Taiwan Customer Defaults

Abhay Kulkarri

12/02/2019

Contents

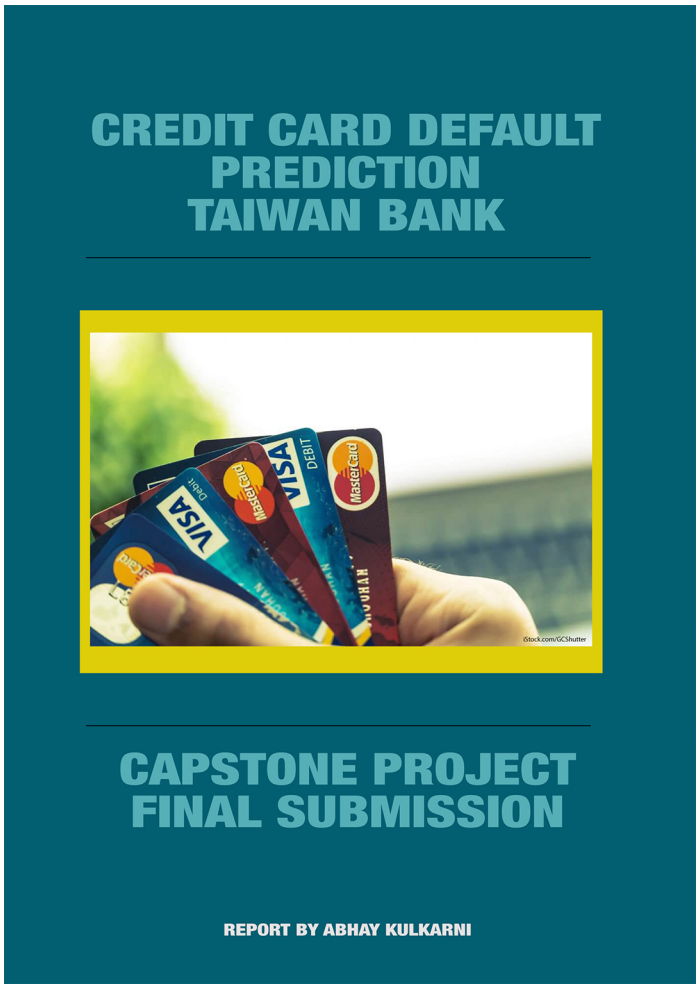


Figure 1: Taiwan Customer Defaults

1 Libraries

```
library(knitr)
library(readxl)
library(DataExplorer)
library(memisc)
library(funModeling)
library(cowplot)
library(MASS)
library(DMwR)
library(caTools)
library(DataExplorer)
library(ggplot2)
library(caTools)
library(skimr)
library(caret)
library(cowplot)
library(caTools)
library(ROSE)
library(ROCR)
library(MLmetrics)
library(MASS)
library(class)
library(e1071)
library(car)
library(ROSE)
library(MASS)
library(pROC)
library(e1071)
library(class)
library(lattice)
library(klaR)
library(ipred)
library(rpart)
library(xgboost)
library(adabag)
library(pROC)
library(rattle)
```

2 Introduction

2.1 Problem Statement

Beginning in 1990, the Taiwanese government allowed the formation of new banks. These new banks lent large sums of money to real estate companies with the goal of expanding their businesses and increasing profits. The new banks turned to other new business – credit cards and cash cards. In expanding this area of business, banks lavished money on commercials encouraging people to apply for credit cards to consume, apparently without consequences. These banks lowered the requirements for credit card approvals to get more customers.

In Taiwan, in February 2006, debt from credit cards and cash cards reached $268 billion USD. More than half a million people were not able to repay their loans. They became "credit card slaves", a term coined in Taiwan to refer to people who could only pay the minimum balance on their credit card debt every month ("News & Important policy"). This issue resulted in significant societal problems.

2.2 Need of the study

In 2005, to prevent more and more new credit card slaves from appearing, the Taiwanese Finance Supervisory Commission issued some orders to require banks to modify their requirements of credit card applications. Some of the changes included raising the income and job requirements, prohibiting improper credit card commercials, prohibiting inappropriate collection behaviors and prohibiting compound interest.

2.3 Business/Social Opportunity

A Taiwan-based bank wants to improve their prediction of defaults of their customers, as well as identify the patterns that determine this likelihood. This would help the bank decide whether to issue the credit card or not. Also, fix credit limt and risk type to the customer and avoid future defaults.

We would be analyzing the dataset and build a predictive model to identify and predict default payments.

3 Acknowledgement

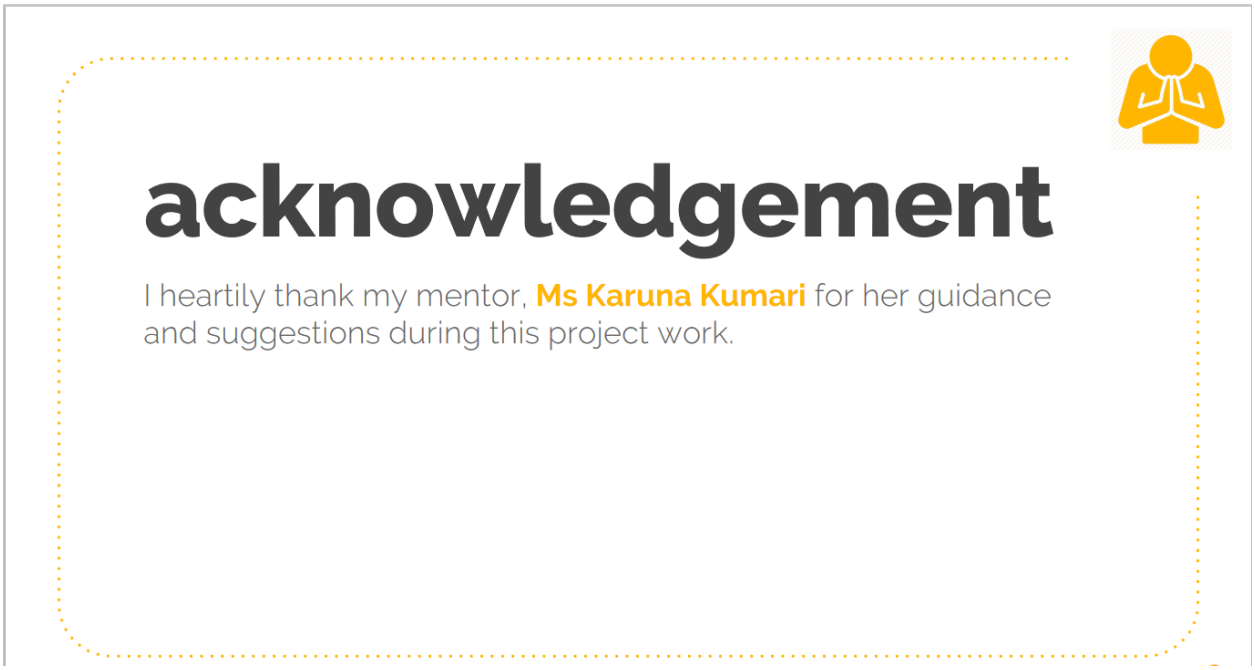


Figure 2: Acknowledgement

4 Speeding Processor Cores

```
library(parallel)
library(doParallel)
clusterforspeed <- makeCluster(detectCores() - 1) ## convention to leave 1 core for OS
registerDoParallel(clusterforspeed)
```

5 Understanding the dataset and Data Cleaning

5.1 Set Working Directory

```
setwd("Z:\\Projects\\Capstone")
getwd()
```

```
## [1] "Z:/Projects/Capstone"
```

5.2 Import dataset

```
myrawdata <- read_excel("Taiwan-Customer defaults.xls", skip = 1)
```

5.3 Data Dictionary/ Description

	Name
ID	ID of each client

	Name
LIMIT_BAL	Amount of given credit in NT dollars $includesindividualandfamily/supplementarycredit$
SEX	Gender $1 = male, 2 = female$
EDUCATION	$1 = graduateschool, 2 = university, 3 = highschool, 4 = others, 5 = unknown, 6 = unknown$
MARRIAGE	Marital status $1 = married, 2 = single, 3 = others$
AGE	Age in years
PAY_0	Repayment status in September, 2005 $2 = noconsumption, 1 = payduly, 0 = theuseofrevolvingcredit, 1 = paymentdelayforonemonth, 2 = paymentdelayfortwomonths, \ldots 8 = paymentdelayforeightmonths, 9 = paymentdelayfornինemonthsandabove$
PAY_2	Repayment status in August, 2005 $scalesameasabove$
PAY_3	Repayment status in July, 2005 $scalesameasabove$
PAY_4	Repayment status in June, 2005 $scalesameasabove$
PAY_5	Repayment status in May, 2005 $scalesameasabove$
PAY_6	Repayment status in April, 2005 $scalesameasabove$
BILL_AMT1	Amount of bill statement in September, 2005 $NTdollar$
BILL_AMT2	Amount of bill statement in August, 2005 $NTdollar$
BILL_AMT3	Amount of bill statement in July, 2005 $NTdollar$
BILL_AMT4	Amount of bill statement in June, 2005 $NTdollar$
BILL_AMT5	Amount of bill statement in May, 2005 $NTdollar$
BILL_AMT6	Amount of bill statement in April, 2005 $NTdollar$
PAY_AMT1	Amount of previous payment in September, 2005 $NTdollar$
PAY_AMT2	Amount of previous payment in August, 2005 $NTdollar$
PAY_AMT3	Amount of previous payment in July, 2005 $NTdollar$
PAY_AMT4	Amount of previous payment in June, 2005 $NTdollar$
PAY_AMT5	Amount of previous payment in May, 2005 $NTdollar$
PAY_AMT6	Amount of previous payment in April, 2005 $NTdollar$
default.payment.next.month	Default payment $1 = yes, 0 = no$

