus on controlling and avoiding risks and try their best to reduce risk losses. Based on the idea of machine learning integration methods, they can comprehensively use each superior classifier to develop a more versatile risk control model.

Data Availability

This paper uses data on credit card usage, which comes from the Kaggle website (https://www.kaggle.com/uciml/default-of-credit-card-clients-dataset).

Conflicts of Interest

The authors declare that they have no conflict of interest.

References

[1] Z. Feng and M. Feng, “Research on credit card scoring model based on AHP,” *Finance Theory and Practice*, vol. 1, pp. 74–77, 2016.

[2] R. Mei, Y. Xu, and G. Wang, “Study on analysis and influence factors of credit card default prediction model,” *Statistics and Applications*, vol. 5, no. 3, pp. 263–275, 2016.

[3] F. Thabtah, S. Hammoud, F. Kamalov, and A. Gonsalves, “Data imbalance in classification: experimental evaluation,” *Information Sciences*, vol. 513, pp. 429–441, 2020.

[4] T. Khoshgoftaar, N. Seliya, and D. Drown, “Evolutionary data analysis for the class imbalance problem,” *Intelligent Data Analysis*, vol. 14, no. 1, pp. 69–88, 2010.

[5] Z. Zhao, G. Wang, and X. Li, “Improved undersampling method for imbalanced data classification based on support vector machine,” *Journal of Sun Yat-Sen University (Natural Science Edition)*, vol. 6, pp. 10–16, 2012.

[6] M. Zan, G. Yanrong, and F. Guanlong, “Credit card fraud classification based on GAN-AdaBoost-DT imbalance classification algorithm,” *Journal of Computer Applications*, vol. 39, no. 2, pp. 314–318, 2019.

[7] L. Hu, Z. Peng, W. Xiang, and X. Rongze, “A new combination sampling method for imbalanced data,” in *Proceedings of the 2013 Chinese Intelligent Automation Conference: Intelligent Information Processing*, vol. 256, pp. 547–554, Yangzhou, China, 2013.

[8] H. Han, W.-Y. Wang, and B.-H. Mao, “Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning,” *Advances in Intelligent Computing*, vol. 3644, no. 1, pp. 878–887, 2005.

[9] S. Wang, W. Liu, and J. Wu, “Training deep neural networks on imbalanced data sets,” in *Proceedings of the International Joint Conference on Neural Networks*, pp. 4368–4374, IEEE, Vancouver, Canada, July 2016.

[10] J. Jiao, X. Zhang, F. Li, and Z. Niu, “Identically distributed multi-decision tree based on reinforcement learning and its application in imbalanced data sets,” *Journal of Central South University (Science and Technology)*, vol. 50, no. 5, pp. 1112–1118, 2019.

[11] D. Hong, L. Balzano, and J. A. Fessler, “Asymptotic performance of PCA for high-dimensional heteroscedastic data,” *Journal of Multivariate Analysis*, vol. 167, pp. 435–452, 2018.

[12] K. Mens, E. Elzinga, and M. Nielen, “Applying machine learning on health record data from general practitioners to predict suicidality,” *Internet Interventions*, vol. 21, Article ID 100337, 2020.

[13] V. A. Sylvester Emma, B. Paul, and R. Bradbury Ian, “Applications of random forest feature selection for fine scale genetic population assignment,” *Evolutionary Applications*, vol. 11, no. 2, pp. 153–165, 2018.

[14] Y. Zhou and G. Qiu, “Random forest for label ranking,” *Expert Systems with Applications*, vol. 112, pp. 99–109, 2018.

[15] B. Gregorutti, M. Bertrand, and S.-P. Philippe, “Correlation and variable importance in random forests,” *Statistics and Computing*, vol. 27, no. 3, pp. 659–678, 2017.

[16] Z. Jin-Hua, “Modeling based on RS and BPNN for credit risk assessment in commercial banks,” *Computer Simulation*, vol. 32, pp. 372–379, 2011.

[17] C. D. Li, H. M. Tang, Y. F. Ge, X. L. Hu, and L. Q. Wang, “Application of back-propagation neural network on bank destruction forecasting for accumulative landslides in the three Gorges Reservoir Region, China,” *Stochastic Environmental Research and Risk Assessment*, vol. 28, no. 6, pp. 1465–1477, 2014.

[18] J. Zhu, A. Wu, X. Wang, and H. Zhang, “Identification of grape diseases using image analysis and BP neural networks,” *Multimedia Tools & Applications*, vol. 79, no. 21-22, pp. 14539–14551, 2020.

[19] C. Min-Rong, C. Bi-Peng, Z. Guo-Qiang, L. Kang-Di, and P. Chu, “An adaptive fractional-order BP neural network based on extremal optimization for handwritten digits recognition,” *Neurocomputing*, vol. 391, pp. 260–272, 2020.

[20] N. Chawla, K. Bowyer, and L. Hall, “Smote: synthetic minority over-sampling technique,” *Journal of Artificial Intelligence Research*, vol. 16, no. 1, pp. 321–357, 2002.

[21] N. Zhao, X. Zhang, and L. Zhang, “Overview of research on unbalanced data classification,” *Computer Science*, vol. 45, no. A1, pp. 22–27, 2018.