RMSC4002 GROUP PROJECT
2016-17 FALL
CHINESE UNIVERSITY OF HONG KONG

# Models for Predicting Credit Card Default


| | |
|---|---|
| CHU Jing | 1155047032 |
| JIIAN Xiaoxue | 1155029058 |
| XIE Peijie | 1155047076 |
| ZHANG Puchen | 1155046891 |



## Abstract

This research investigated on the case of customer's default payments in Taiwan, and aimed at comparing the accuracy of predicting the probability of default among four data mining methods based on the personal information and bank statement of each customer. Before fitting the models, outliers were removed and PCA was applied to reduce the dimension of explanatory variables according to the correlation between them. Then Classification Tree, Binary Logistic Regression and Artificial Neural Network were applied and compared for predicting the probability of default.

# 1. Introduction

The traditional risk of banking industry is credit risk and managing credit risk has been a key part of the banking business. In recent years, to increase market share, the credit card issuers over-issued credit cards to many unqualified applicants. On the other hand, many credit card holders overused credit card regardless of their repayment ability. It is a necessity for banks to decide whether a client would default his credit card payment.

In this project, historical data in Taiwan was analyzed. We implemented various classification methods including Classification Tree, Logistic Regression, and Artificial Neural Network. In Section 2 we described our dataset and how we initially dealt the data before applying the classification methods. In Section 3 we described our methodology and their outputs. In Section 4 we compared previous methods and gave our conclusion and discussions.

# 2. Data

## 2.1 Data Description

We used the payment data from a bank in Taiwan in October, 2005. The dataset was downloaded from UCI Machine Learning Repository website. This dataset used 23 variables as explanatory variables and a binary variable to indicate whether the client defaulted. There were 30000 observations in total and the following table (Table 1) showed the variable names, their categories and meaning.

| Variable Name | Category | Description |
|---|---|---|
| LIMIT_BAL | Numerical | Amount of the given credit |
| SEX | Categorical (Nominal) | Gender (1 = male; 2 = female) |
| EDUCATION | Categorical (Ordinal) | Education (1 = graduate school; 2 = university; 3 = high school; 4 = others) |
| MARRIAGE | Categorical (Nominal) | Marital status (1 = married; 2 = single; 3 = others) |

| | | |
|---|---|---|
| AGE | Numerical | Age (year) |
| PAY_0 - PAY_6 | Categorical (Ordinal) | Historical payment. PAY_0 = status in 09/2015; ...; PAY_6 = status in 04/2015. -1 = pay duly; 1 = 1 month payment delay; ...; 9 = 9 months and more payment delay. |
| BILL_AMT1-<br><br>BILL_AMT6 | Numerical | Bill statement amount. Bill_AMT1 = amount in 09/2005; ...; X17 = amount in 04/2005. |
| PAY_AMT1-<br>PAY_AMT6 | Numerical | Previous payment amount. PAY_AMT1 = amount paid in 09/2005; ...; PAY_AMT6= amount paid in 04/2005. |

Table 1

### 2.2 Outlier Detection

Before constructing the predict model, the first step was to use Mahalanobis distance to detect outliers. We chose the 99% quantile from the chi-square distribution as the cut-off value (See Appendix Figure 1). Finally, 27428 observations were left (See Appendix Figure 2). Note that many important clients who had huge bill or payment amount may be categorized as outliers. Those clients would be very crucial for the bank. However, we believed that those "key clients" required special models, which we would not discuss here.

### 2.3 Principal Component Analysis

Principle Component Analysis (PCA) is a traditional multivariate statistical procedure to reduce the number of possibly correlated variables and retain information as much as possible. It uses orthogonal transformation to find linear combinations of original variables called principal components so that each component has the highest possible variance and are orthogonal to each other.

The original dataset contained 23 explanatory variables, among which 18 variables recorded monthly historical value of three indicators for six months. The three indicators were repayment status, amount of bill statement and amount of payment. It

is reasonable to conjecture that the monthly data was correlated to each other and could be represented by fewer variables without loss of large amount of information.

The correlation relationship among these 18 variables was illustrated using correlation heatmap (See Appendix Figure 3). Although no strong relationship was detected among monthly data in terms of repayment status or amount of payment, it is clear to see that the six variables regarding amount of bill statements were indeed highly correlated with correlation close to one.

Variable reduction was performed on the six variables denoting monthly bill statement using PCA. The loadings of the six principal components were shown in Appendix Table 1. Particularly, the loadings of first three components were plotted against months (See Appendix Figure 4-6). For the first principal component, the loadings stayed stable at around -0.41 for each month recorded. This denoted a parallel component and indicated that months had equally-weighted effect on the amount of bill statement. It also explained major variance existing in the six variables. In comparison, the loadings of the second principal component increased with month while the loadings of the third component decreased with months to some point and then started to increase. They denoted tilt component and curvature component of the relationship between months and the amount of bill statement respectively.

Furthermore, Appendix Table 2 illustrated the cumulative proportion of variance. The scree plot (See Appendix Figure 7) indicated how much information was preserved by each component and could be used to determine the suitable number of principal components to use. From the scree plot, we could just use the first principal component to represent all the six variables, which explained 95.6% of all variances. The six variables were then reduced to one variable named *BILL_AMT* by linear transformation using the loadings of the first principal component.

### 2.4 Variable Scaling

The magnitude of variables varied significantly. Therefore, we scaled our data before constructing models. Range scale method was applied for *PAY_0* to *PAY_6* (ordinal categorical data). Standardize score method was applied for *BILL_AMT1* to *BILL_AMT6* and *PAY_AMT1* to *PAY_AMT6* (numerical data).