5.4 Convert to Data Frame

```
myrawdata<- as.data.frame(myrawdata)
```

5.5 Understanding how data was collected in terms of time and frequency

```
##   ID LIMIT_BAL SEX EDUCATION MARRIAGE AGE PAY_0 PAY_2 PAY_3 PAY_4 PAY_5 PAY_6
## 1  1     20000   2         2        1  24     2     2    -1    -1    -2    -2
## 2  2    120000   2         2        2  26    -1     2     0     0     0     2
## 3  3     90000   2         2        2  34     0     0     0     0     0     0
## 4  4     50000   2         2        1  37     0     0     0     0     0     0
## 5  5     50000   1         2        1  57    -1     0    -1     0     0     0
## 6  6     50000   1         1        2  37     0     0     0     0     0     0
##   BILL_AMT1 BILL_AMT2 BILL_AMT3 BILL_AMT4 BILL_AMT5 BILL_AMT6 PAY_AMT1 PAY_AMT2
## 1      3913      3102       689         0         0         0        0      689
## 2      2682      1725      2682      3272      3455      3261        0     1000
## 3     29239     14027     13559     14331     14948     15549     1518     1500
## 4     46990     48233     49291     28314     28959     29547     2000     2019
## 5      8617      5670     35835     20940     19146     19131     2000    36681
## 6     64400     57069     57608     19394     19619     20024     2500     1815
##   PAY_AMT3 PAY_AMT4 PAY_AMT5 PAY_AMT6 default payment next month
## 1        0        0        0        0                          1
```

```
## 2     1000     1000        0     2000                          1
## 3     1000     1000     1000     5000                          0
## 4     1200     1100     1069     1000                          0
## 5    10000     9000      689      679                          0
## 6      657     1000     1000      800                          0
```

Findings

- The dataset contains information on default payments, demographic factors, credit data, history of payment, and bill statements of credit card clients in Taiwan from April 2005 to September 2005.

5.6 Dimension (Rows and Column in the dataset)

```
## [1] 30000    25
```

Findings

- There are 30000 observations and 25 Features

5.7 Number of Discreate and Continous Variables

```
##    rows columns discrete_columns continuous_columns all_missing_columns
## 1 30000      25               0                 25                   0
##   total_missing_values complete_rows total_observations memory_usage
## 1                    0         30000             750000      6005856
```

Findings

- The above table incorrectly reads dataset as 25 Continuos and 0 Discreate
- Have to convert 'Sex', 'Education', 'Marriage' and 'Default Payment'as factors.

5.8 Converting incorrectly read data types to factors

```
##    rows columns discrete_columns continuous_columns all_missing_columns
## 1 30000      25               4                 21                   0
##   total_missing_values complete_rows total_observations memory_usage
## 1                    0         30000             750000      5528360
```

Findings

- Have converted 'Sex', 'Education', 'Marriage' and 'Default Payment'as factors.

5.9 Will look into pay_1,2,3,4,5,6 later

- For fields pay1 to pay6 , roughly 50% of them have 0s and there are many -2s as well.
- Cannot conclude anything about the data right now.
- PAY_* is an ordinal variable where the levels are ordered and have some meaning

Only exploring the data further can reveal some insights

Retaining it as numeric for the moment

5.10 Check for data summary/ data details

```
##                     variable q_zeros p_zeros q_na p_na q_inf p_inf    type
## 1                         ID       0    0.00    0    0     0     0 numeric
## 2                  LIMIT_BAL       0    0.00    0    0     0     0 numeric
## 3                        SEX       0    0.00    0    0     0     0  factor
```

```
## 4                    EDUCATION      14    0.05    0    0     0     0  factor
## 5                     MARRIAGE      54    0.18    0    0     0     0  factor
## 6                          AGE       0    0.00    0    0     0     0 numeric
## 7                        PAY_0   14737   49.12    0    0     0     0 numeric
## 8                        PAY_2   15730   52.43    0    0     0     0 numeric
## 9                        PAY_3   15764   52.55    0    0     0     0 numeric
## 10                       PAY_4   16455   54.85    0    0     0     0 numeric
## 11                       PAY_5   16947   56.49    0    0     0     0 numeric
## 12                       PAY_6   16286   54.29    0    0     0     0 numeric
## 13                   BILL_AMT1    2008    6.69    0    0     0     0 numeric
## 14                   BILL_AMT2    2506    8.35    0    0     0     0 numeric
## 15                   BILL_AMT3    2870    9.57    0    0     0     0 numeric
## 16                   BILL_AMT4    3195   10.65    0    0     0     0 numeric
## 17                   BILL_AMT5    3506   11.69    0    0     0     0 numeric
## 18                   BILL_AMT6    4020   13.40    0    0     0     0 numeric
## 19                    PAY_AMT1    5249   17.50    0    0     0     0 numeric
## 20                    PAY_AMT2    5396   17.99    0    0     0     0 numeric
## 21                    PAY_AMT3    5968   19.89    0    0     0     0 numeric
## 22                    PAY_AMT4    6408   21.36    0    0     0     0 numeric
## 23                    PAY_AMT5    6703   22.34    0    0     0     0 numeric
## 24                    PAY_AMT6    7173   23.91    0    0     0     0 numeric
## 25 default payment next month   23364   77.88    0    0     0     0  factor
##    unique
## 1   30000
## 2      81
## 3       2
## 4       7
## 5       4
## 6      56
## 7      11
## 8      11
## 9      11
## 10     11
## 11     10
## 12     10
## 13  22723
## 14  22346
## 15  22026
## 16  21548
## 17  21010
## 18  20604
## 19   7943
## 20   7899
## 21   7518
## 22   6937
## 23   6897
## 24   6939
## 25      2
```

5.11 Change names of few columns

Findings

- To have similar column names, changing "PAY_0" to "PAY_1"

- Column default.payment.next.month rename to DEFAULT

5.12 Converting data levels of category

Findings

- Converting Default factor from “0” and “1” to “No” and “Yes”
- Converting Marriage “1”, “2” and “3” to “Married”, “Single” and “Other”
- As there is no description for “5” and “6”. Converting Education to “Graduate.School”, “University”, ""High.School” and “Unknown”.

5.13 Let’s check the above conversion

```
##
## Married   Other  Single
##   13659     377   15964
```

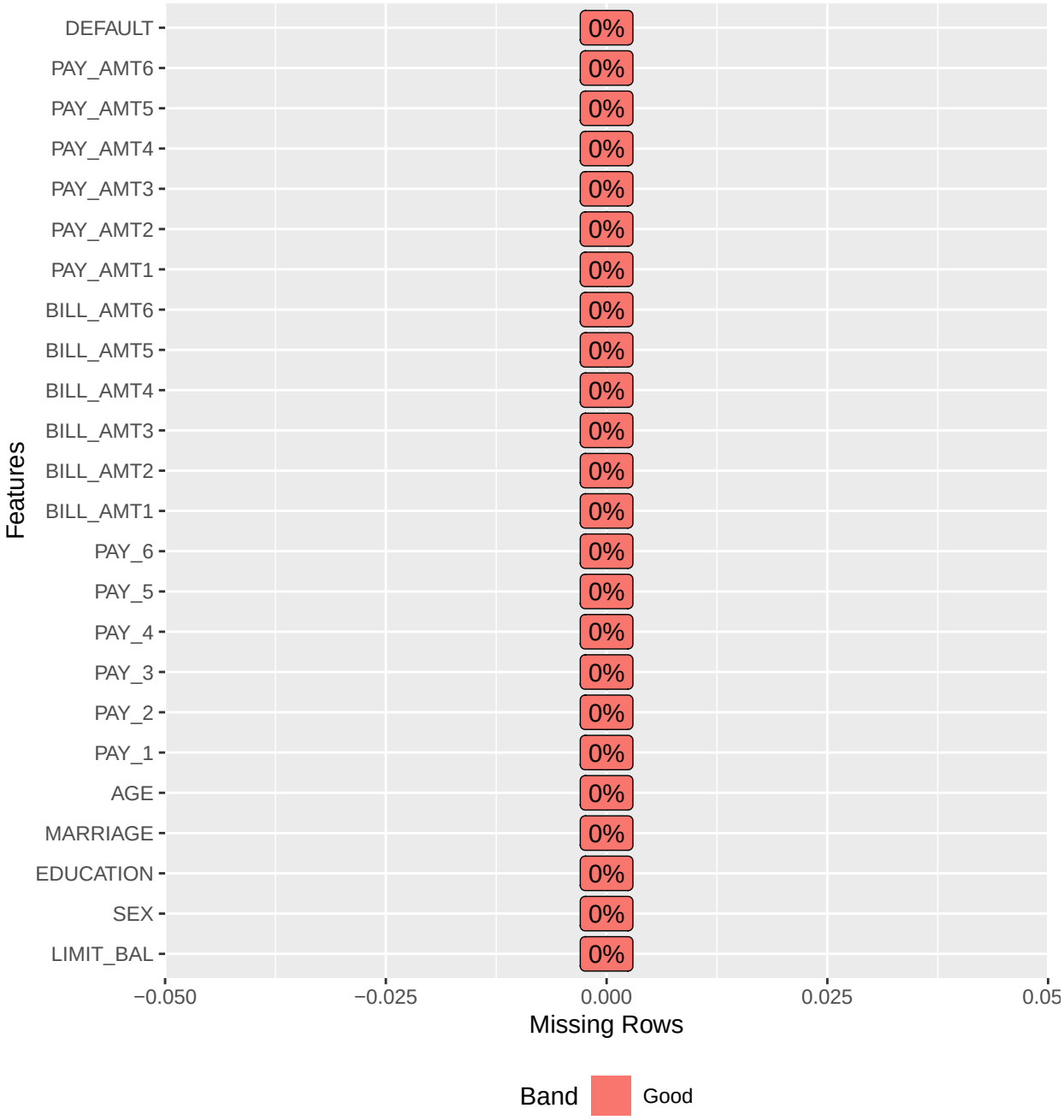
```
##
## Female   Male
##  18112  11888
```

```
##
## Graduate.School     High.School           Other      University          Unkown
##           10585            4917             123           14030             345
```

