Credit Evaluation Model Based on Machine Learning Algorithm

Zhenming Hu

Beijing University of Posts and Telecommunications, Haidian, Beijing

Keyword: Logistic regression; KNN; GaussainNB; credit evaluation model

Abstract: This paper adopts the machine learning algorithm to construct a scientific credit scoring method for the credit evaluation of commercial bank customers, and combines the results of the model to analyze the pros and cons of the algorithm. In the empirical analysis of this paper, the data preprocessing is first carried out. Refer to the relevant literature to screen out some variables to participate in modeling. Then the missing value processing and the outlier filling are performed. After that, this paper discusses three algorithms, logistic regression, KNN and GaussianNB, and their built models. The modeling results show that the KNN algorithm takes a long time and is not recommended to be applied under actual conditions. The logistic regression model has applicable conditions. Over-fitting will occur when the conditions are not met. The GaussainNB algorithm performs best.

1. Introduction

The personal credit rating system is an important part of the national credit system and the basis for financial institutions such as banks to conduct credit business. In western developed countries, the personal credit system has a history of more than 160 years. The personal credit rating system of these countries has been developed and improved. In the aspects of national economic growth and social life, especially for commercial banks, personal consumer credit business is carried out. It played an important role. At this stage, the construction of the personal credit rating system not only provides a good market operation order, but also has great significance for China to expand domestic demand and stimulate economic growth.

There are many algorithms for establishing a personal credit scoring model, such as discriminant analysis, logistic regression, and neural networks. Based on the existing data, this paper uses four algorithms for prediction and analysis. The research focuses on how to build personal credit information through information mining to establish a highly credible personal scoring model, to achieve a combination of theory and practice, and to establish a personal credit scoring model with practical application significance.

2. Data acquisition and preprocessing

2.1. Source of data

Considering the difficulty of obtaining real customer credit data from commercial banks, the data set used in this paper is the open source data set “Give Me Some Credit” provided by the Kaggle Data Algorithm Competition. The acquired data set includes a training set and a test set, as well as an explanatory table of the variables. The competition uses the predictive accuracy of the entrant's program on the test set as a basis for scoring. Since the competition is over and cannot be submitted, only the training set is used to build the model, and the cross-validation method is used to test the accuracy of the model.

2.2. Missing value processing

Data preprocessing includes padding of missing values and handling of outliers. First, the indicators of the original data set are renamed from top to bottom in the order of Table 2 as "y", "x1", "x2", "x3", "x4", "x5", "x6", "x7". , "x8", "x9", "x10", and then use the md.pattern() function of the m package in R to observe the distribution of missing values. The result is shown in Figure 1.

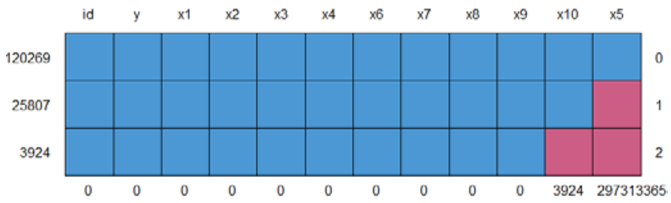


Figure 1 Distribution of missing values in the original dataset

As can be seen from Figure 1, the missing values are concentrated in two columns x5 and x10. The most common method used here is to fill in the average of the columns in which the missing values are located.

3. The establishment of the model

Model establishment

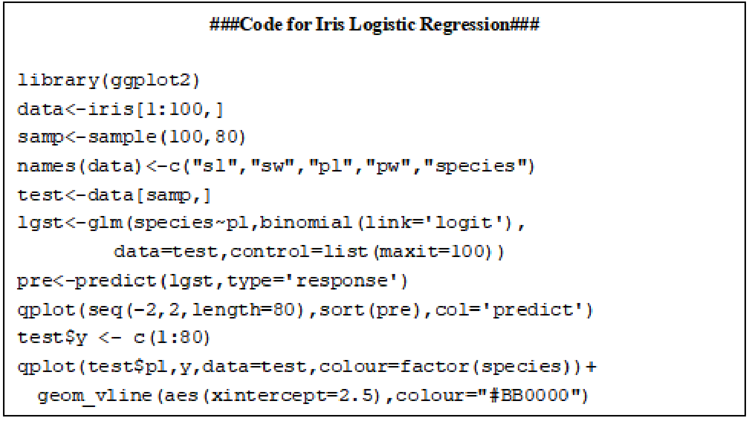
3.1. Logistic regression model

Use the glm () function of R to build a logistic regression model. The full name of the glm() function is called "Generalized Linear Model". After the model is established, the corresponding training error and test error are obtained by the method of ten-fold cross-validation.