### 2.5 Training and Testing Dataset

After deleting outliers, the total number of data was 27428. We randomly chose 20000 (around 75%) data as training data, and the rest 7428 records were testing data. Training dataset was used to build the classification model while testing dataset was used to measure the predicting power of the model.

## 3. Methodology and Output

### 3.1 Model Evaluation

Error rates were usually used when measuring the classification accuracy of models. But in this case, most credit card customers were non-risky; it is obviously not a good way to use the error rate. Therefore, area-ratio in lift chart, which provided a better explanation, was defined to measure and compare the performance.

Definition of Area Ratio:

$$\text{Area Ratio} = \frac{Area\ between\ Model\ curve\ and\ Baseline}{Area\ between\ theoretically\ Perfect\ Curve\ and\ Baseline}$$

There are two kinds of lift charts, the cumulative percentage one was chosen for our analysis. The horizontal axis represented the percentage of total number of data, where the vertical axis showed the cumulative proportion of target data. Model curve, theoretically perfect curve and baseline could be found in the chart. Apparently, greater area between baseline and model curve implied better accuracy. The closer the ratio is to one, the better the model fits.

### 3.2 Classification Tree

The Classification Tree method is based on the binary splitting of variables, aiming to minimize impurity. Impurity is measured by Entropy, Gini or Classification Error, etc. In the classification tree, each internal node represents a splitting criterion and each leaf node represents the predict class. Classification Tree could provide with a simple classification rule and deal with nonlinear predictions.

We had reduced the variable's dimension by PCA. We then applied Classification Tree based on the dimension-reduced data. From the outputs, we could observe that the depth of this tree was only one and the criterion was whether the normalized *PAY_0* was less than 0.4375 (See Appendix Figure 8). That is, we predicted a client would not default if his payment status in last month was paid on time or payment delayed for less than one month and we predicted a client would default if his payment status in last month was delayed larger than or equal to two months. To visualized this criterion, we plotted a stacked bar chart (See Appendix Figure 9-10). From the chart, if their *PAY_0* was less than 0.4375, the ratio of default and no default would be much less than those whose *PAY_0* was larger than 0.4375. If the credit card owner did not pay his debt last month, he was prone to continue default in this month, which seemed logical.

We measured the predict accuracy by the classification table and measure of accuracy (See Table 2). The performance of training dataset and testing dataset were very similar. From the F1 scores, we could see that the predict error for those default clients was comparable large. Therefore, we need to further improve our model.

| **Classification Table for Training Data** | | | |
|---|---|---|---|
| **Test \ True** | **Default** | **No Default** | **Row Sum** |
| **Default** | 1433 | 630 | 2063 |
| **No Default** | 2997 | 14940 | 17937 |
| **Column Sum** | 4430 | 15570 | 20000 |
| **Precision = 0.6946; Recall = 0.3235; F1 = 0.4414** | | | |

| **Classification Table for Testing Data** | | | |
|---|---|---|---|
| **Test \ True** | **Default** | **No Default** | **Row Sum** |
| **Positive** | 542 | 213 | 755 |
| **Negative** | 1134 | 5539 | 6673 |
| **Column Sum** | 1676 | 5725 | 7428 |
| **Precision = 0.7179; Recall = 0.3233; F1 = 0.4458** | | | |

Table 2

### 3.3 Binary Logistic Regression

Logistic Regression is a regression model where the dependent response variable is categorical. The binary regression model, which we applied in the data, is the case that response variable can only take two values. The major advantage of this approach is that normality assumptions are not necessary and simple linear formula relating the explanatory variables and the probability of success of response variable.

There were 18 explanatory variables, where two of them (*SEX* & *MARRIAGE*) were treated as factors. In the result of Classification Tree, $PAY\_0 > 0.4375$ was noticed as an important rule to classify $Y = 0$ or $1$. So a categorical variable $g$ was created based on $PAY\_0$, $g = 1$ if $PAY\_0 < 0.4375$ and $g = 2$ if $PAY\_0 > 0.4375$. Then $g$, as well as all the interaction terms between $g$ and other non-category variables, were used to fit a logistic regression model. Stepwise elimination was performed to reduce the number of variables using AIC.

The final model could be found in Appendix. The classification table and measure of accuracy were shown in Table 3. The cumulative percentage of success of training and testing data was plotted in the lift charts (See Appendix Figure 11-12). The training area ratio was 0.7799 with F1 score of 0.4371 and the testing ratio was 0.7741 with F1 score of 0.4410.

| **Classification Table for Training Data** | | | |
|---|---|---|---|
| **Test \ True** | **Default** | **No Default** | **Row Sum** |
| **Default** | 1407 | 601 | 2008 |
| **No Default** | 3023 | 14969 | 17992 |
| **Column Sum** | 4430 | 15570 | 20000 |
| **Precision = 0.7007; Recall = 0.3176; F1 = 0.4371** | | | |

| **Classification Table for Testing Data** | | | |
|---|---|---|---|
| **Test \ True** | **Default** | **No Default** | **Row Sum** |
| **Positive** | 533 | 208 | 741 |
| **Negative** | 1143 | 5544 | 6687 |
| **Column Sum** | 1676 | 5752 | 7428 |
| **Precision = 0.7193; Recall = 0.3180; F1 = 0.4410** | | | |

Table 3

### 3.3 Artificial Neural Network

Artificial Neural Network (ANN) aims to detect the relationship between inputs and outputs based on mimicking the neural structure of our brain. In a feed-forward network, artificial neurons are connected from input layer to hidden layers, and then to output layer to form a network. Back-propagation algorithm is used through the learning process. ANN can easily denoise the complex relationships among explanatory variables.

