BT2101 Group Project

Group members: Dong Yizhuo, Han Jiatong, Hong Simeng, Wang Zifei, Yao Xinyi

1. Introduction

The dataset describes 30,000 card holders in Taiwan by their demographic features, payment status, bill statements and default payments from April 2005 to September 2005.

Each data sample is characterized by 23 attributes except its ID and target feature. Among these 23 attributes, there exists 9 categorical variables in which 3 of them are nominal(SEX,EDUCATION,MARRIAGE) while the other 6(PAY_0 to PAY_6) are ordinal. As they are currently interpreted as integers, typecasting might be necessary later on.

The remaining 14 attributes are ratio variables (LIMIT_BAL,AGE,BILL_AMT1-BILL_AMT6,PAY_AMT1-PAY_AMT6). The target feature (default payment next month) is binary valued 0 (= not default) or 1 (= default).

The aim of our project is to evaluate and determine the most robust and accurately tuned model for predicting default value (yes or no) based on the 23 explanatory attributes.

This prediction is useful for banks to plan ahead of whether to give loans to a particular applicant, given his or her demographic information and payment performance, having an idea of its liquidity. Meanwhile, banks can use this model to predict if a client will go default in the future and intervene early to prevent that from happening.

2. Data Preparation

2.1 Remove redundant values and check for missing data

1) Remove the ID attribute

It is obvious that ID Variable has no relation with predicting the target feature, thus it is excluded from further analysis.

2) Rename the attributes

Target feature name is changed to ‘Default’ for simplicity. The attribute name ‘PAY_0’ is changed to ‘PAY_1’ to be more continuous with other attribute names such as PAY_2, PAY_3.

3) Check missing data

There is no missing data across this dataset. Thus, no additional steps are necessary to count for missing data.

2.2 Identify undefined values

(a) MARRIAGE

There is a category 0 undefined in the description of this dataset. Since we do not know the exact marital status of category 0, we combined class 0 with class 3 which indicates an unknown status.

(b) EDUCATION

Category 0, 5 and 6 are undefined and rarely occurred in the dataset thus we merge them into category '4' which later denotes 'Unknown' values.

2.3 Factorize categorical variables

To build the prediction models later on, we changed the attribute types from ‘int’ to ‘factor’ and assigned unique label names for each attribute.

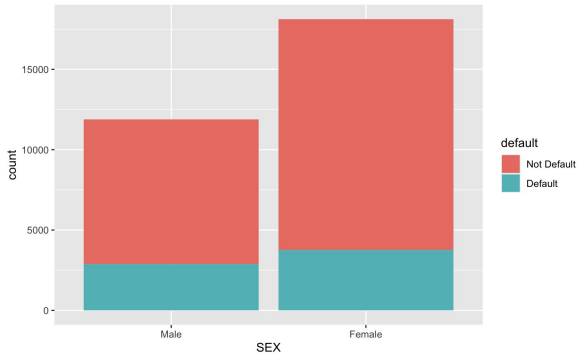
3. Exploratory Data Analysis

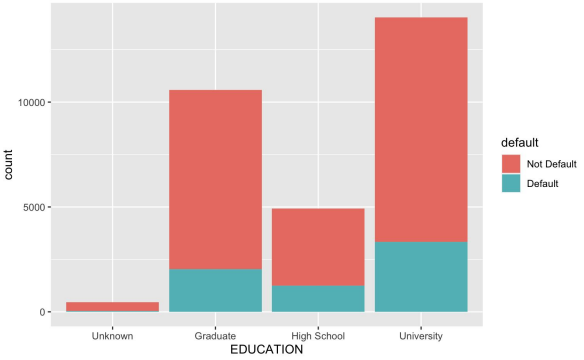
3.1 Target Attribute

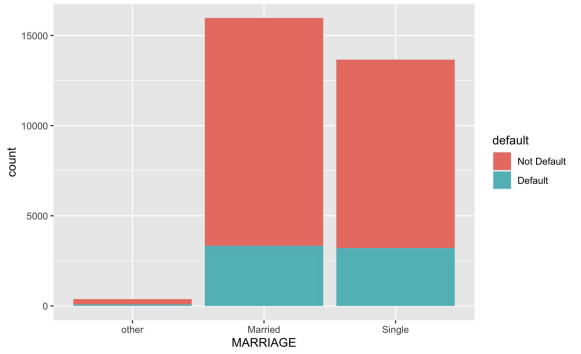
```
Not Default     Default
      23364        6636
[1] 0.2840267
```

The ratio of default payment next month is 0.28 amongst all customer payments, indicating that the dataset is imbalanced and should be considered later in the modeling.

3.2 Demographic variables







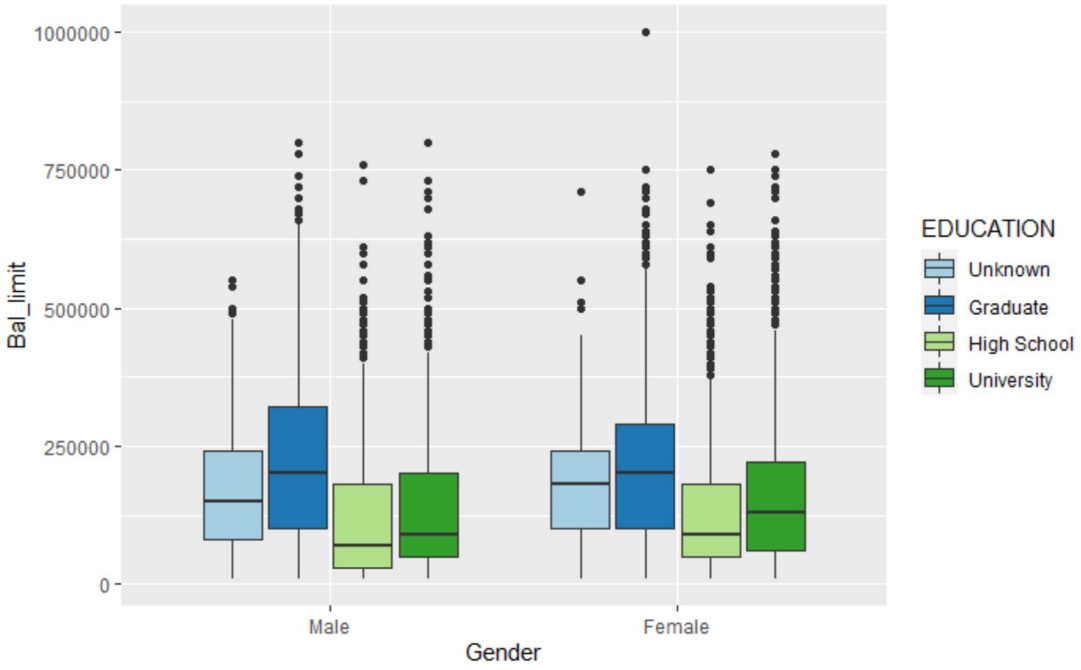

```
Sex:
  Male Female
 11888  18112
Education:
    Unknown    Graduate High School  University
        468       10585        4917       14030
Marriage:
  other Married  Single
    377   15964   13659
```

From the stacked histograms and summary statistics, we notice that the number of female customers far outweigh that of male clients but the number of default clients in two groups only differs a bit, hence the percentage of default clients among each gender implies that male clients are more likely to default. The same is true with marriage: single clients are more likely to default, as compared to their married counterparts.

Moreover, the majority of the clients are found to have a University or Graduate degree, with comparably less number of high school graduates. Even so the proportions of default clients in these groups are roughly the same, which means education level might not be significant.

To further investigate their correlations, we plot them jointly with the Balance limit.

(a) Education, gender and balance limit

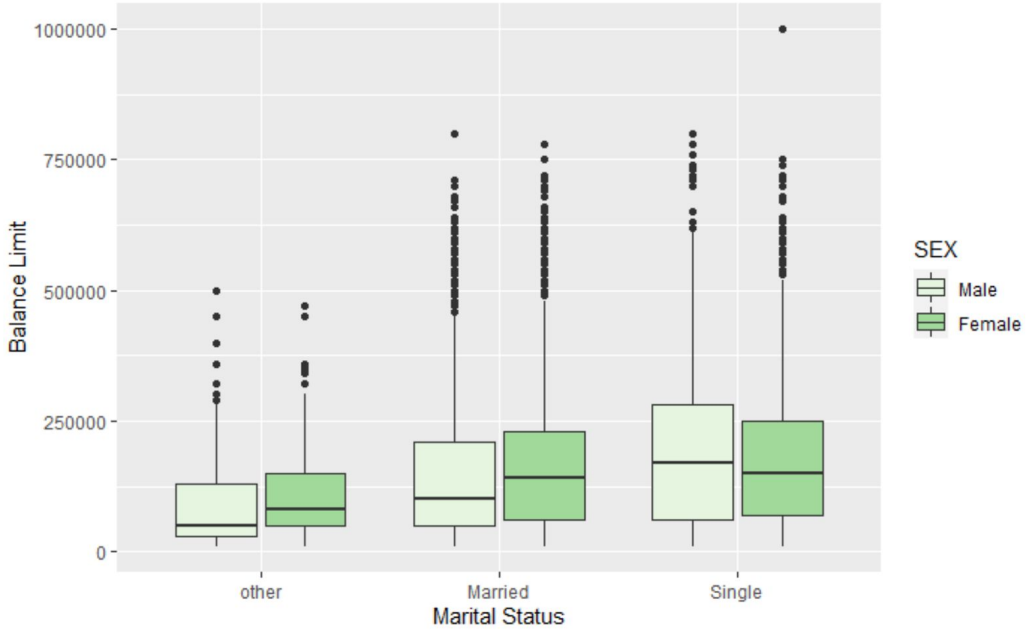


*Due to the lack of information on class 'unknown', we exclude it in our analysis.

To visualize the relationship between education, gender and if they have similar balance limits, we plot a box and whiskers diagram. The results show that graduates have a higher median and upper whisker than others, it also has the widest distribution. Gender seems to not affect the education level of clients. Lastly, more of the outliers are High School or University graduates.

We then do the same visualization for SEX, MARRIAGE and balance limit.