5.14 Before we begin EDA. Let’s get rid off ID column

5.15 Create backup and proceed with EDA

5.16 Check for Missing Values

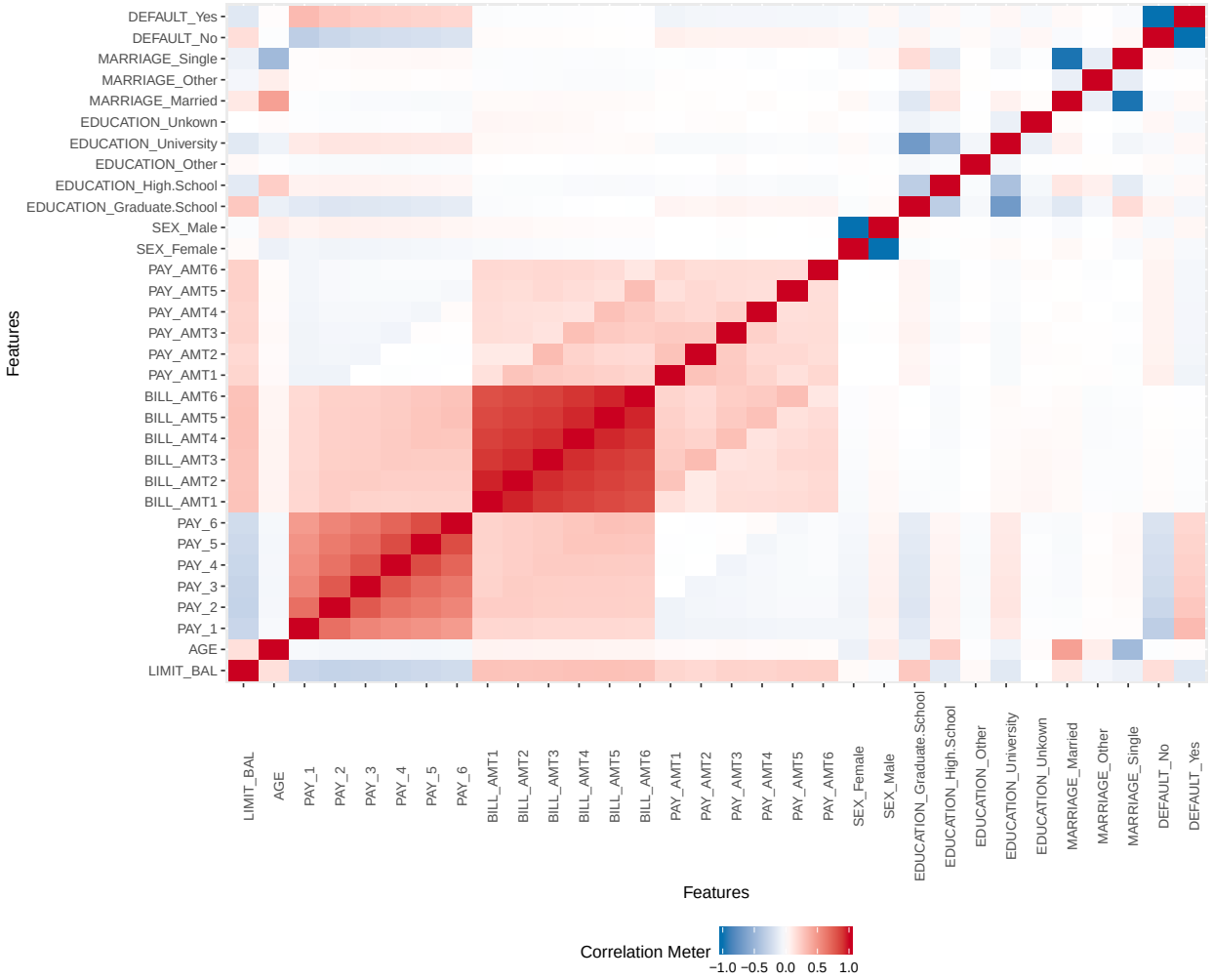


Findings

- There are no missing values

6 EDA

6.1 Before Univariate Analysis. A quick check with corr plot and understand pattern

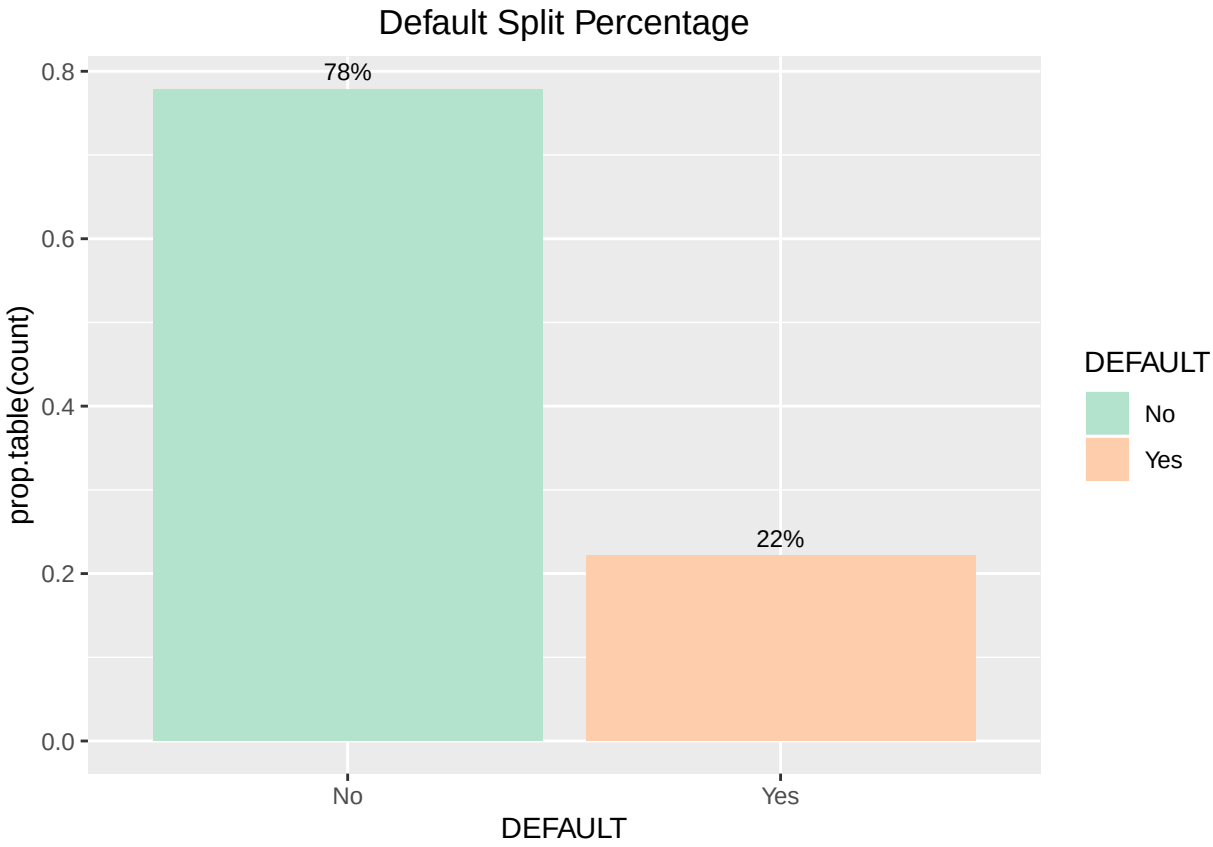


Findings

- Look at DEFAULT correlation with other variables.
- Lowest is with LIMIT_BAL.LIMIT_BAL and DEFAULT are Negatively Correlated. Negative correlation indicates higher Credit Limit, lower Default.
- Highest is correlation with PAY_1. PAY_1 and DEFAULT are positively correlated. Positive correlation indicates longer period of Delay Payment, higher Default.
- In general PAY_1 ~ PAY_6 have higher correlation to DEFAULT compare to other variables.
- Clients payment behaviour give strong indication on Default.