Scaled data using standardized score method in training dataset was feed into the ANN so that each 18 input had approximately equal importance. We first fixed the number of hidden layer as one and applied different levels of size of hidden layer varying from 6 to 36, with the maximum size twice as large as the input layer. It is expected that the capacity of the network to recognize patterns increased with the number of neurons in the hidden layer in terms of training dataset. On the other hand, larger size of hidden layer required much more parameters, raising the risk of over-fitting and poorer prediction performance. To avoid the problem, we monitored the area ratio of both training and testing dataset as we increased the size of hidden layer. Appendix Figure 13 showed the trend and helped us identify the suitable number of neurons in hidden layer. It was in concordance with the fact that more complicated model fitted training dataset better. However, the testing area ratio became volatile and displayed a downward trend when the hidden layer size exceeded 12, which gave a warning sign of over-fitting. Therefore, it is plausible that 12 was a suitable candidate for the size of the single hidden layer. Consequently, we chose an 18-12-1 ANN model.

The structure of the 18-12-1 ANN model was shown in Appendix Figure 14, where the brown lines were positive weights while the blue lines were negative weights. Moreover, the widths of lines were in proportion to the corresponding weights. The number of weights or parameters in this model was 241, which accounted for 1.205% of the size of the training dataset. Area ratio was 0.7310 for training dataset and 0.7023 for testing

dataset respectively. Classification table was shown in Table 4. And lift charts for both dataset were shown in Appendix Figure 15-16.

| Classification Table for Training Data | | | |
|---|---|---|---|
| **Test \ True** | **Default** | **No Default** | **Row Sum** |
| **Default** | 1724 | 705 | 2429 |
| **No Default** | 2706 | 14865 | 17571 |
| **Column Sum** | 4430 | 15570 | 20000 |
| **Precision = 0.7098; Recall = 0.3891; F1 = 0.5027** | | | |

| Classification Table for Testing Data | | | |
|---|---|---|---|
| **Test \ True** | **Default** | **No Default** | **Row Sum** |
| **Positive** | 576 | 321 | 897 |
| **Negative** | 1100 | 5431 | 6531 |
| **Column Sum** | 1676 | 5752 | 7428 |
| **Precision = 0.6421; Recall = 0.3437; F1 = 0.4477** | | | |

Table 4

Both the area ratio and F1 scores indicated that the 18-12-1 ANN model performed relatively poorly on the testing dataset, although it fitted better on the training dataset compared with previous classification methods.

## 4. Model Comparison

In this research, Classification Tree, Logistic Regression and Artificial Neural Network were used to fit the data. Since the result of Classification Tree was interpreted as a dummy variable *g* in Logistic Regression, it was not considered into comparison.

The lift charts of Logistic Regression and ANN had similar trend. To compare, the detailed area ratio and F1 scores were shown in Table 5 as below.

**Area Ratio Comparison**

| Data \ Model | Logistic Regression | ANN |
|---|---|---|
| **Training** | 0.4744 | 0.6156 |
| **Testing** | 0.4824 | 0.5047 |

**F1 Score Comparison**

| Data \ Model | Logistic Regression | ANN |
|---|---|---|
| **Training** | 0.4371 | 0.5027 |
| **Testing** | 0.4410 | 0.4477 |

Table 5

It can be seen from the table that Artificial Neural Network apparently had a greater area ratio as well as the F1 score. When it came to testing data, they had similar accuracy. To sum up, Artificial Neural Network basically performed better than Logistic

Regression and therefore was the best one in the three models used.

## 5. Limitation

Although Artificial Neural Network showed a better explanation than other models, the area ratio and F1 score were still relatively unsatisfying. There were two possible reasons. First, the data found has limited range of predictors. Some important parameters such as personal income may also influence the response variable. Besides, in practical case, one single model is perhaps not sufficient to thoroughly detect the underlying structure of the whole dataset. Especially in banking industry, VIP customers with significant amount of money should be analyzed separated from others.

## Reference:

Johnson, R.A., Wichern, D.W. (2007). Applied Multivariate Statistical Analysis (6th Ed.) Prentice Hall.

Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., &Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. Journal of the Operational Research Society, 54(6), 627–635.

Yeh, I. C., & Lien, C. H. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, *36*(2), 2473-2480.

**Data Source:**

UCI Machine Learning Repository (Dec 2016). *default of credit card clients Data Set* Retrieved from https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients

# Appendix

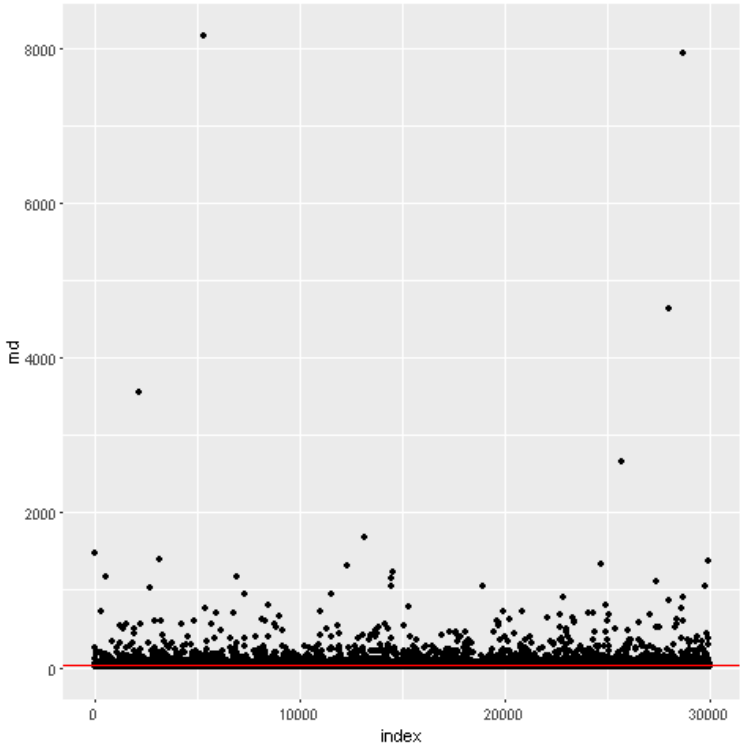


Figure 1

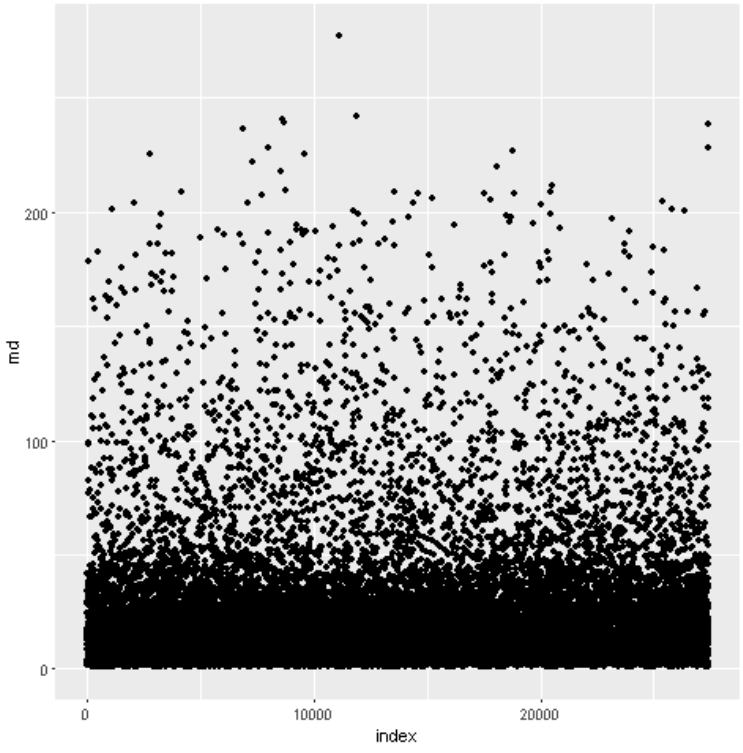


Figure 2

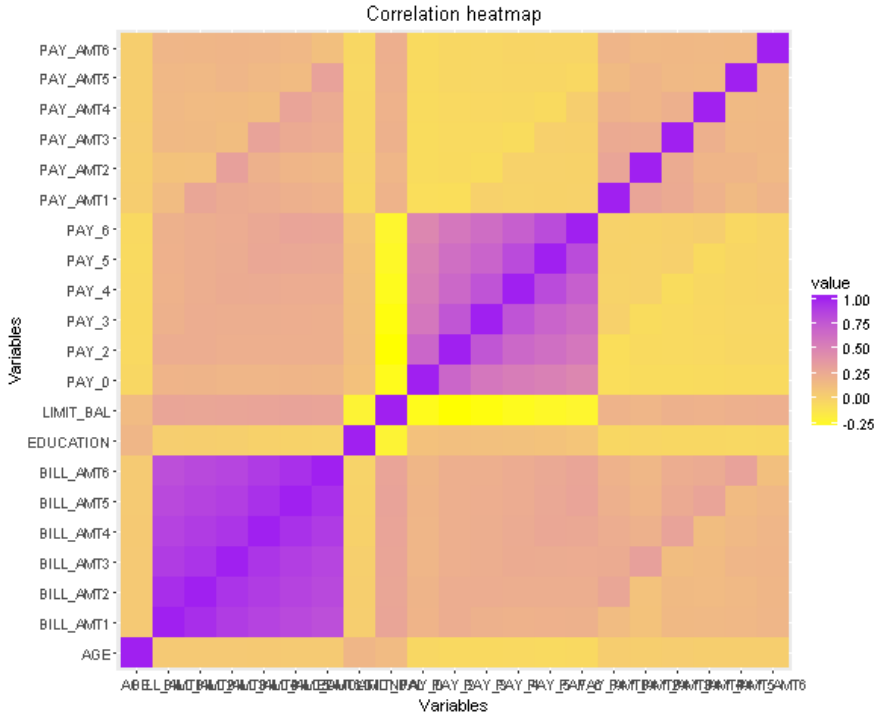


Figure 3

**Loadings for PCA analysis**

| | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 |
|---|---|---|---|---|---|---|
| **BILL_AMT1** | -0.404 | -0.541 | 0.487 | 0.379 | 0.393 | 0.000 |
| **BILL_AMT2** | -0.409 | -0.422 | 0.000 | -0.308 | -0.708 | 0.238 |
| **BILL_AMT3** | -0.412 | -0.175 | -0.552 | -0.489 | 0.424 | -0.277 |
| **BILL_AMT4** | -0.412 | 0.175 | -0.515 | 0.610 | 0.000 | 0.401 |
| **BILL_AMT5** | -0.409 | 0.432 | 0.134 | 0.161 | -0.320 | -0.707 |
| **BILL_AMT6** | -0.404 | 0.530 | 0.415 | -0.353 | 0.249 | 0.444 |