(b) Sex, marriage and balance limit



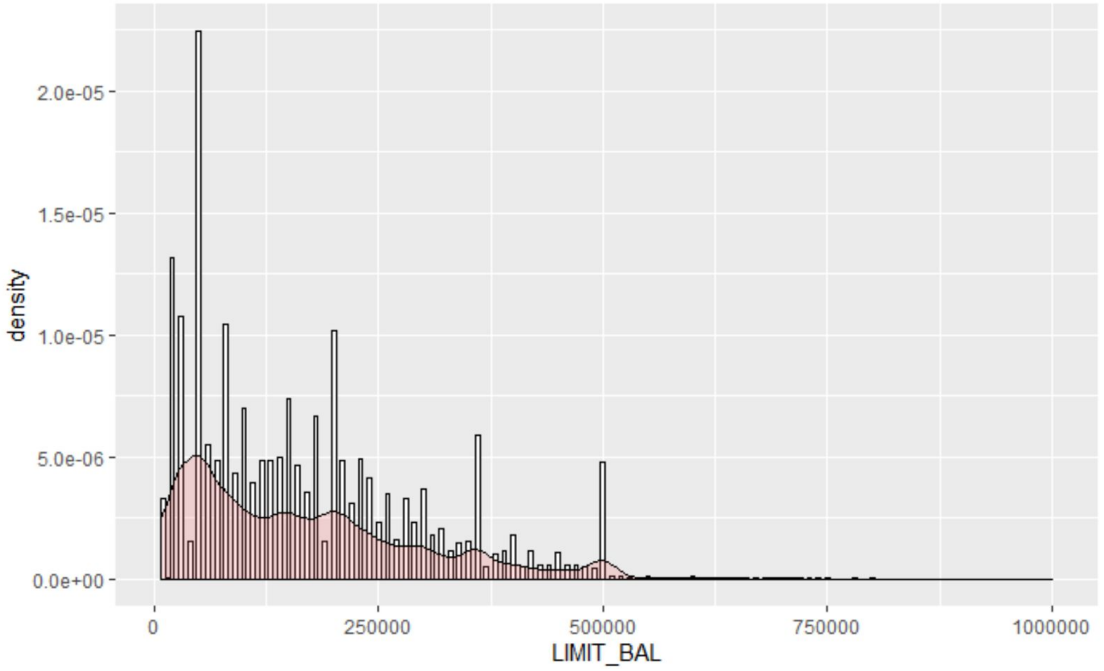
It is observed that single clients have higher median, 1st and 3rd quantile values than married clients for both genders. Furthermore, 'Other' clients have lower quantile values than both married and single clients. A logical explanation may be that ‘Others’ might include divorced or widowed clients. Marriage seems to have a greater effect on males than on females, as observed from the gap between single males and married males. Continued from above, the group of single males have the highest 1st quartile and median value across all groups. Lastly, there are more outliers in balance limits for married clients in general, however, the most deviated outlier is a single female.

In conclusion, single male status and higher educated clients have higher balance limits. Education and balance limits are positively correlated, while marriage is negatively correlated.

3.3 Continuous Variables and their interactions

(a) Balance limit

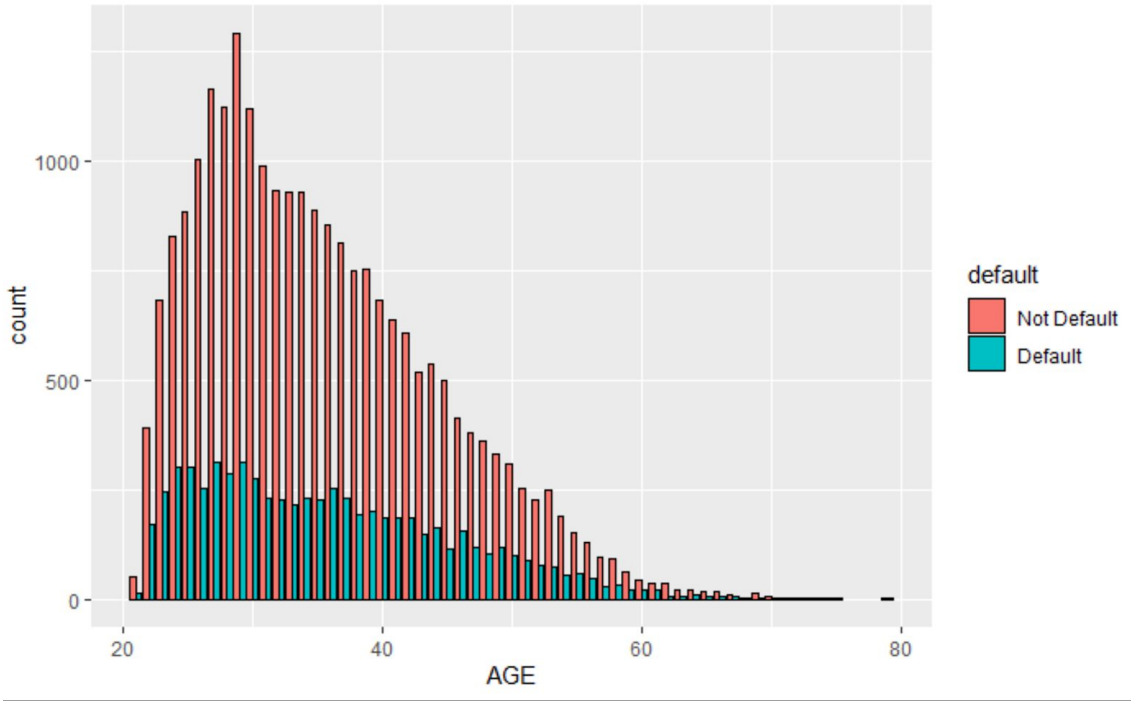
Density Function Graph:



From the graph above, we observed that the balance limit is right skewed with unusually large outlier values (1,000,000). The most aggregated group of clients has approximately 50,000 balance limits. Most of the clients are located in the range from 0 to 500,000. Checking with the analysis done earlier, this is logical since there is one outlier(female graduate) with 1,000,000 balance limit, while the median is closer to the lower extreme, suggesting most of the data points are towards the lower end of balance limits.

(b) Age

Density Function Graph:



Age is also right skewed with the largest age group at around 27-29 years old. Elder clients (>60) are in the minority. Default or not seems to distribute similarly across all ages. The percentage of default clients, the chance of a client going default within his/her age group, is lower for clients at the age of 25-45, roughly less than 50%.

(c) PAY_AMTs and BILL_AMTs

Summary:

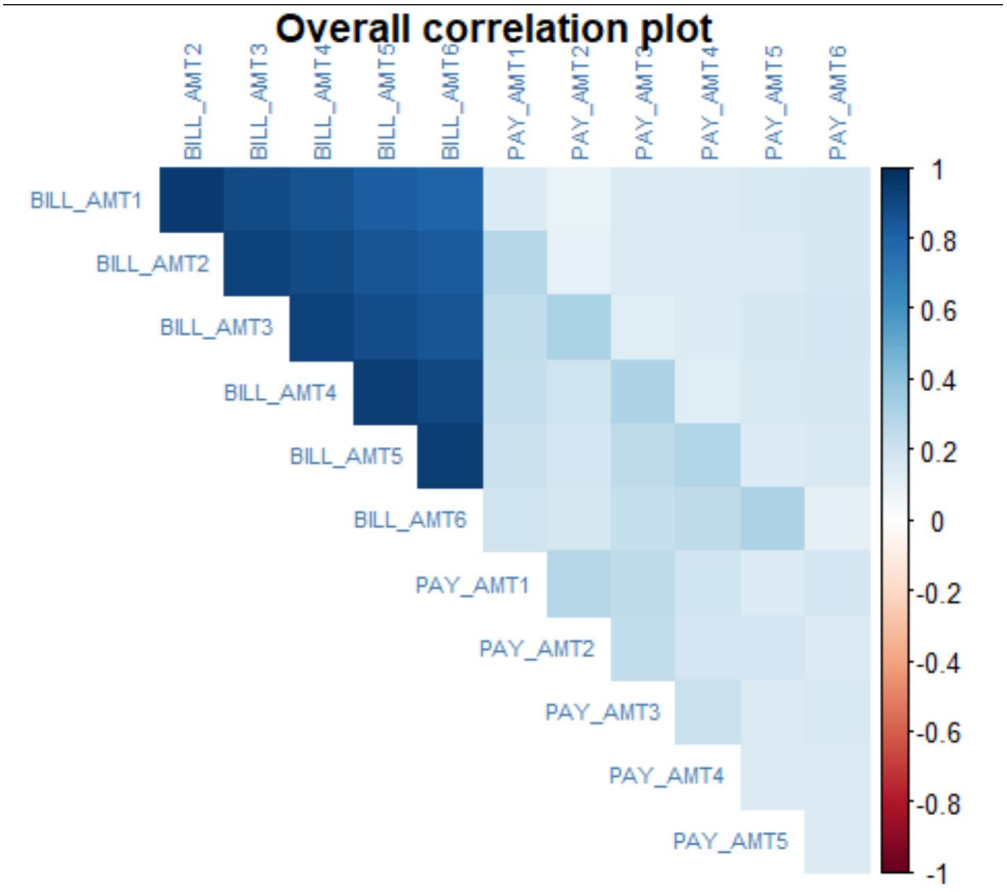
BILL_AMT1	BILL_AMT2	BILL_AMT3	BILL_AMT4	BILL_AMT5	BILL_AMT6
Min. :-165580	Min. :-69777	Min. :-157264	Min. :-170000	Min. :-81334	Min. :-339603
1st Qu.: 3559	1st Qu.: 2985	1st Qu.: 2666	1st Qu.: 2327	1st Qu.: 1763	1st Qu.: 1256
Median : 22382	Median : 21200	Median : 20088	Median : 19052	Median : 18104	Median : 17071
Mean : 51223	Mean : 49179	Mean : 47013	Mean : 43263	Mean : 40311	Mean : 38872
3rd Qu.: 67091	3rd Qu.: 64006	3rd Qu.: 60165	3rd Qu.: 54506	3rd Qu.: 50190	3rd Qu.: 49198
Max. : 964511	Max. :983931	Max. :1664089	Max. : 891586	Max. :927171	Max. : 961664
PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4	PAY_AMT5	PAY_AMT6
Min. : 0	Min. : 0	Min. : 0	Min. : 0	Min. : 0.0	Min. : 0.0
1st Qu.: 1000	1st Qu.: 833	1st Qu.: 390	1st Qu.: 296	1st Qu.: 252.5	1st Qu.: 117.8
Median : 2100	Median : 2009	Median : 1800	Median : 1500	Median : 1500.0	Median : 1500.0
Mean : 5664	Mean : 5921	Mean : 5226	Mean : 4826	Mean : 4799.4	Mean : 5215.5
3rd Qu.: 5006	3rd Qu.: 5000	3rd Qu.: 4505	3rd Qu.: 4013	3rd Qu.: 4031.5	3rd Qu.: 4000.0
Max. :873552	Max. :1684259	Max. :896040	Max. :621000	Max. :426529.0	Max. :528666.0

It can be seen that there are negative entries in the bill amount attributes.This may be interpreted as clients had prepaid their bills in previous months. Meanwhile, from April to September, the bill amount generally falls, as seen in the decreasing trend in its mean, median, 1st quartile and 3rd quartile. This implies that time of the year may also be a factor that affects bill amount for most clients. In fact, month can also be a factor affecting the payment amount: less payment is made in July to September, as compared to the previous months. This can be seen in the lower mean, median and 3rd quartile from July to September. One possible explanation can be that clients

choose to have summer vacations from July to September, hence more money is spent to finance their vacations instead of payment for bank.