6.2 UNIVARIATE ANALYSIS. Let's start with Categorical Variable

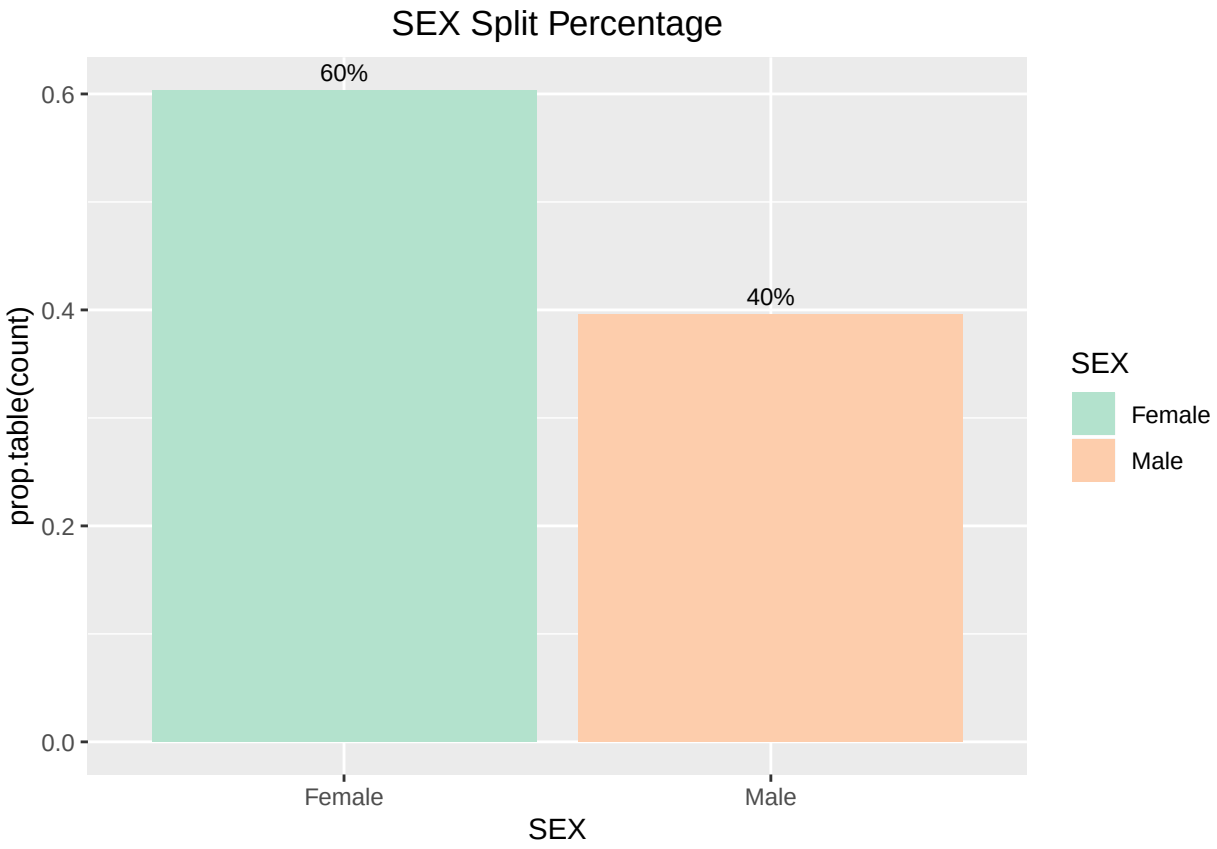
6.3 Check for dependent(DEFAULT) column split



Findings

- The dependent data is not evenly distributed. We have more of "NO 78% and 22%"YES"
- Dataset of Bank Credit Defaults are good examples of imbalanced data. We'll check if the data needs to be balanced during Model Building.

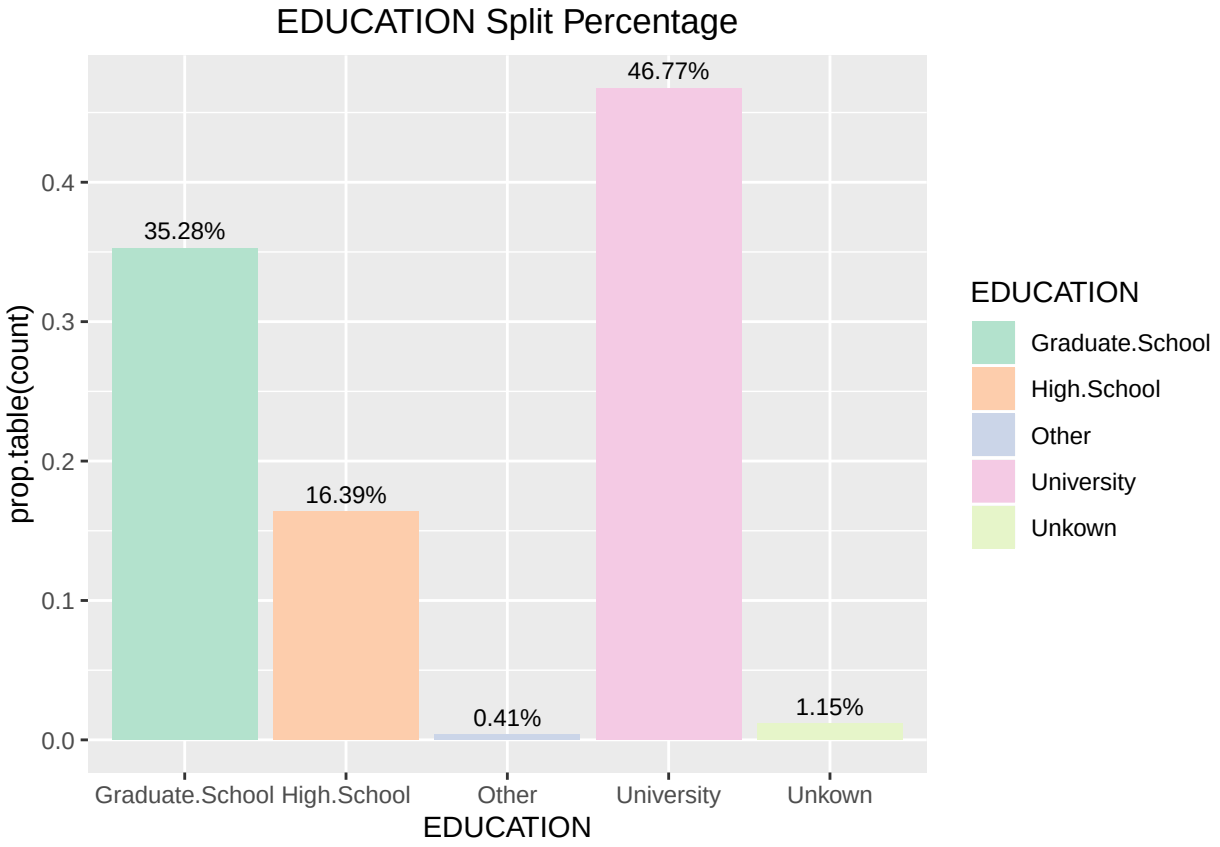
6.4 Let's check percentage of customer based on SEX



Findings

- There are more Female credit card holers than male.
- There are 20% more female customer than male. This could be an important feature to build deault prediction model. We'll investigate further during bivariate analysis(gender vs default)