Table 1

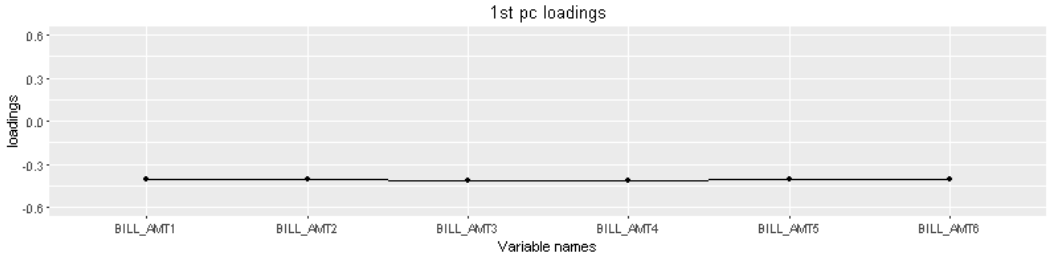


Figure 4

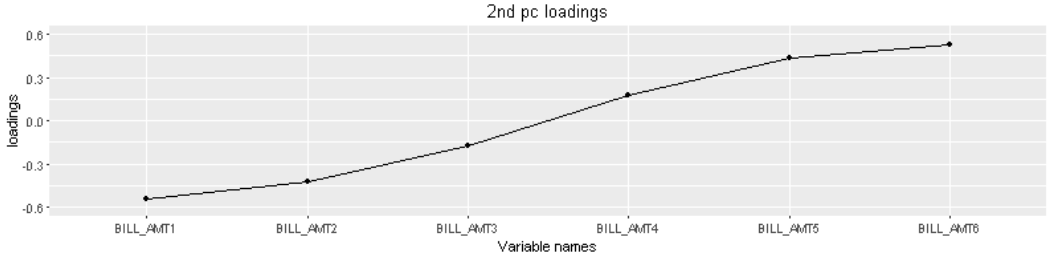


Figure 5

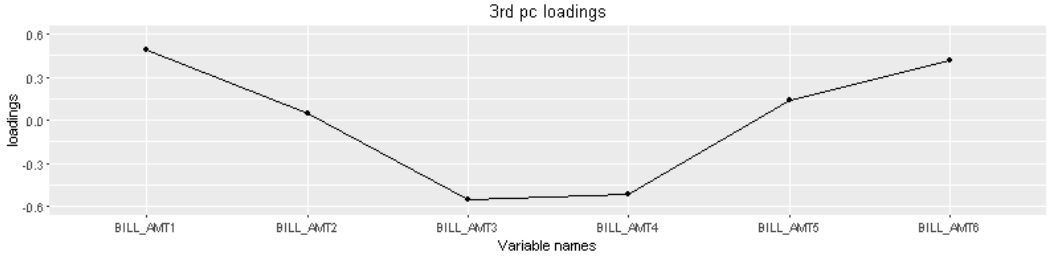


Figure 6

**Cumulative Variance Percentage**

| Comp. 1 | Comp.2 | Comp.3 | Comp.4 | Comp.5 | Comp.6 |
|---|---|---|---|---|---|
| **0.9559** | 0.9848 | 0.9916 | 0.9953 | 0.9977 | 1.0000 |