Additionally, for bill amount, all these attributes have extremely high-valued outliers, which can explain the rise to the huge gaps between the 3rd quartile and the maximum visualized earlier(3.2(a)). In the later analysis we will seek to remove the skewness.

Correlation of bill amounts:



From the visualization, all bill amounts are closely correlated (dark blue) while the payment amounts are weakly connected(pale blue).Moderate correlations are found between each pair of BILL_AMT(n+1) and PAY_AMT(n) with n > 0. This shows that last month's payment amount directly affects this month's bill amount; namely the more paid last month, the less charged this month.

Using a one-way anova test with significance level of 0.05, we seek to find insights of the six bill_amounts and pay_amounts relationship with the target feature(default payment) and hope to reject the null hypothesis that the pair of attributes is independent.

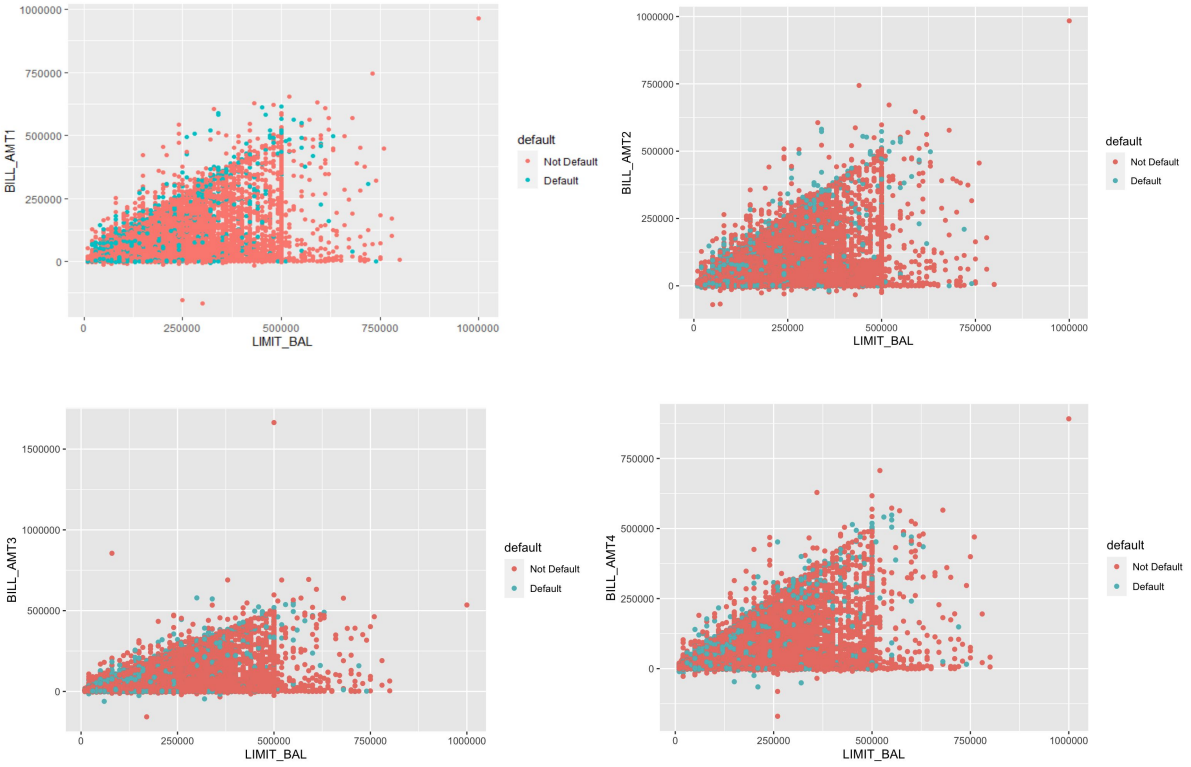
The results are:

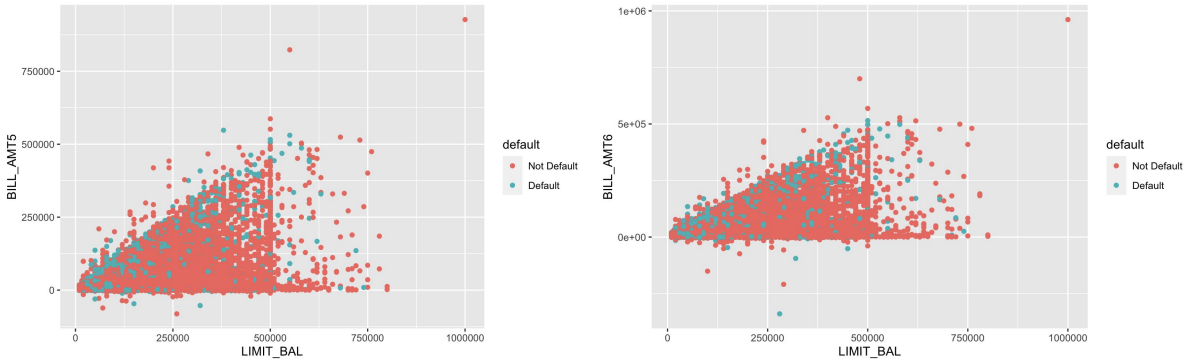
Pay_amounts: Bill_amounts:

```
[1] 1.146488e-36     [1] 0.0006673295
[1] 3.166657e-24     [1] 0.01395736
[1] 1.84177e-22      [1] 0.01476998
[1] 6.830942e-23     [1] 0.07855564
[1] 1.241345e-21     [1] 0.2416344
[1] 3.033589e-20     [1] 0.3521225
```

A common pattern found is that the further away a payment or bill was made from now, the less likely for it to influence much on the default value next month (have higher sig.fig levels). For payment amounts, all the p values are less than 0.05 and we conclude they have strong predictive power. For bill amounts, except the most recent 3 months, all others are not correlated to default payment next month. Summarised from above bill amounts didn't vary much over the 6 month period, thus might not contribute much to our prediction of default status.

(d) BILL_AMTs and balance limit



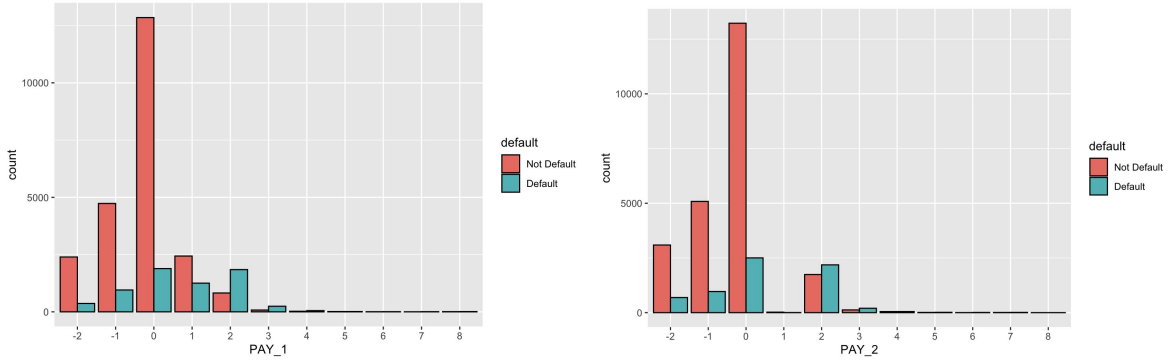


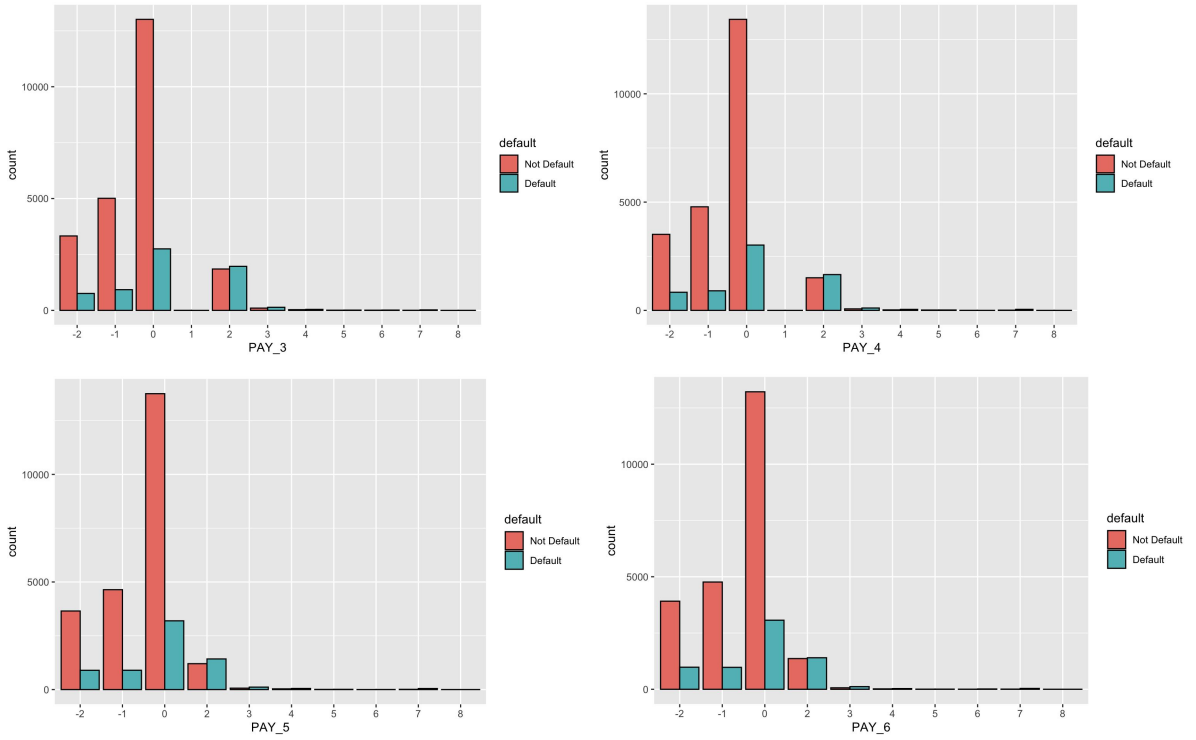
The triangular area suggests that bill amounts paid by clients are limited by their balance limit; most data points lie within the balance limit of 500000, which is an important threshold. Beyond this threshold, there are more non-default clients as compared to default clients, suggesting that high balance limit may be because of the clients' previous good record. Lastly, the default clients(blue) often lie on the upper side of the triangle, suggesting that default clients often meet their balance limit in their monthly bills.