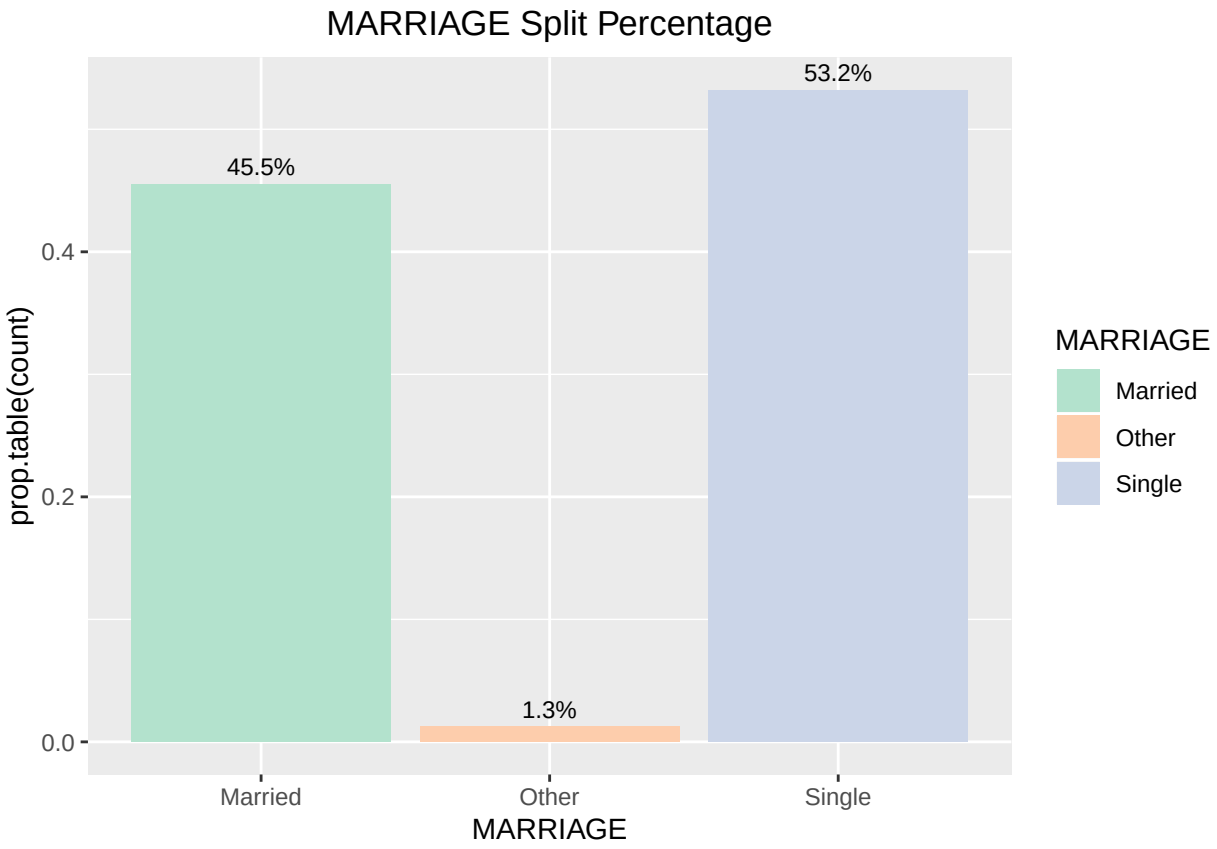
6.5 Let's check percentage of customer based on EDUCATION



Findings

- We can immediately see that "Other" and "Unknown" are negligible. However, there are more credit card holders who have "University" level education, followed by "Graduate School" and "High School"
- We" investigate this further during Bivariate Analysis and check if this is an important feature to preidct default.

6.6 Let's check percentage of customer based on MARRIAGE

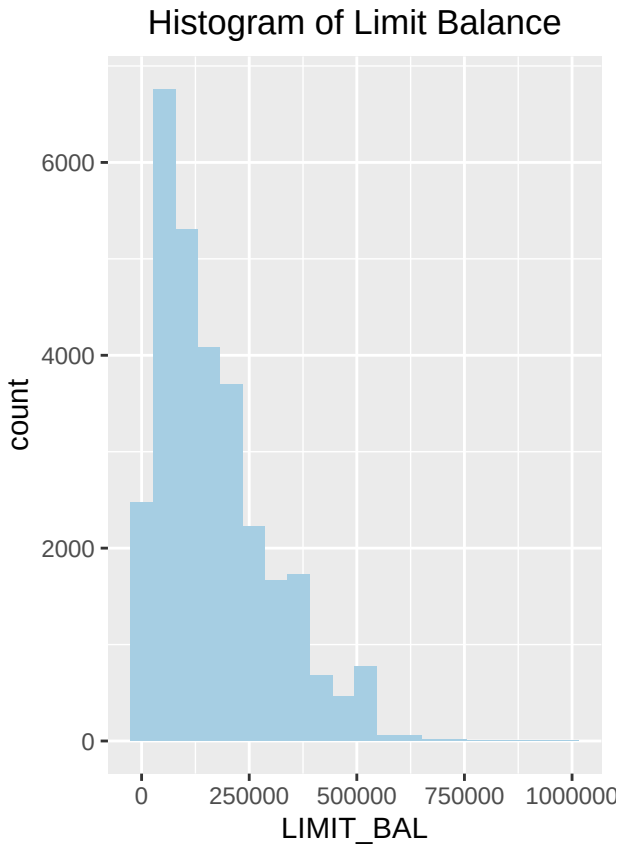


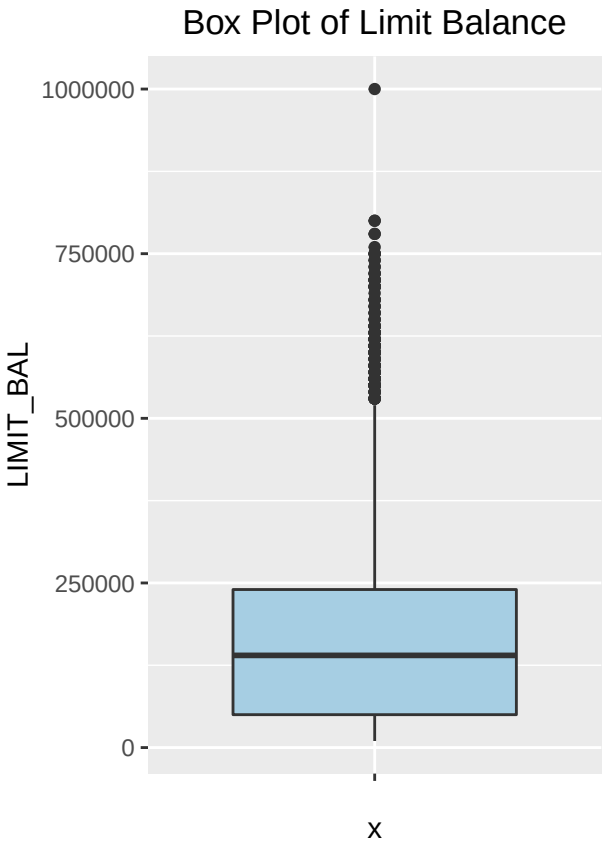
Findings

- We can immediately see that "Other" has negligible data. However, there is difference of 7.7% between "Married" and "Single" customers.