Table 2

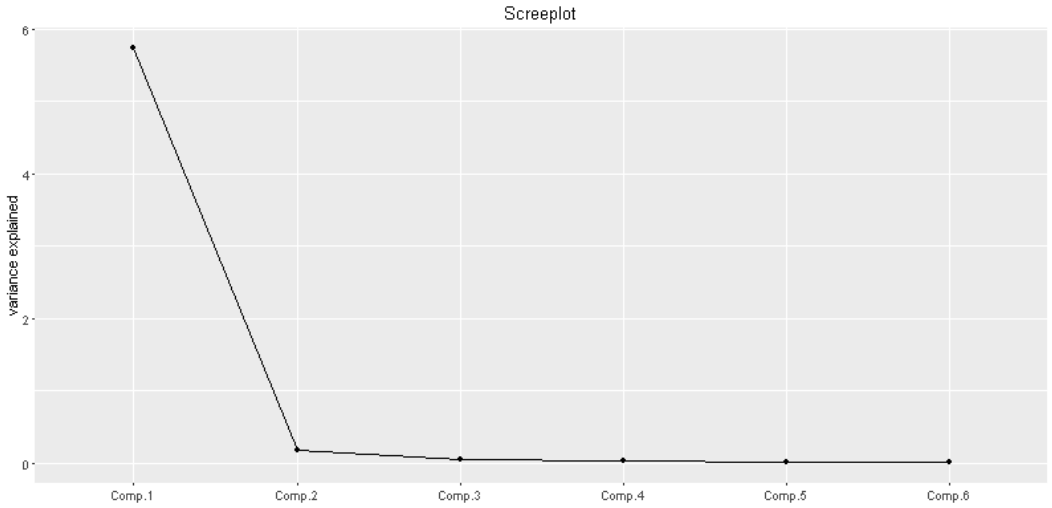


Figure 7

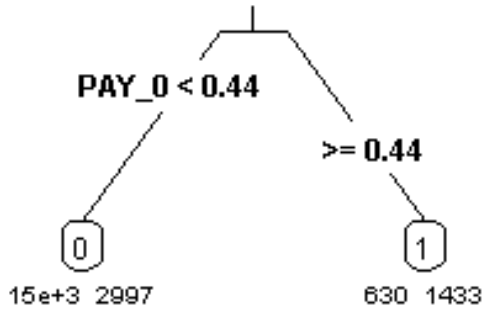


Figure 8

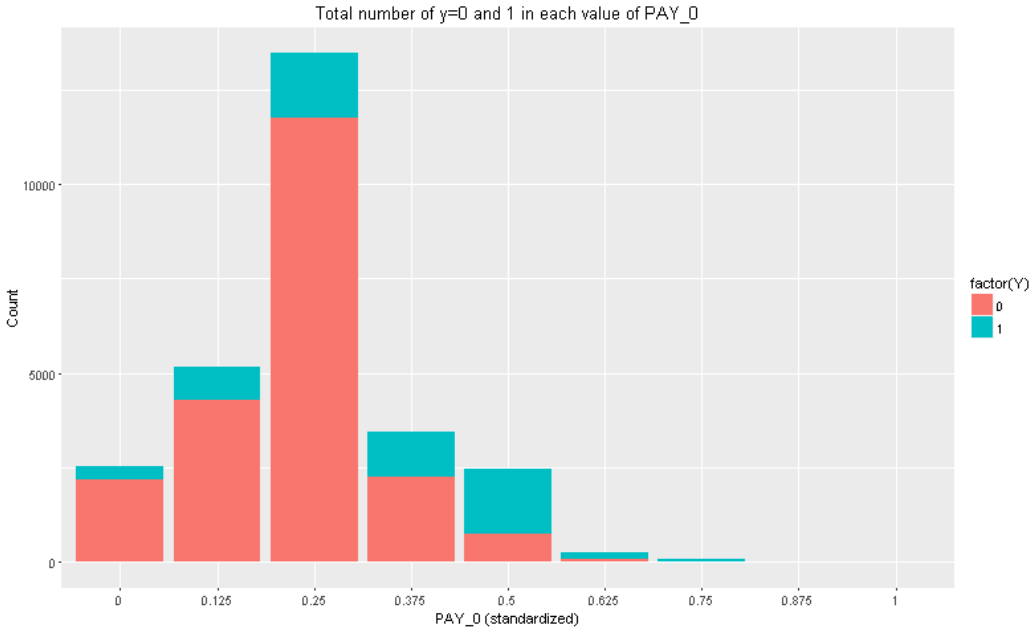


Figure 9

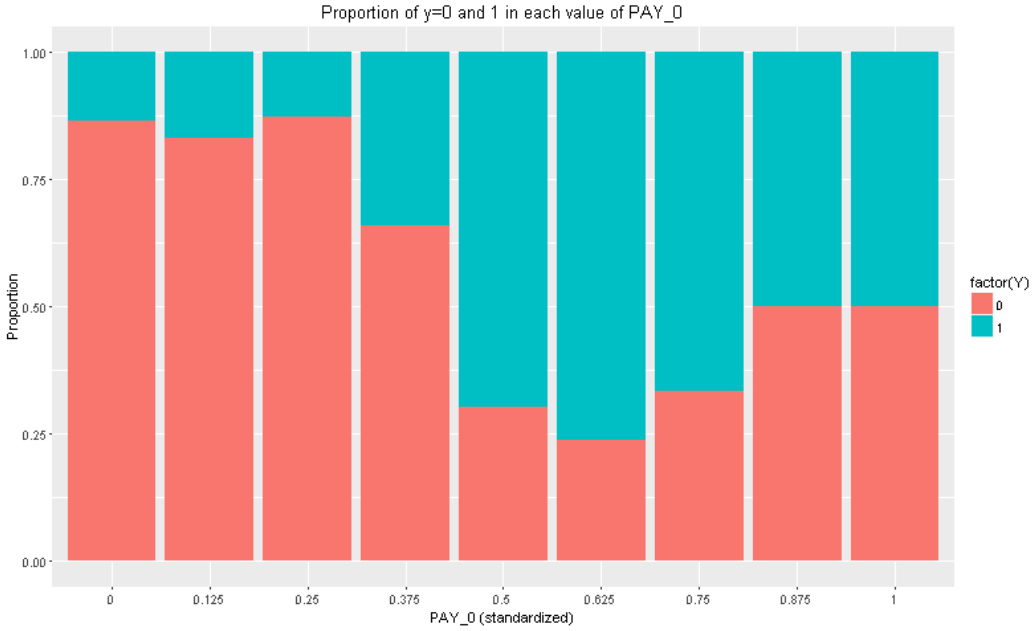


Figure 10

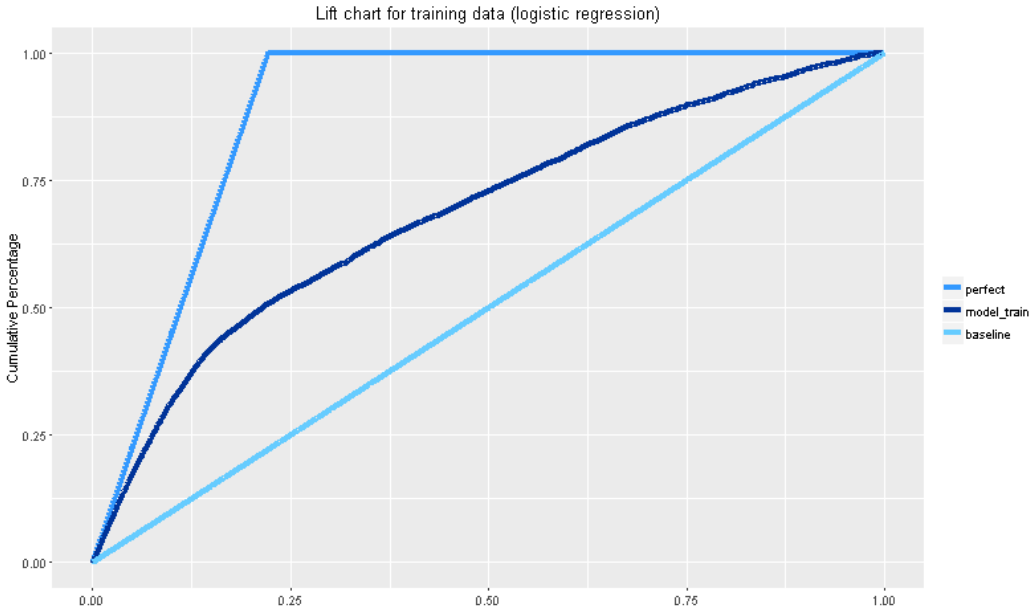


Figure 11

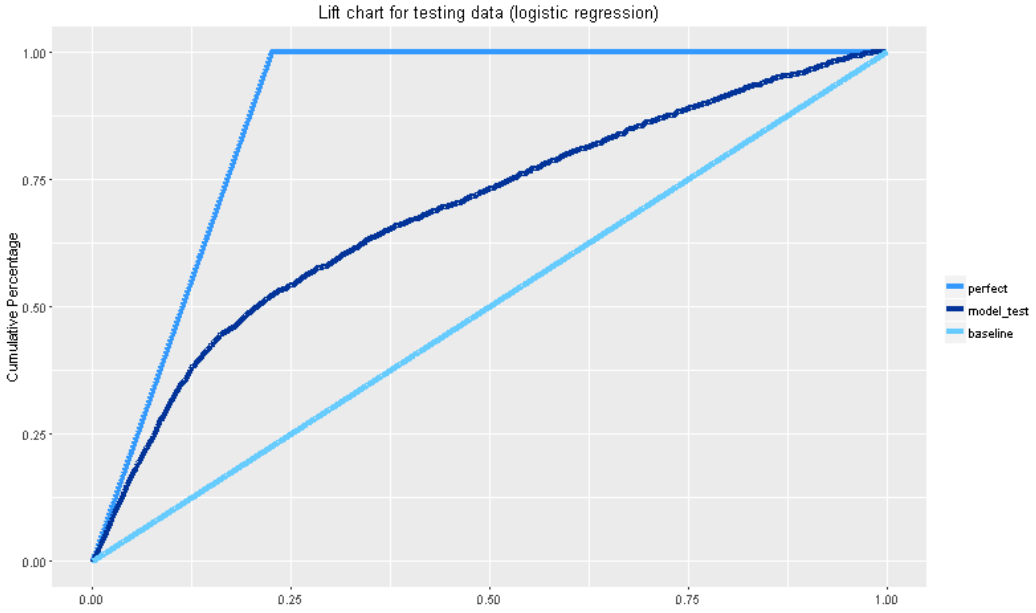


Figure 12


Figure 13

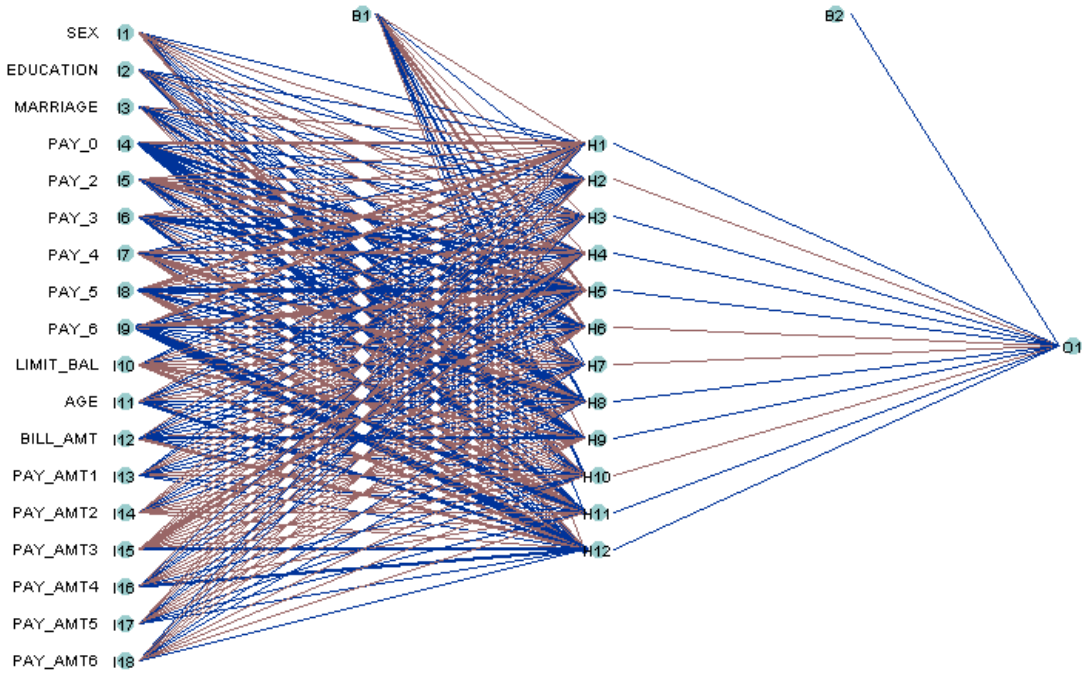