3.4 The series of PAY_Ns

PAY_1		PAY_2		PAY_3		PAY_4		PAY_5		PAY_6	
0	:14737	0	:15730	0	:15764	0	:16455	0	:16947	0	:16286
-1	: 5686	-1	: 6050	-1	: 5938	-1	: 5687	-1	: 5539	-1	: 5740
1	: 3688	2	: 3927	-2	: 4085	-2	: 4348	-2	: 4546	-2	: 4895
-2	: 2759	-2	: 3782	2	: 3819	2	: 3159	2	: 2626	2	: 2766
2	: 2667	3	: 326	3	: 240	3	: 180	3	: 178	3	: 184
3	: 322	4	: 99	4	: 76	4	: 69	4	: 84	4	: 49
(Other):	141	(Other):	86	(Other):	78	(Other):	102	(Other):	80	(Other):	80

*PAY_1 starts from September 2005 to PAY_6, April 2005





There is a considerable amount of undefined entries ('0', '-2'), existing across the payment history attributes. To be consistent with the dataset description, we regard '-2' as clients paid ahead of time, '-1' as on time, '0' as delayed yet the clients had balance to maintain for a while. From the plot we can infer that clients who go default next month were more likely to have overdue payments for these months. Thus distinguishing whether clients paid on time for the past 6 months might be a good predictor of their default status next month.

3.5 Hypothesis Testing

Lastly, we conduct hypothesis testing to check for relationships between certain attributes to prevent multicollinearity and hope to merge some attributes together.

Hypothesis 1: Does Sex correlate to marital status?

The null hypothesis is that sex is independent of marital status. We adopt the Chi-Square method for this testing as both SEX and MARRIAGE are categorial.

```
        Pearson's Chi-squared test

data:  data$SEX and data$MARRIAGE
X-squared = 28.87, df = 2, p-value = 5.382e-07
```

Results show p-value of 5.382e-07. Since it is less than the threshold 0.05 (at 5% significance level), we have sufficient evidence to reject the hypotheses and conclude that *sex is correlated to marital status.*

Hypothesis 2: Does default status correlate to balance limit?

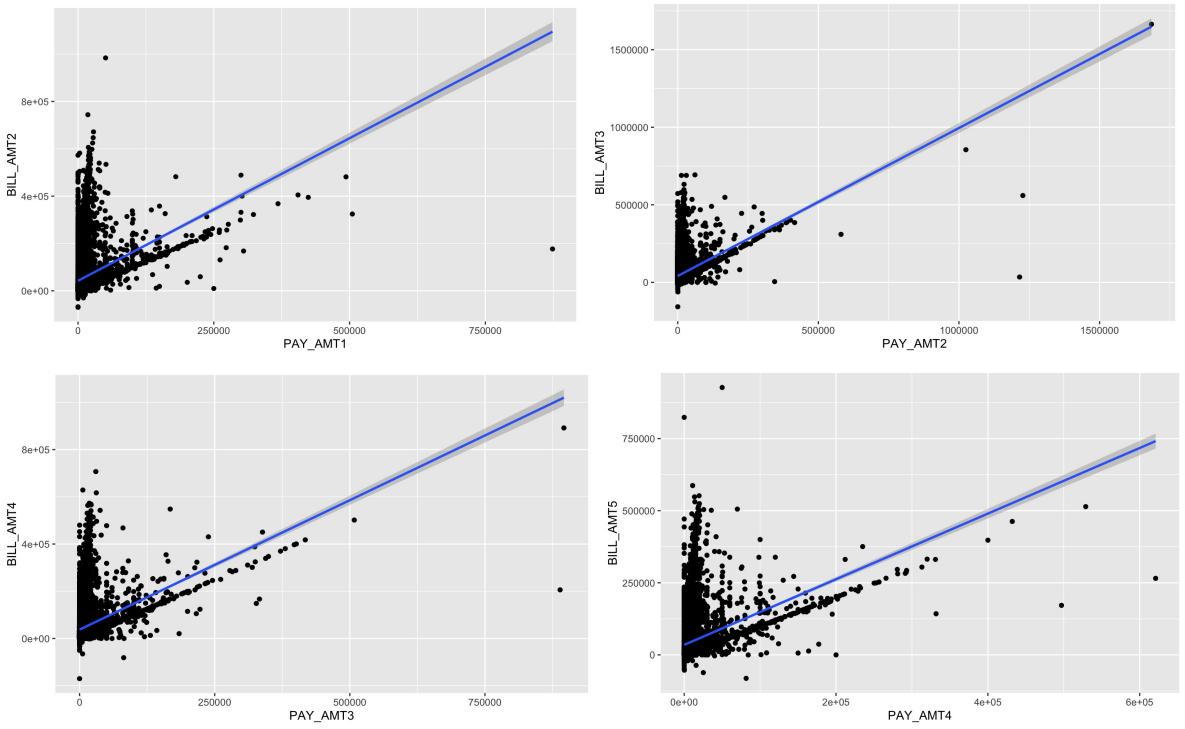
The null hypothesis states that default status is independent of balance limit. As default value is categorical and balance limit is continuous, we adopt the one-way anova test.