6.7 Let's investigate Continous features

6.8 Let's check the distribution and outliers(if any) of Limit Balance





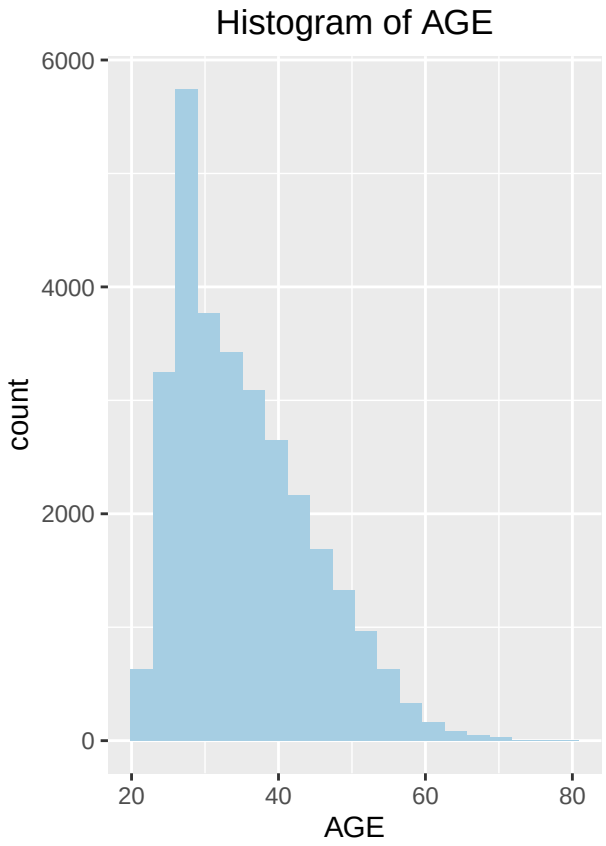

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10000   50000  140000  167484  240000 1000000
```

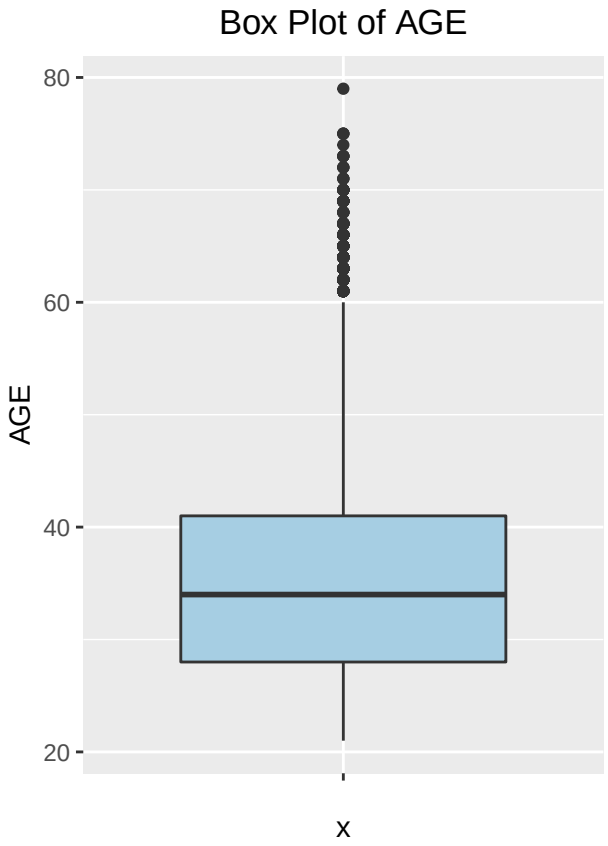
```
## [1] 129747.7
```

Findings

- Histogram : By looking at histogram we can see that it is a bit right skewed. The **Mean** of distribution is **167484** and **Standard Deviation** of **129747.7**
- Box Plot : We can see that there are outliers present per the BOX PLOT. We'll treat it later (Notes2 Submission). Would treat outliers using Winsorizing transformation.
- Min Value is 10000, Q1 is 50000, Median is 140000, Mean is 167484, Q3 is 240000 and Max is 1000000.

6.9 AGE





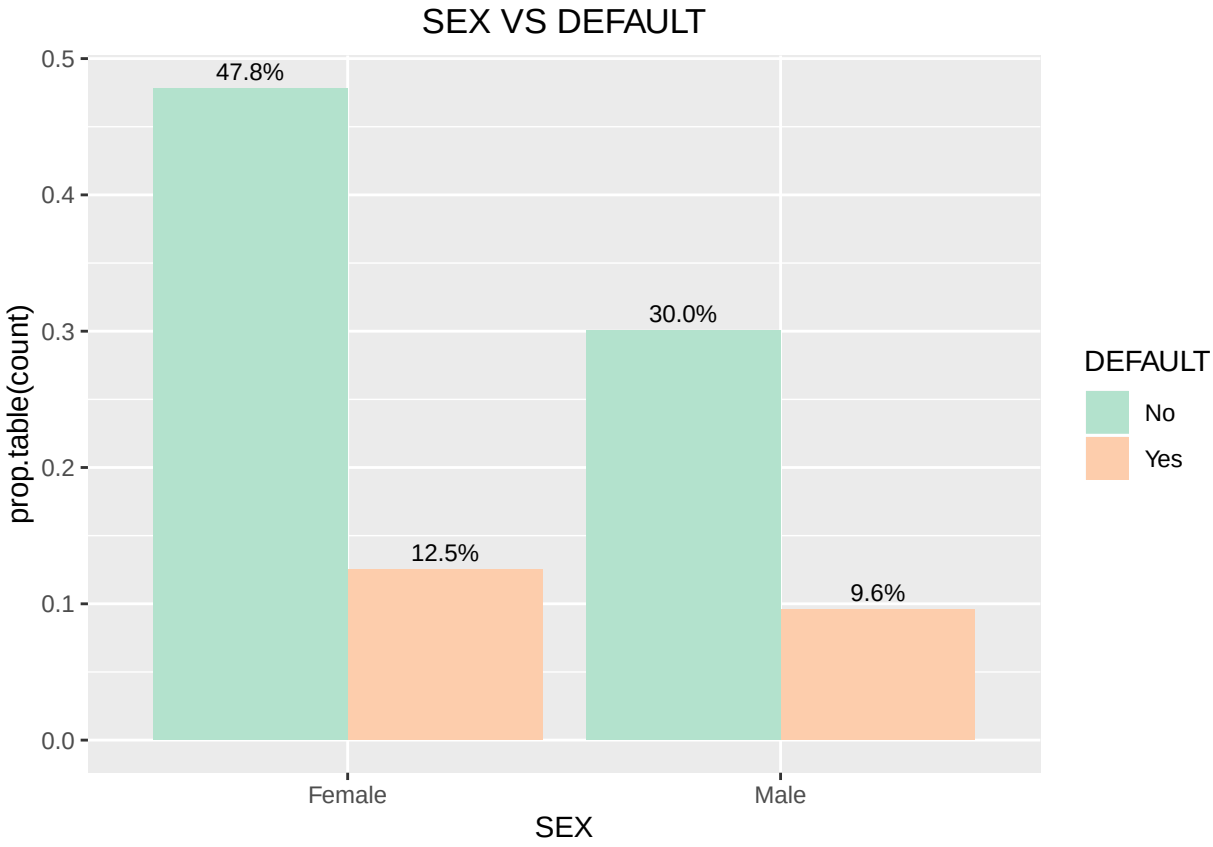

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   21.00   28.00   34.00   35.49   41.00   79.00
```

```
## [1] 9.217904
```

7 Bivariate Analysis

7.1 Let's see how each feature reacts with dependent feature, DEAFULT. Let's start Bivariate Analysis with Categorical Features.

7.2 Let's check if there is significant difference between "Male" and "Female" with respect to Default



Findings

- From the above figure. We can see that the Default of Female is 12.5% and Male is 9.6%.
- Let's check Chi-Square test of Independence

Test the hypothesis whether the Default is independent of the SEX level at .05 significance level.

Null Hypothesis(HO) : Default is Independent of SEX

Alternative Hypothesis(H1) : Default is dependent on SEX

```
##
##             No   Yes
##   Female 14349  3763
##   Male    9015  2873
```

Let's run Chi-Square test

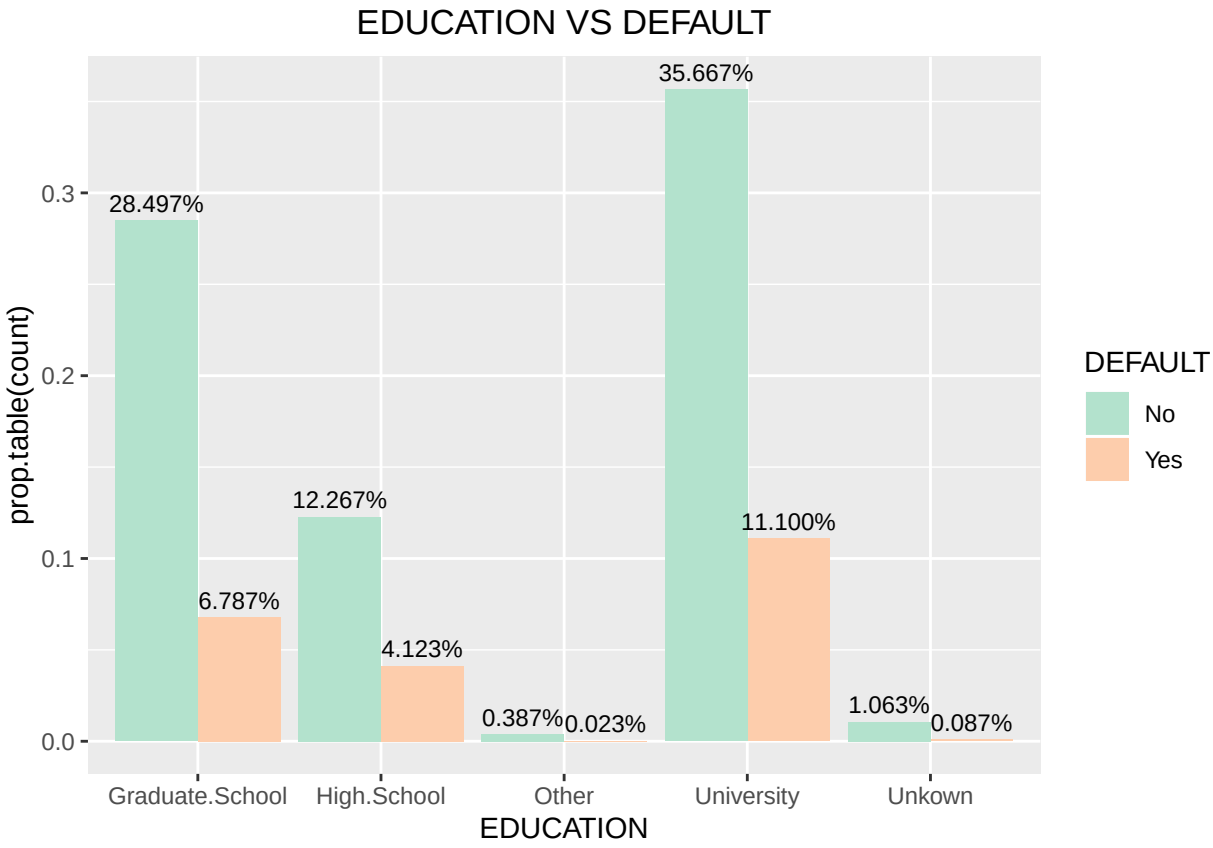
```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
```

```
## data:  SEXtable
## X-squared = 47.709, df = 1, p-value = 4.945e-12
```

Findings

- p- value is way lesser than 0.05. We reject Null Hypothesis and go with Alternative hypothesis that Default" depends on "SEX".
- SEX is an important feature to distinguish between Default vs No Default.

7.3 Let's check EDUCATION VS DEFAULT




```
##
##                     No   Yes
##   Graduate.School  8549  2036
##   High.School      3680  1237
##   Other             116     7
##   University      10700  3330
##   Unkown            319    26
```

Let's run Chi-Square test

```
##
##  Pearson's Chi-squared test
##
## data:  EducationTable
## X-squared = 160.59, df = 4, p-value < 2.2e-16
```

Findings

- From the above figure. We can see that the difference Default percentage of University is the most at 24.56% followed by Graduate School at 21.71%
- Let's check Chi-Square test of Independence

Test the hypothesis whether the Default is independent of the EDUCATION level at .05 significance level.