In the process of model building, R will report two errors. The first error is "Warning: glm.fit: algorithm did not converge", that is, the logistic regression algorithm does not converge. The second error is "Warning: glm. Fit: fitted probabilities numerically 0 or 1 occurred", that is, the fitting probability is calculated to be 0 or 1. The reasons for these two errors were analyzed by consulting relevant information.

The reason for the first error is that the number of iterations of the algorithm is not enough. Since the regression coefficient is iteratively solved according to the principle of maximum likelihood estimation when performing logistic regression, the default maximum number of iterations of the glm() function is “maxit=25”. When the data is not very good, the algorithm may not converge after 25 iterations, so you can try to solve the problem that the algorithm does not converge by increasing the number of iterations.

The reason for the second error is that the data processed is very good, with obvious classification boundaries, resulting in a model with a probability of non-zero, which is different from the logistic regression curve we have in mind. We replaced the dataset to illustrate the problem, using the iris (Iris) dataset in R as an example to create a logistic regression model.



###Code for Iris Logistic Regression###

```
library(ggplot2)
data<-iris[1:100,]
samp<-sample(100,80)
names(data)<-c("sl","sw","pl","pw","species")
test<-data[samp,]
lgst<-glm(species~pl,binomial(link='logit'),
        data=test,control=list(maxit=100))
pre<-predict(lgst,type='response')
qplot(seq(-2,2,length=80),sort(pre),col='predict')
test$y <- c(1:80)
qplot(test$pl,y,data=test,colour=factor(species))+
  geom_vline(aes(xintercept=2.5),colour="#BB0000")
```

Iris is a data set of R that comes with Iris. This data set has "Sepal.Length", "Sepal.Width", "Petal.Length", "Petal." Width (petal width) and "Species" have 5 attributes. "Species" includes "setosa" and "versicolor". We use this data set to explain the reason for the second error.

The predicted results of the model are shown in Figure 2.

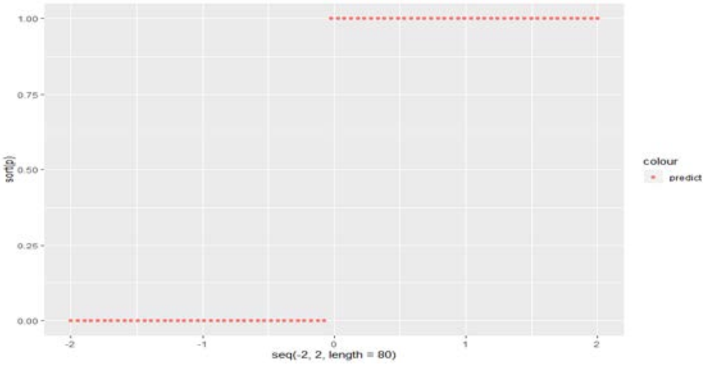


Figure 2 Iris data set logistic regression prediction results

As seen from FIG. 2, after the data set is replaced, a case where the prediction probability shown in FIG. 2 is not 0 or 1 may occur. The reason for this is that the independent variable x is completely linearly separable with respect to the probability value y. In this case, it is no longer necessary to require the training error and test error of the logistic regression model. The following figure shows the scatter plot of the Petal Length property of Iris. It can be seen from the figure that the two irises can be completely classified by the property Petal Length, that is, the training error is 0, resulting in over-fitting.