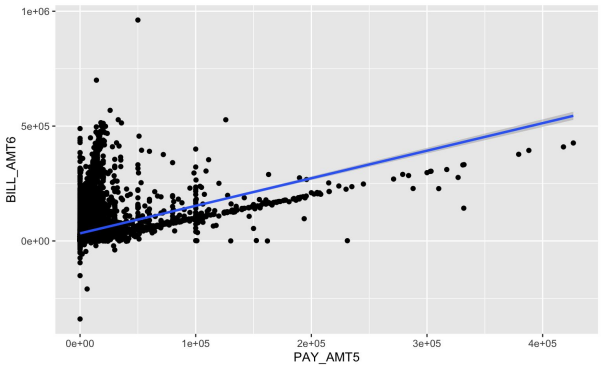
```
Analysis of Variance Table

Response: data$LIMIT_BAL
                Df     Sum Sq    Mean Sq F value    Pr(>F)
data$default     1 1.1902e+13 1.1902e+13  724.07 < 2.2e-16 ***
Residuals    29998 4.9311e+14 1.6438e+10
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The results show p-value of < 2.2e-16. Since it is far less than 0.05, we have sufficient evidence to reject the null hypothesis and conclude that *default status is correlated to balance limit.*

Hypothesis 3: Do BILL_AMT(n+1) correlate to PAY_AMT(n) for n =1,2,3,4,5?

The null hypothesis is that for all pairs of attributes, their correlation coefficient is 0. The scatter plot suggests that they are linearly correlated.





The resulting p-values for the above attribute's correlation tests were all extremely close to 0, which is smaller than 0.05. Thus we have sufficient evidence to reject the null hypothesis and conclude that *BILL_AMT(n+1) is correlated to PAY_AMT(n)* (for n =1,2,3,4,5)

In conclusion, the attributes are useful for prediction and did not happen by chance.

4. Data Transformation and feature selection

4.1 Include new terms

From the exploration on part 3.2 above, we decided to add in a new term SingleMale indicating whether a client is a single male. Secondly, add in new terms bill_limit (6 in all) which calculates the difference between the monthly bill and the balance limit, then normalized by the balance limit.Thirdly, add in new term expenseN to derive more information on client expenditure, which is computed by (last month's bill - (this month's bill - last months payment)) for monthN. Lastly, add in a new term goodClient which is binary-valued and indicates whether a client has balance limit > 500K.

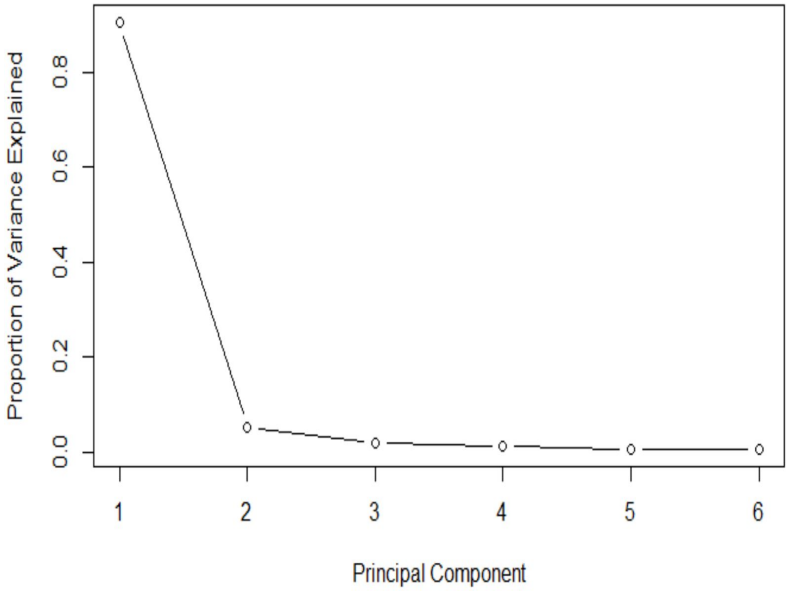
4.2 Attribute transformation

4.2.1 PAY_Ns (N = 1,2,3,4,5,6)

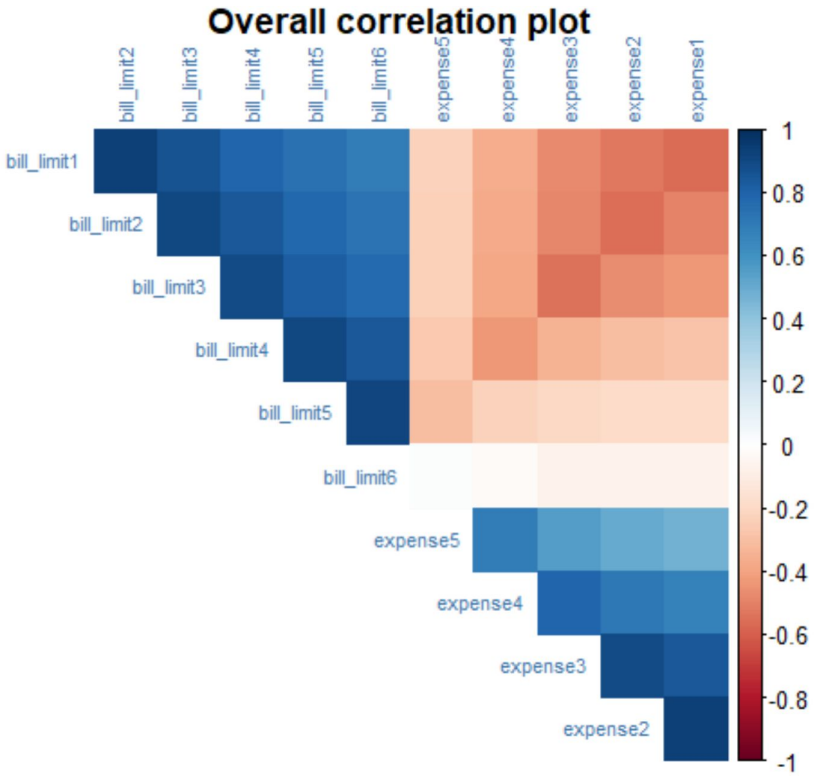
Since all entries with PAY_N less than or equal to 0 are payments on time while the rest are overdue payments, we classify and assign new binary attributes pay(n) to indicate whether the clients are good payers for each month. (PAY_N > 0 or not)

4.2.2 Use PCA to reduce collinearity for BILL_AMTs

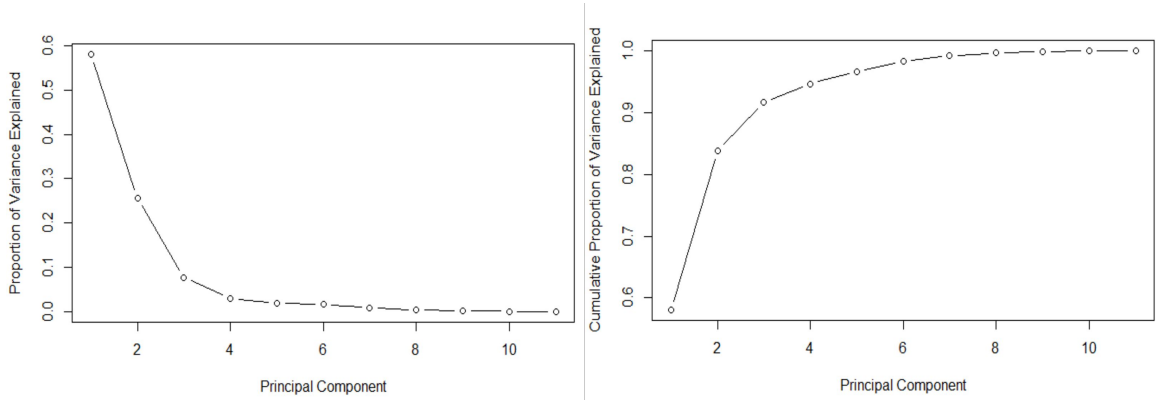
As the series of bill amounts have been illustrated to be closely correlated in section 3.3, we substitute for their principal components to reduce the overlapped dimensions. Results show that the first two PCs explain over 98% of the total variance. Thus we substitute the bill amounts for these 2 PCs to simplify our model.



4.2.3 Use PCA to reduce collinearity for Bill_limits and Expenses



As plotted in the overall correlation plot, we have two sets of strongly correlated features. Now we try to find their principle components to further reduce overlapped dimensions.



It turns out that choosing 6 out of the 11 components can already explain over 98% of the original variance. Thus we remove the 11 features and include the new 6 PCs named **Bill_exp(n)**'s.

4.3. Split training and test set

We have split the data to train and test sets for training the model and to check accuracy of model later on. As mentioned in *section 3.1*, the dataset is quite imbalanced (0.28) against the default clients next month. To overcome the imbalance, we chose to use the stratified sampling approach, i.e. select equal proportion of default clients in each set. The method we applied was the stratified k-fold cross validation method to overcome the imbalance issue and evaluate the models more accurately.

4.4. Feature Selection

To shortlist useful features and reduce the data complexity, we adopted the logistic regression model and performed a stepwise feature selection process. The selection results are as follows: LIMIT_BAL + EDUCATION + MARRIAGE + AGE + PAY_AMT1 + PAY_AMT2 + SingleMale + pay1 + pay2 + pay3 + pay4 + pay5 + pay6 + bill_amt1 + bill_amt2 + bill_exp1 + bill_exp2 + bill_exp3 + bill_exp4

5. Model Selection

5.0 Null model

If all clients were to be predicted not default, the accuracy would be 1 - 0.2840 = 0.7160. Thus our models' performance should be no worse than this. This means

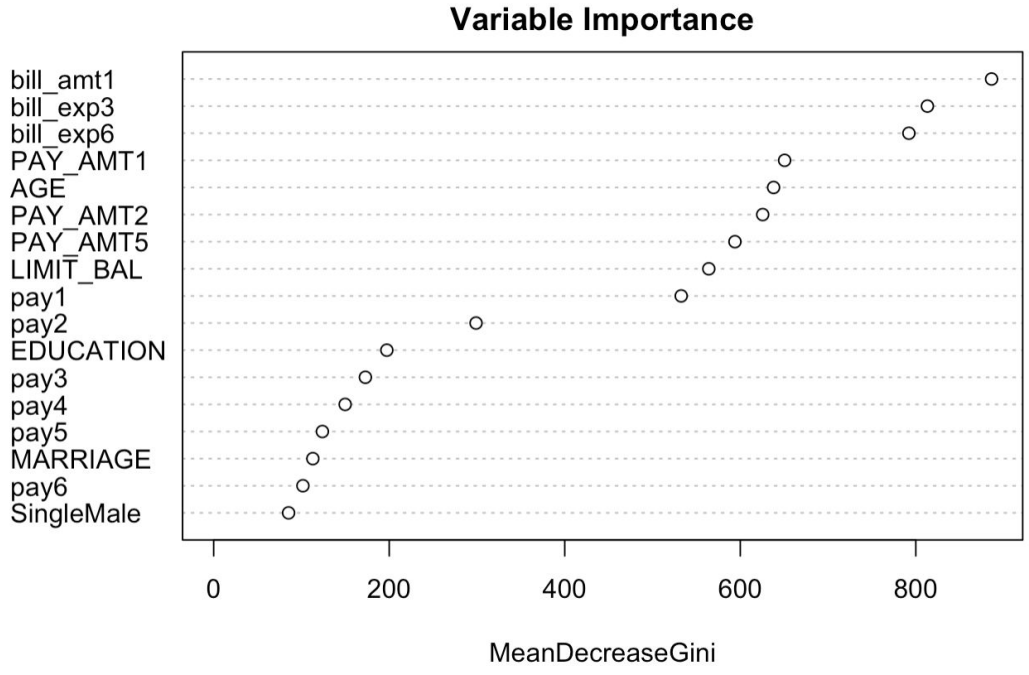
that given 100 data samples, the model can predict 71 of them correctly. The models chosen in *Section* 5 should be able to have more correct predictions than 71.

5.1. Random Forest

Random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees.

It is usually more accurate than decision trees as it is built on random samples. This solves the problem of decision tree being unstable, since changes to the sample will not affect the model greatly. Random forest can handle large data assets with higher dimensionality, which is useful for reducing the dimensionality of our model. It also handles imbalance data well, as random forest has an inbuilt method of balancing errors in data sets. However, random forest does not give the best continuous nature prediction, and does not predict beyond the range of training data, which results in possible limitation of over-fitting.

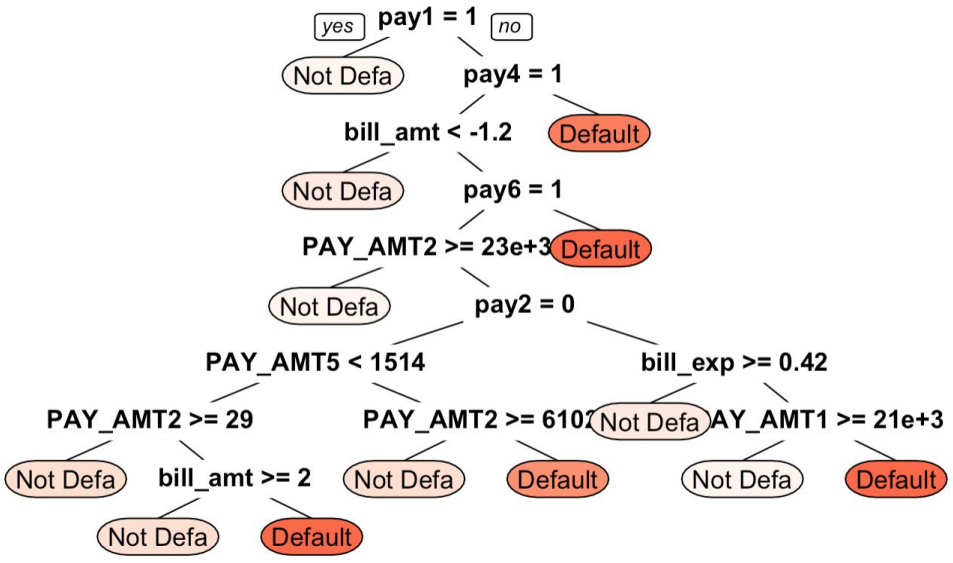
From the graph below, bill amounts and bill expenses were the most important in deciding the target feature. This model has an accuracy of 0.818, which is above the threshold. Hence, we consider this to be a good model.



5.2. Decision Tree

A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). Although the decision tree is safe and easy to understand from visualization, it is unable to detect linear relationships between the attribute and target feature.It is also unstable as a small change in the dataset will cause the model to change greatly.

In our case, we have 18 predictors to predict if a customer belongs in class 'Not Default' or 'Default' using repeated cross validation and reducing GINI as the criteria in splitting the tree. In this model, the payment amounts to the bank and bill amounts were used to decide if a client will be ‘default’ or not. We can see how if we add some new data from clients such that the first deciding attribute is not pay1, the whole tree will be affected, thus it is unstable. The accuracy resulted to **0.812**, since it is above the threshold, we consider it a good model.



5.3. SVM

Support-vector machines are supervised models with associated learning algorithms that analyze data used for classification and regression analysis. In general, SVM works well when some variables are correlated.

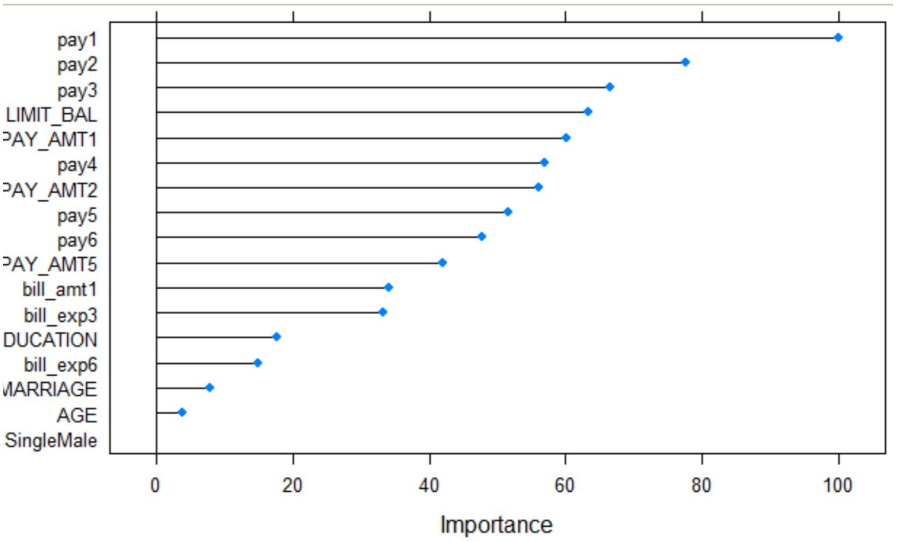
In this case, LIMIT_BAL and default status, SEX and MARRIAGE and AGE and EDUCATION as seen in the anova test. Unlike Naive Bayes classification which requires the variables to be independent of each other, SVM hence is a better choice. Additionally, SVM might be more robust even when the training sample has some bias. In this case, distribution of variables such as LIMIT_BAL, AGE and default is pretty skewed to the left, requiring a model that can perform well on a sample with some bias and weaknesses.

However, it has bad interpretability: unlike logistic and linear regressions, for which we can interpret the meaning of the intercepts, it is hard to interpret SVM. Furthermore, high computational cost, SVMs scale exponentially in training time.Therefore, because of the large number of data we have in this case, the training time can be longer than other models. Our SVM model resulted in **0.817** accuracy level. Since it is above the threshold, we consider it a good model.

5.4. Naive Bayes

Naive Bayes classifier is a probabilistic classification model based on Bayes' theorem. It relies on strong independence assumptions among the features used.

The plot below shows how each predictor variable is independently responsible for predicting the outcome. It suggests that pay1 to 3 are the top three depending factors.



Using a Naive Bayes classifier, we predict the default payment next month of a card holder. The output above shows that this Naive Bayes classifier has an accuracy of approximately **0.798**. It is above the threshold, so we consider it a good model.

In conclusion, we observe that the amount of bill statements and payment history, particularly the closer months, are more important attributes to predict default payment than others such as marriage or age. This makes sense since recent payment history and bill amounts may reveal the current financial liquidity of the client.

6. Evaluation of model selection

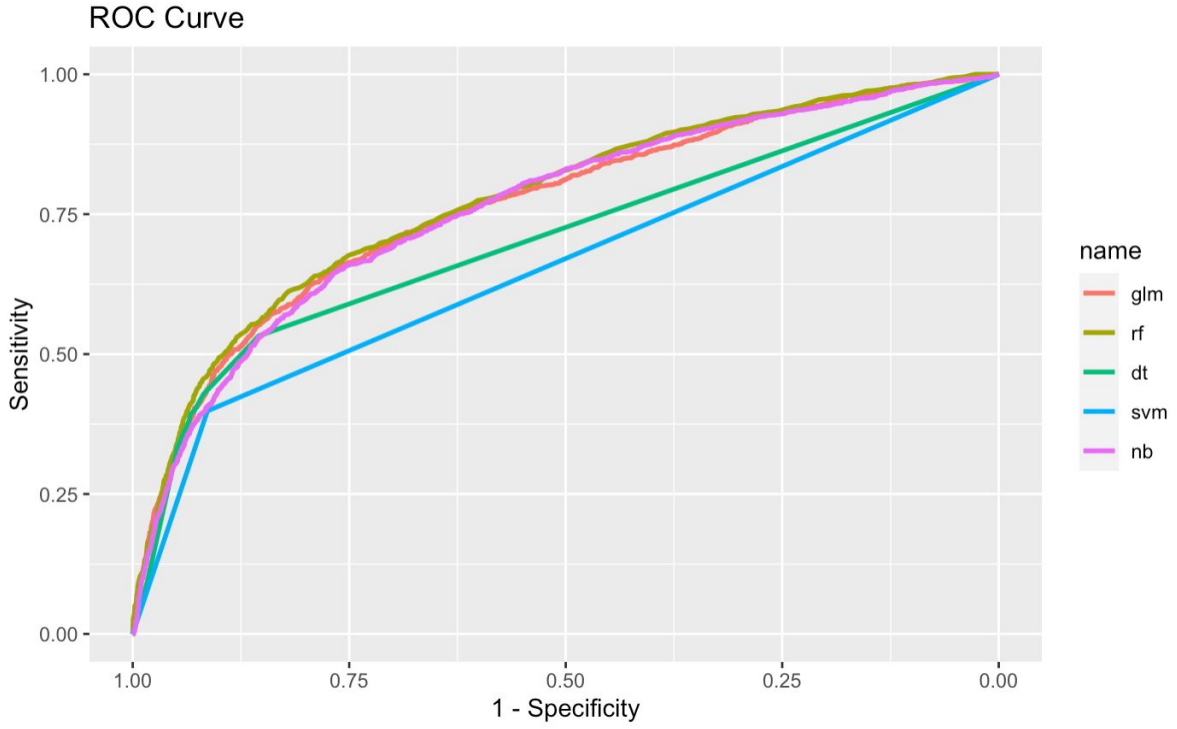
In general, we aim to have low FNR(False Negative Rate) and high accuracy. Hence, from the table below, the random forest model seems to be the best

performing model among all the models we have chosen. It has the highest accuracy of 0.8181, and also a moderately low FNR of 0.0594.

The FPR(False Positive Rate), the ratio of FP against the sum of FP and TNs, is relatively high for all models. As the sample we are using are biased with more positive than negatives, this ratio will inherently be high. Hence, FPR is not as important as FNR when we are choosing the best model. Furthermore, from the bank's point of view, we would want to have a low FNR since the bank would not want to wrongly predict a client will default next month when he or she will be repaying on time. This may result in an incorrect decision when deciding if the client can make loans or not, which will affect the bank's earnings as well.

Evaluation Table

	Total Samples	Accuracy	FPR	FNR	Recall	Precision	F1 Score
glm	7500	0.8119	0.6854	0.0469	0.9531	0.8304	0.8875
Random Forest	7500	0.8181	0.6130	0.0594	0.9406	0.8438	0.8896
Decision Tree	7500	0.8124	0.6166	0.0657	0.9343	0.8421	0.8858
SVM radial	7500	0.8167	0.6323	0.0558	0.9442	0.8402	0.8892
Naive Bayes	7500	0.7979	0.8354	0.0223	0.9777	0.8047	0.8828