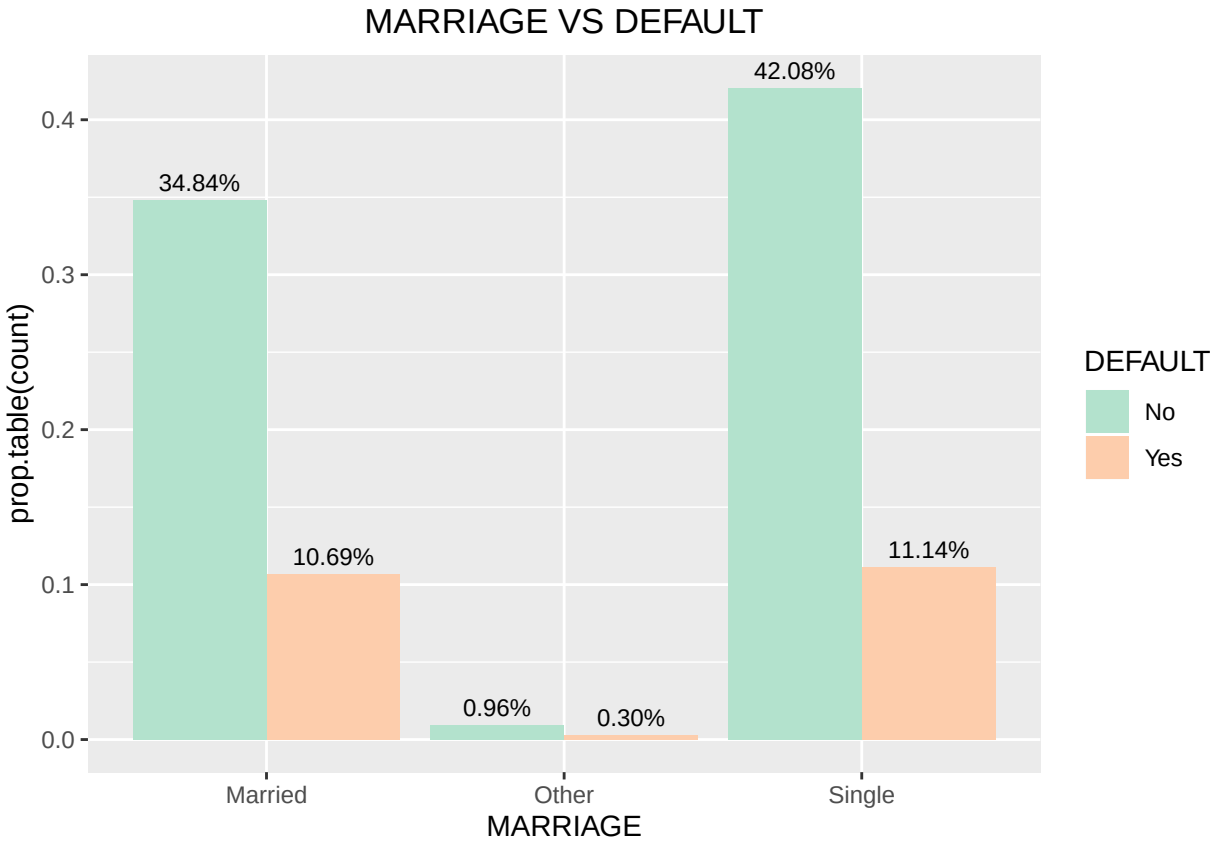
Null Hypothesis(HO) : Default is Independent of EDUCATION

Alternative Hypothesis(H1) : Default is dependent on EDUCATION

Findings

- p- value is way lesser than 0.05. We reject Null Hypothesis and go with Alternative hypothesis that Default" depends on "EDUCATION".
- EDUCATION is an important feature to distinguish between Default vs No Default.

7.4 MARRIAGE VS DEFAULT



Findings

- From the above figure. It looks like "Married" and "Single" default percentage is very close.
- Lets run Chi Square test of independence

```
##
##       Married Other Single
##   No    10453   288  12623
##   Yes    3206    89   3341
```

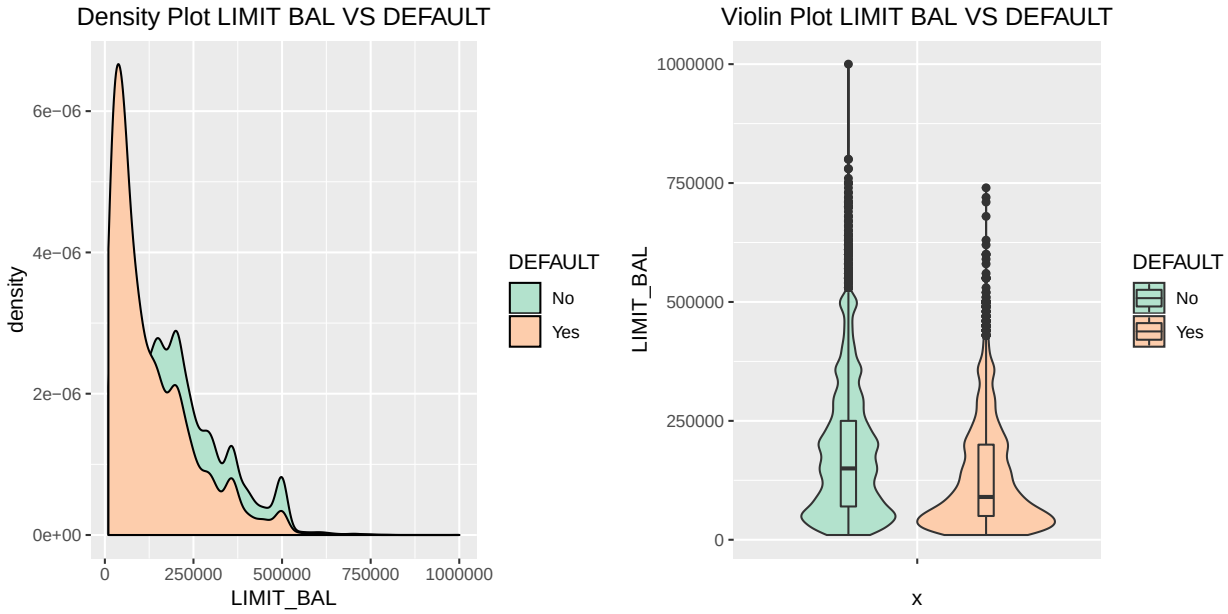
Let's run Chi-Square test

```
##
##  Pearson's Chi-squared test
##
## data:  MarriageTable
## X-squared = 28.13, df = 2, p-value = 7.791e-07
```

Findings

- p- value is way lesser than 0.05. We reject Null Hypothesis and go with Alternative hypothesis that Default" depends on "MARRIAGE".
- MARRIAGE is an important feature to distinguish between Default vs No Default.

7.5 Let's start BiVariate Analysis with Numerical feature VS Dependent Feature(DEFAULT). Let's dig in with the information received from Correlation Plot



Findings

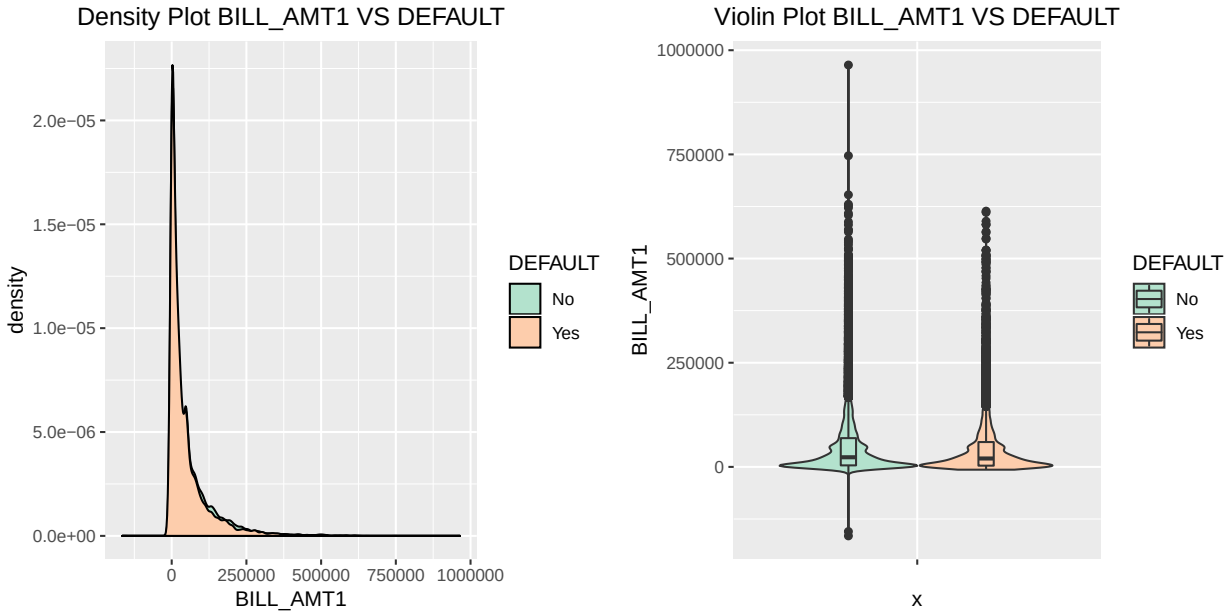
- We can see the inverse relationship between LIMIT BAL and DEFAULT. Lesser the BALANCE More the DEFAULT