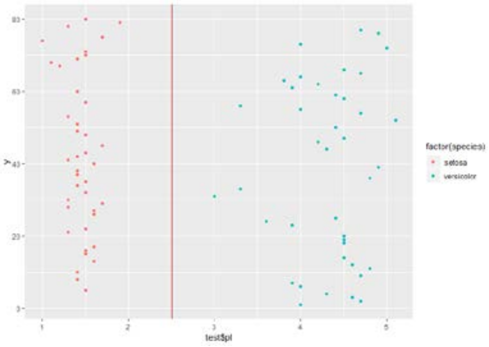


Figure 3 Petal Length scatter plot in the Iris dataset

3.2. KNN model

When establishing the KNN model, the choice of parameter k should satisfy the minimum error of the model on the test set, or reach the local minimum. The process of parameter selection is called tuning parameter selection. The method of selecting parameters here is cross-validation. In the process of this modeling, the range of candidate k is 1 to 10, and the test error corresponding to k is plotted by R, as shown in FIG 4.

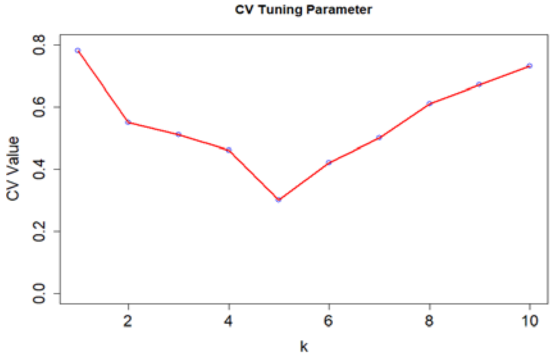


Figure 4 KNN tuning parameter selection

As can be seen from the above figure, the CV value is minimized when k=5, so the parameter k should be set to 5 when establishing the KNN model. The KNN model is built using the kknn() function in the kknn package of R, and the corresponding training error and test error are output. The average value of the training error is 0.095, and the average value of the test error is 0.089.

The KNN algorithm takes a long time in the modeling process, so it should be fully considered in the actual application process.

3.3. GaussainNB model

To build the GaussainNB model, you need to use the naiveBayes() function in the e1071 package. The boxing plot of the output training error and test error is shown in Figure 5.

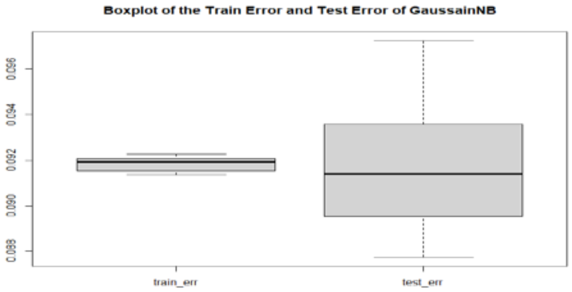


Figure 5 Training error and test error of GaussianNB model

4. Suggestions for the credit scoring system

Through the use of KNN, Logistic Regression, and GaussianNB to model personal credit data, it is found that several models have certain ability to distinguish between good and bad customers, and have certain reference for commercial banks' credit decision. First of all, when using the logistic regression algorithm, the phenomenon of overfitting appears, indicating that the sample is completely separable about a certain attribute. This situation is undesired in the modeling process, because the over-fitting of the model will lead to weaker generalization of the model, and then the nature of the data and the selection of the algorithm need to be reconsidered. Secondly, although the prediction ability of KNN algorithm is strong, it can also improve the model by continuously adjusting the parameter configuration, and improve the prediction accuracy. However, due to its high complexity and long running time of the program, the KNN algorithm is practically applied. Feasibility is low in the process. Finally, under the condition that the KNN and GaussainNB algorithms have high prediction accuracy and similar accuracy, we conclude that the GaussianNB algorithm is optimal.

References

[1] Yang Longguang, Wu Jingmei.A New Credit Rating Method Based on Statistics and Clustering[J].Statistics & Decision,2016(03):13-16.

[2] Zhang Lina, Zhao Min. Analysis of Personal Credit Score Index System of China's Commercial Banks[J].Marketing Weekly (Theory),2007(08):115-117.

[3] R in Action：Data Analysis and Graphics With R by Robert I. Kabacoff, published by Manning Publications. 178 South Hill Drive, Westampton, NJ 08060 USA. Copyright © 2011 by Manning Publications.