```
Area under ROC:
Generlised Linear Model 0.7526968
Random Forest:  0.7660556
Decision Tree:  0.7014506
Support Vector Machine:  0.656166
Naive Bayes:  0.7501506
```

With a random classifier, the AUC would be 0.5. Thus, all the models will improve the prediction of target features. Based on the ROC Curves plotted from the predictive models, we can see that GLM, RF, NB perform much better than SVM and DT. (i.e. have a larger area under curve). We observe that the trade-off between sensitivity and specificity for RF is the least, having a AUC of 0.766. This is as expected as we know RF in general has better accuracy than decision trees given that it won't create a highly biased model and reduces the variance. Hence, we select the Random Forest Model as our final decision.

7. Improvements

To improve the dataset, the bank may replace some attributes that may be more important towards predicting if clients will be ‘default’ payment next month or not. It is shown that some features such as gender, or bill amount in April (6 months ago) are less important. We may consider other attributes such as monthly income or job status that may be more useful since it shows the client’s financial liquidity.

We can further improve our models by monitoring changes in performance measures over time. After a period of deployment, we can collect all the new instances presented to the model and calculate all the statistics and ROC curve again. If there is a huge change in ROC index or accuracy scores, it would flag that our model may need to be modified.

Besides monitoring the model statistics, we can also monitor the future changes in distribution of model outputs. Stability index can be used in this case, where a stability index of greater than 0.1 indicates that the model should be closely monitored, and a stabilit index of greater than 0.25 indicates that corrective action may be required.

R Notebook

Import packages

```
# Data manipulation packages:
library(dplyr)
library(plyr)
library(psych)
library(splitTools)
library(tidyverse)

# Plot packages:
library(ggplot2)
library(gridExtra)
library(corrplot)
library(pROC)
library(rpart.plot)

# Model packages:
library(randomForest)
library(rpart)
library(klaR)
library(class)
library(algaeClassify)
library(dismo)

# Evaluation packages:
library(imbalance)
library(InformationValue)
```

```
data.raw <- read.table("card.csv",sep=",",skip=2,header=FALSE)
header <- scan("card.csv",sep=",",nlines=2,what=character())
names(data.raw) <- header[26:50]
```