Figure 14

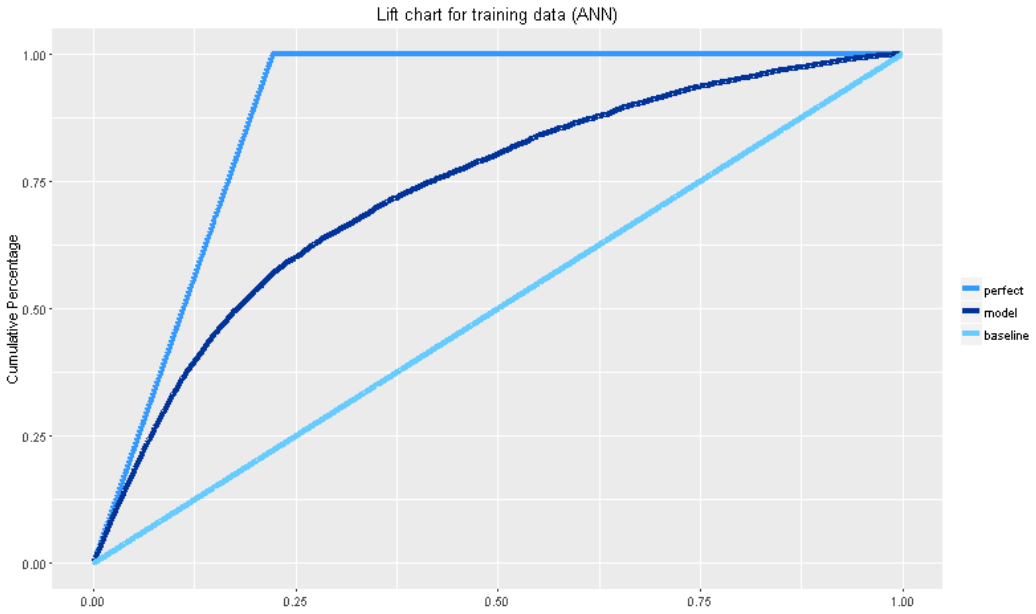


Figure 15

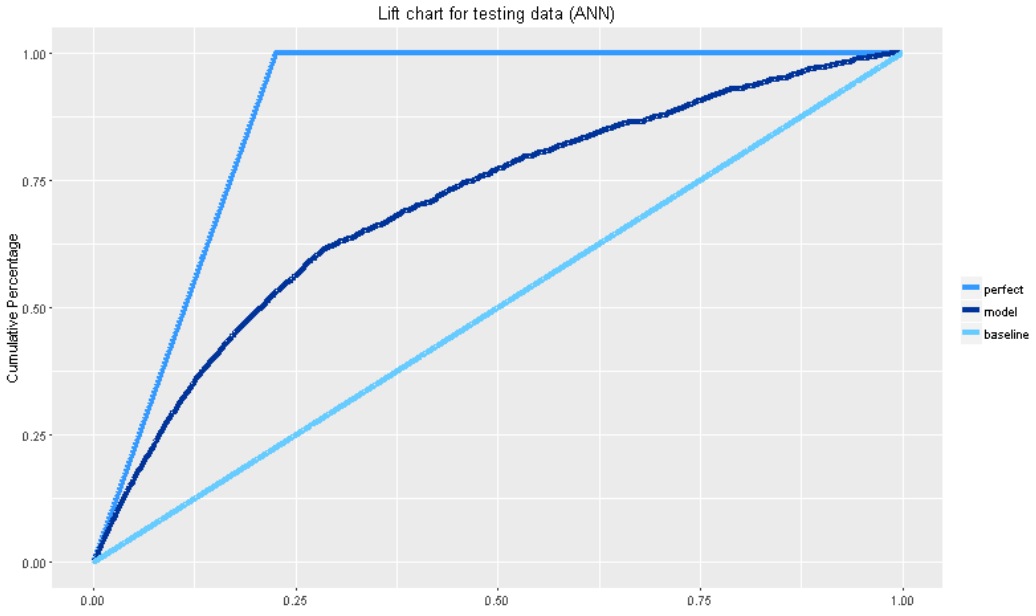


Figure 16

Logistic Regression Final Model:

$ln\frac{\pi}{1-\pi} =$

*-5.26914 -* $\beta^{SEX}$ + $\beta^{MARRIAGE}$ *- 0.12221EDUCATION + 2.93612PAY_2 + 0.43678PAY_3 + 0.48636PAY_4 - 1.71413PAY_6 - 0.11085LIMIT_BAL + 0.03295AGE - 0.35924BILL_AMT - 0.16916PAY_AMT1 - 0.45017PAY_AMT2 - 0.15838 PAY_AMT3 -0.06775PAY_AMT4 -0.10475PAY_AMT5 +-0.06257 PAY_AMT6* + 13.504*g - 1.85664PAY_2*g + 1.59872PAY_6*g +0.27837BILL_AMT*g + 0.23484PAY_AMT2*g*

Where when SEX=1, $\beta^{SEX}$=0, when SEX=2, $\beta^{SEX}$=-0.12356

when MARRIAGE=1, $\beta^{MARRIAGE}$=1.19980, when MARRIAGE=2, $\beta^{MARRIAGE}$=1.02882, when MARRIAGE=3, $\beta^{MARRIAGE}$=1.27113