```
# First glance of the data
dim(data.raw)

head(data.raw)

describe(data.raw)
```

Data Preparation

Remove redundant values and check for missing data

1. Remove the ID attribute

```
data <- data.raw[,2:25]
```

2. Rename the attributes

```
colnames(data)[24] <- 'default'
```

```
colnames(data)[6] <- 'PAY_1'
```

Check missing data

```
sum(sapply(data, is.na))
```

Identify undefined values

1. MARRIAGE

```
for (i in 1:nrow(data)) {
  if (data[i,4] == 1) {
    data[i,4] <- "Single"
  } else if (data[i,4] == 2) {
    data[i,4] <- "Married"
  } else {
    data[i,4] <- "other"
  }
}
```

2. EDUCATION

```
for (i in 1:nrow(data)) {
  if (data[i,3] == 1) {
    data[i,3] <- "Graduate"
  } else if (data[i,3] == 2) {
    data[i,3] <- "University"
  } else if (data[i,3] == 3) {
    data[i,3] <- "High School"
  } else {
    data[i,3] <- "Unknown"
  }
}
```

Factorize categorical variables

```
data$default <- factor(data$default)
levels(data$default) <- c("Not Default", "Default")
data$SEX <- factor(data$SEX)
levels(data$SEX) <- c("Male", "Female")
data$MARRIAGE <- factor(data$MARRIAGE)
```

```r
data$MARRIAGE <- relevel(data$MARRIAGE,"other") # Reorder the levels
data$EDUCATION <- factor(data$EDUCATION)
data$EDUCATION <- relevel(data$EDUCATION,"Unknown") # Reorder the levels
data$PAY_1 <- factor(data$PAY_1)
data$PAY_2 <- factor(data$PAY_2)
data$PAY_3 <- factor(data$PAY_3)
data$PAY_4 <- factor(data$PAY_4)
data$PAY_5 <- factor(data$PAY_5)
data$PAY_6 <- factor(data$PAY_6)
```

Exploratory Data Analysis

1. Target Attribute

```r
summary(data$default)
imbalanceRatio(data,classAttr='default')
```

2. Demographic variables

```r
# Freq counts:
cat("Sex:\n")
summary(data$SEX)
cat("Education:\n")
summary(data$EDUCATION)
cat("Marriage:\n")
summary(data$MARRIAGE)
```

```r
ggplot(data, aes(x=SEX, fill=default)) +
  geom_histogram(stat="count", show.legend = F)
```

```r
ggplot(data, aes(x=EDUCATION, fill=default)) +
  geom_histogram(stat="count", show.legend = F)
```

```r
ggplot(data, aes(x=MARRIAGE, fill=default)) +
  geom_histogram(stat="count")
```

(a) Education, gender and balance limit

```r
ggplot(data, aes(SEX, (LIMIT_BAL), fill=EDUCATION)) +
  geom_boxplot() +
  xlab("Gender") +
  ylab("Bal_limit") +
  scale_fill_brewer(palette = "Paired")
```

(b) Sex, marriage and balance limit

```r
ggplot(data, aes(MARRIAGE, LIMIT_BAL, fill=SEX)) +
  geom_boxplot() +
  xlab("Marital Status") +
  ylab("Balance Limit")  +
  scale_fill_brewer(palette = "Pair")
```

3. Continuous Variables and their interactions

(a) Balance limit:

```
ggplot(data, mapping = aes(x = LIMIT_BAL)) +
  geom_histogram(aes(y=..density..), binwidth=5000, color="black", fill="white") +
  geom_density(fill="red",alpha=0.3)
```

(b) Age

```
ggplot(data, aes(x=AGE , fill=default)) +
  geom_bar(colour="black", position="dodge")
```

(c) PAY_AMTs and BILL_AMTs

```
# Summary statistics for PAY_AMTs and BILL_AMTs.
summary(data[,12:23])
```

```
# Correlation plot (upper triangular)
M <- cor(data.raw[,c(13:24)])
corrplot(M,method="color",
         diag=F,
         type="upper",
         title='Overall correlation plot',
         tl.col='#4477AA',
         tl.cex=0.7,
         mar=c(0,0,1,0))
```

```
# One-way anova test
anova(lm(data$PAY_AMT1~data$default))$`Pr(>F)`[1]
anova(lm(data$PAY_AMT2~data$default))$`Pr(>F)`[1]
anova(lm(data$PAY_AMT3~data$default))$`Pr(>F)`[1]
anova(lm(data$PAY_AMT4~data$default))$`Pr(>F)`[1]
anova(lm(data$PAY_AMT5~data$default))$`Pr(>F)`[1]
anova(lm(data$PAY_AMT6~data$default))$`Pr(>F)`[1]
```

```
anova(lm(data$BILL_AMT1~data$default))$`Pr(>F)`[1]
anova(lm(data$BILL_AMT2~data$default))$`Pr(>F)`[1]
anova(lm(data$BILL_AMT3~data$default))$`Pr(>F)`[1]
anova(lm(data$BILL_AMT4~data$default))$`Pr(>F)`[1]
anova(lm(data$BILL_AMT5~data$default))$`Pr(>F)`[1]
anova(lm(data$BILL_AMT6~data$default))$`Pr(>F)`[1]
```

(d) BILL_AMTs and balance limit

```
# Scatter plots
par(mfrow=c(3,2))
ggplot(data, aes(x=LIMIT_BAL, y=BILL_AMT1, color=default)) +
  geom_point()
```

```
ggplot(data, aes(x=LIMIT_BAL, y=BILL_AMT2, color=default)) +
  geom_point()
```

```r
ggplot(data, aes(x=LIMIT_BAL, y=BILL_AMT3, color=default)) +
  geom_point()
```

```r
ggplot(data, aes(x=LIMIT_BAL, y=BILL_AMT4, color=default)) +
  geom_point()
```

```r
ggplot(data, aes(x=LIMIT_BAL, y=BILL_AMT5, color=default)) +
  geom_point()
```

```r
ggplot(data, aes(x=LIMIT_BAL, y=BILL_AMT6, color=default)) +
  geom_point()
```

4. The series of PAY__Ns

```r
# Frequencies of PAY_N's
summary(data[,6:11])
```

```r
# Dodged bar plots
par(mfrow=c(3,2))
ggplot(data, aes(x=PAY_1 , fill=default)) +
  geom_bar(colour="black", position="dodge")
```

```r
ggplot(data, aes(x=PAY_2 , fill=default)) +
  geom_bar(colour="black", position="dodge")
```

```r
ggplot(data, aes(x=PAY_3 , fill=default)) +
  geom_bar(colour="black", position="dodge")
```

```r
ggplot(data, aes(x=PAY_4 , fill=default)) +
  geom_bar(colour="black", position="dodge")
```

```r
ggplot(data, aes(x=PAY_5 , fill=default)) +
  geom_bar(colour="black", position="dodge")
```

```r
ggplot(data, aes(x=PAY_6 , fill=default)) +
  geom_bar(colour="black", position="dodge")
```

```r
# Correlation plot
M <- cor(data.raw[,c(2,7:12)])
corrplot(M,method="color",
         diag=F,
         type="upper",
         title='Overall correlation plot',
         tl.col='#4477AA',
         tl.cex=0.7,
         mar=c(0,0,1,0))
```

5. Hypothesis Testing

Hypothesis 1: Does Sex correlate to marital status?

```r
# Chi-square test
chisq.test(data$SEX,data$MARRIAGE)
```

Hypothesis 2: Do clients with different default status next month have different mean balance limit?

```
# One-way anova test
anova(lm(data$LIMIT_BAL~data$default))
```

Hypothesis 3: do BILL_AMT(n+1) correlate to PAY_AMT(n) for n =1,2,3,4,5?

```
# Scatter plot with regression line
par(mfrow=c(3,2))
ggplot(data, aes(x=PAY_AMT1, y=BILL_AMT2)) +
  geom_point()+
  geom_smooth(method=lm)
```

```
ggplot(data, aes(x=PAY_AMT2, y=BILL_AMT3)) +
  geom_point()+
  geom_smooth(method=lm)
```

```
ggplot(data, aes(x=PAY_AMT3, y=BILL_AMT4)) +
  geom_point()+
  geom_smooth(method=lm)
```

```
ggplot(data, aes(x=PAY_AMT4, y=BILL_AMT5)) +
  geom_point()+
  geom_smooth(method=lm)
```

```
ggplot(data, aes(x=PAY_AMT5, y=BILL_AMT6)) +
  geom_point()+
  geom_smooth(method=lm)
```

```
# Correlation test (only p-value retrieved)
cor.test(data$PAY_AMT1,data$BILL_AMT2)$p.value
cor.test(data$PAY_AMT2,data$BILL_AMT3)$p.value
cor.test(data$PAY_AMT3,data$BILL_AMT4)$p.value
cor.test(data$PAY_AMT4,data$BILL_AMT5)$p.value
cor.test(data$PAY_AMT5,data$BILL_AMT6)$p.value
```

Data Transformation and feature selection

1. Include new terms:

```
# Add in new terms
data$SingleMale <- factor(dummy.code(data$MARRIAGE)[,2] * dummy.code(data$SEX)[,1])
data$bill_limit1 <- (data$LIMIT_BAL - data$BILL_AMT1) / data$LIMIT_BAL
data$bill_limit2 <- (data$LIMIT_BAL - data$BILL_AMT2) / data$LIMIT_BAL
data$bill_limit3 <- (data$LIMIT_BAL - data$BILL_AMT3) / data$LIMIT_BAL
data$bill_limit4 <- (data$LIMIT_BAL - data$BILL_AMT4) / data$LIMIT_BAL
data$bill_limit5 <- (data$LIMIT_BAL - data$BILL_AMT5) / data$LIMIT_BAL
data$bill_limit6 <- (data$LIMIT_BAL - data$BILL_AMT6) / data$LIMIT_BAL
data$goodClient <- factor(data$LIMIT_BAL > 500000)
data$expense5 <- (data$BILL_AMT5 - (data$BILL_AMT6 - data$PAY_AMT5)) / data$LIMIT_BAL
data$expense4 <- (((data$BILL_AMT5 - (data$BILL_AMT6 - data$PAY_AMT5)) +
                     (data$BILL_AMT4 - (data$BILL_AMT5 - data$PAY_AMT4))) / 2) / data$LIMIT_BAL
data$expense3 <- (((data$BILL_AMT5 - (data$BILL_AMT6 - data$PAY_AMT5)) +
```

```
                    (data$BILL_AMT4 - (data$BILL_AMT5 - data$PAY_AMT4)) +
                    (data$BILL_AMT3 - (data$BILL_AMT4 - data$PAY_AMT3))) / 3) / data$LIMIT_BAL
data$expense2 <- (((data$BILL_AMT5 - (data$BILL_AMT6 - data$PAY_AMT5)) +
                    (data$BILL_AMT4 - (data$BILL_AMT5 - data$PAY_AMT4)) +
                    (data$BILL_AMT3 - (data$BILL_AMT4 - data$PAY_AMT3)) +
                    (data$BILL_AMT2 - (data$BILL_AMT3 - data$PAY_AMT2))) / 4) / data$LIMIT_BAL
data$expense1 <- (((data$BILL_AMT5 - (data$BILL_AMT6 - data$PAY_AMT5)) +
                    (data$BILL_AMT4 - (data$BILL_AMT5 - data$PAY_AMT4)) +
                    (data$BILL_AMT3 - (data$BILL_AMT4 - data$PAY_AMT3)) +
                    (data$BILL_AMT2 - (data$BILL_AMT3 - data$PAY_AMT2)) +
                    (data$BILL_AMT1 - (data$BILL_AMT2 - data$PAY_AMT1))) / 5) / data$LIMIT_BAL
```

```
# Correlation plot
M <- cor(data[,c(26:31,33:37)])
corrplot(M,method="color",diag=F,type="upper",title='Overall correlation plot',tl.col='#4477AA',tl.cex=0
```

2. Attribute transformation:

2.1 PAY_Ns

```
# Classification of payn's
data$pay1 <- ifelse((data.raw$PAY_0) > 0, 0, 1)
data$pay2 <- ifelse((data.raw$PAY_2) > 0, 0, 1)
data$pay3 <- ifelse((data.raw$PAY_3) > 0, 0, 1)
data$pay4 <- ifelse((data.raw$PAY_4) > 0, 0, 1)
data$pay5 <- ifelse((data.raw$PAY_5) > 0, 0, 1)
data$pay6 <- ifelse((data.raw$PAY_6) > 0, 0, 1)
data_clean <- data[,!(names(data) %in% c("PAY_1","PAY_2","PAY_3","PAY_4","PAY_5","PAY_6",
                                         "BILL_AMT1","BILL_AMT2","BILL_AMT3","BILL_AMT4","BILL_AMT5","BI
```

2.2 BILL_AMTs

```
# PCA plot - variance explained
pca <- prcomp(data[,c(26:31,33:37)], scale = T)
exp_var = (pca$sdev)^2 / sum(( pca$sdev )^2)
plot(exp_var, xlab = "Principal Components", ylab = "Explained Variance", type = "b" )
```

```
# Sub in the PCs
data_clean$bill_amt1 <- pca$x[,1]
data_clean$bill_amt2 <- pca$x[,2]
```

2.3 Bill_limits and expenses

```
# PCA plot - variance explained
pca1 <- prcomp(data[,c(26:31,33:37)], scale = T)
exp_var1 = (pca1$sdev)^2 / sum(( pca1$sdev )^2)
plot(exp_var1, xlab = "Principal Components", ylab = "Explained Variance", type = "b" )
```

```
cum <- cumsum(exp_var1)
plot(cum, xlab = "Principal Components",ylab = "Cumulated Explained Variance",type = "b")
```

```
# Sub in the PCs
data_final <- data_clean[,!(names(data_clean) %in% c("expense1","expense2","expense3","expense4","expens
                                                     "bill_limit1","bill_limit2","bill_limit3","bill_li
```

```r
data_final[,23:28] <- pca1$x[,1:6]
colnames(data_final)[23:28] <- c("bill_exp1","bill_exp2","bill_exp3","bill_exp4","bill_exp5","bill_exp6
```

3. Split train, test set:

```r
# Stratified sampling
library(splitTools)
inds <- partition(data_final$default, p=c(train = 0.75, test = 0.25))
train.data <- data_final[inds$train,]
test.data <- data_final[inds$test,]
train.class <- train.data$default
test.class <- test.data$default
```

```r
# Imbalance ratio
count(train.data$default)
count(test.data$default)
c(imbalanceRatio(train.data,classAttr = "default"),imbalanceRatio(test.data,classAttr = "default"))
```

4. Feature Selection

```r
# Auxiliary function
get_stats <- function(model_results, test_data, model_name){
  cm <- table(actual = test_data,pred = model_results)
  print(cm)
  n <- cm[1] + cm[2] + cm[3] + cm[4]
  accuracy <- sum(diag(cm))/n
  fpr <- cm[2]/(cm[2] + cm[4])
  fnr <- cm[3]/(cm[3] + cm[1])
  recall <- cm[1]/(cm[3]+cm[1])
  precision <- cm[1]/(cm[1]+cm[2])
  f1 <- (2*precision*recall)/(precision + recall)
  cat("Precision: ", precision,"\n")
  cat("Recall: ", recall,"\n")
  cat("Accuracy: ", accuracy,"\n")
  cat("FPR: ", fpr,"\n")
  cat("FNR: ", fnr,"\n")
  cat("F1 score: ",f1,"\n")
  final <- matrix(c(n,accuracy,fpr,fnr,recall,precision,f1),nrow=1,byrow=TRUE)
  colnames(final) <- c("Total Samples","Accuracy","FPR","FNR","Recall","Precision","F1 Score")
  rownames(final) <- model_name
  final <- as.table(final)
  return(final)
}
```

```r
# Full model construction:
full <- glm(default~.,data=train.data,family="binomial")

# Step-wise feature selection (backward):
step <- full %>% stepAIC(trace = FALSE)

# Selection results:
step$anova
```

```
# Predict on the testset.
pred <- predict(step, newdata=test.data, type="response")
predbin <- factor(ifelse(pred<0.5,"Not Default","Default"))
predbin <- relevel(predbin,"Not Default") # Rotate the order of levels to make confusion matrix

# Summary statistics
glm <- get_stats(predbin,test.class,"glm")
```

```
# Retained selected features.
train.log <- train.data[,!(names(train.data) %in%
                             c("SEX","PAY_AMT3","PAY_AMT4","PAY_AMT6","bill_exp1",
                               "bill_exp2","bill_exp4","bill_exp5","bill_amt2","SEX","goodClient"))]
test.log <- test.data[,!(names(test.data) %in%
                           c("SEX","PAY_AMT3","PAY_AMT4","PAY_AMT6","bill_exp1",
                             "bill_exp2","bill_exp4","bill_exp5","bill_amt2","SEX","goodClient"))]

# Extract target feature.
train.clog <- train.log$default
test.clog <- test.log$default
```

Model Selection

1. Random Forest

```
# Select best mtry.
set.seed(123)
rf.model <- tuneRF(train.log[,!(names(train.log) %in% c("default"))],
                   train.log$default,
                   ntreeTry=600,
                   stepFactor=1.5,
                   doBest=TRUE)
```

```
# Plot the variable importance
varImpPlot(rf.model,main="Variable Importance")
```

```
# Get the statistics
test.pred.rf <- predict(rf.model, test.log)
rf <- get_stats(test.pred.rf, test.clog,"Random Forest")
```

2. Decision Tree

```
library(caret)

# Repeated K-fold cross validation.
train <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
set.seed(123)
tree.model <- train(default ~.,
                    data = train.log,
                    method = "rpart",
                    parms = list(split = "gini"),
                    trControl = train,
                    tuneLength = 10)
```

```
# Plot the decision rule
prp(tree.model$finalModel, box.palette = "Red", tweak = 1.2)
```

```
# Get the statistics
predicted.response <- predict(tree.model, test.log)
dt <- get_stats(predicted.response, test.log$default,"Decision Tree")
```

3. SVM

```
library(e1071)
set.seed(123)
```

```
# "C-Classification"
svm_radial <- svm(default ~., data = train.log,type="C-classification",
                  kernel="radial")
summary(svm_radial)

# use the model to predict for train set and test set
results_test <- predict(svm_radial, test.log )
svm <- get_stats(results_test, test.log$default,"SVM radial")
```

4. Naive Bayes

```
set.seed(123)
x = train.log[,-8]
y = train.log$default

# K-fold cross validation exercised.
nb.model = train(x,y,'nb',trControl=trainControl(method='cv',number=10))
nb.model

# Plot the variance importance
X <- varImp(nb.model)
plot(X)
```

```
# Get the evaluation metrics
nb.pred <- predict(nb.model,newdata = test.log)
nb <- get_stats(nb.pred, test.clog,"Naive Bayes")
```

#Evaluation of model selection

```
# Construct the evaluation table
table <- rbind(glm, rf, dt, svm, nb)
table %>% knitr::kable(caption = "Evaluation Table", digit = 3)
```

```
# Create the roc objects.
roc.glm <- roc(test.class,
               predict(step, newdata=test.data,type="response"), quiet = T)
roc.rf <- roc(test.clog,
              predict(rf.model, newdata=test.log[,-8],type="prob")[,1],
              quiet = T)
roc.dt <- roc(test.clog,
              predict(tree.model, newdata=test.log[,- 8],type="prob")[,2],
              quiet = T)
```

```
roc.svm <- roc(test.clog,
               ifelse(predict(svm_radial, newdata=test.log[,- 8],
                              type="prob") == "Default", 1, 0),
               quiet = T)
roc.nb <- roc(test.clog,
              predict(nb.model, newdata=test.log[,- 8],type="prob")[,1],
              quiet = T)
```

```
roclist <- list("glm" = roc.glm, "rf" = roc.rf, "dt" = roc.dt,"svm" = roc.svm, "nb" = roc.nb)

# Plot the multi-line ROC curve.
ggroc(roclist,linetype = 1, size = 1, show.legend=T) +
  ggtitle("ROC Curve") +
  labs(x = "1 - Specificity",
       y = "Sensitivity",
       linetype = "Models")
```

```
# Print the area under the ROC curve.
cat("Area under ROC: \n")
cat("Generlised Linear Model",roc.glm$auc,"\n")
cat("Random Forest: ",roc.rf$auc,"\n")
cat("Decision Tree: ",roc.dt$auc,"\n")
cat("Support Vector Machine: ",roc.svm$auc,"\n")
cat("Naive Bayes: ",roc.nb$auc,"\n")
```