7.6 AGE VS DEFAULT

Findings

- There are more defaulters between Age 20 and Age 25

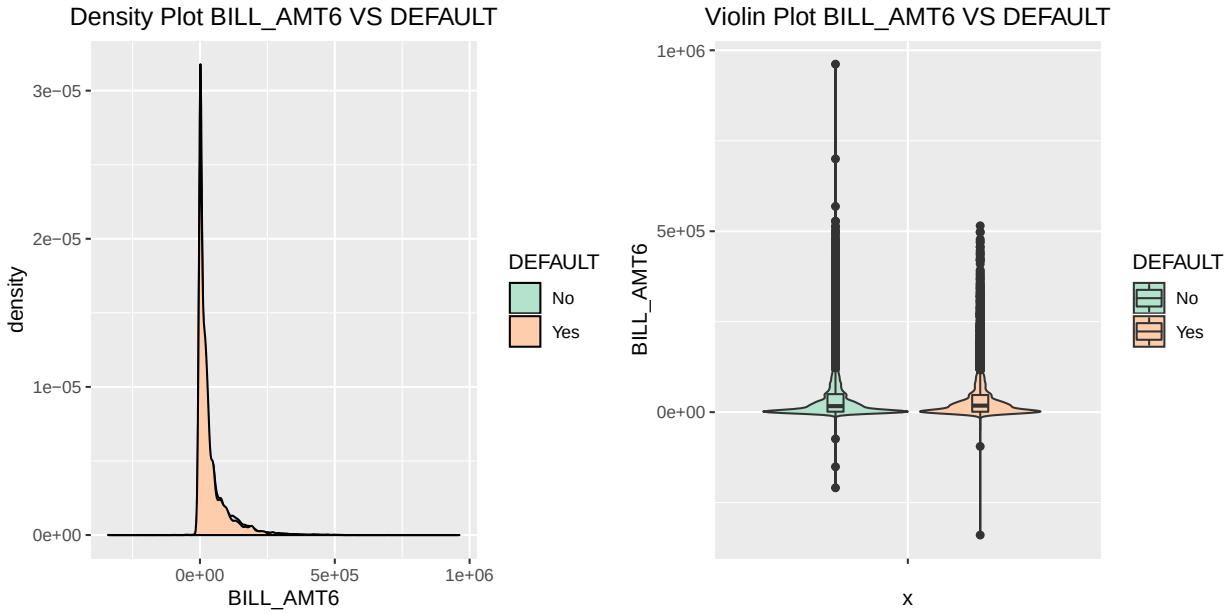
7.7 BILL_AMT1 VS DEFAULT



Findings

- It appears that there is more defaults between **BILL AMT 0 and 250000**

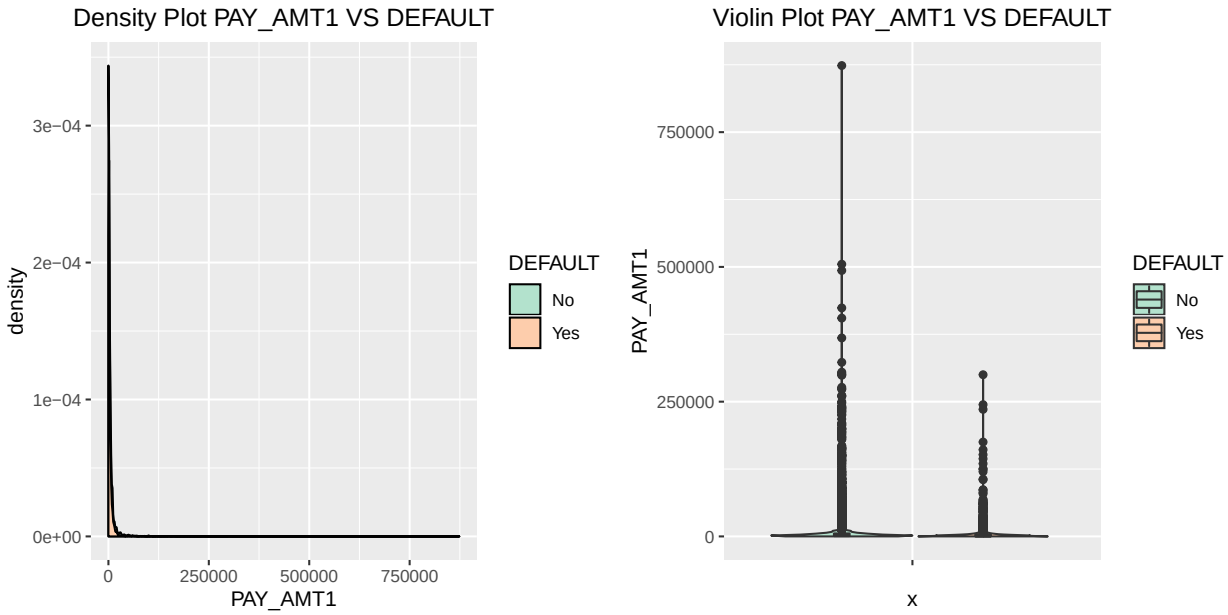
7.8 BILL_AMT6 VS DEFAULT



Findings

- It appears that there is more defaults towards initial Billing AMT 6 and DEFAULTS gradually reduces as Billing Increases.

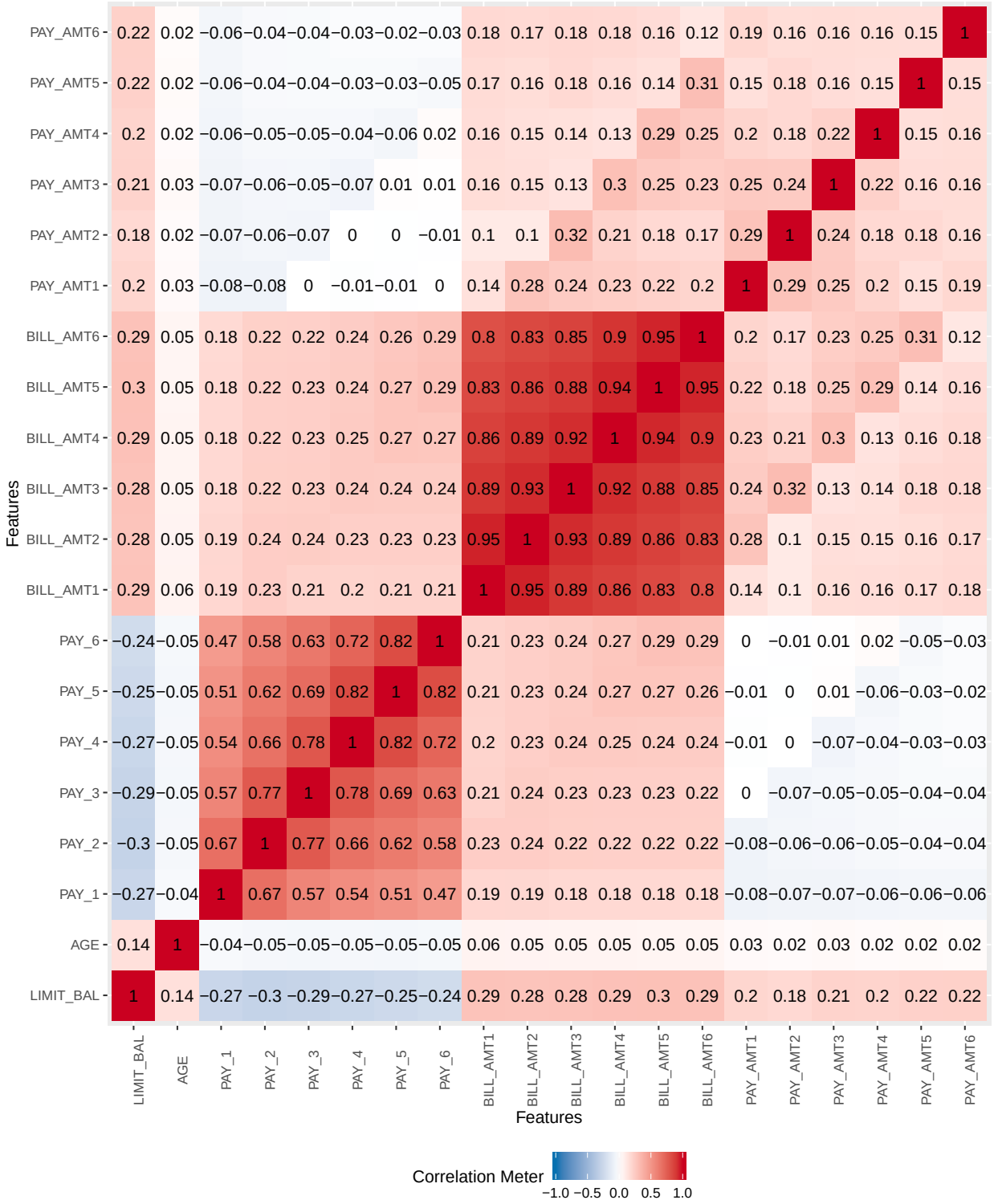
7.9 PAY_AMT1 VS DEFAULT



Findings

- We can immediately see that there are more 0 values in the PAY_AMT1 column. Also, it is directly related to defaults. Amount due and unpaid will result in defaults.

7.10 Finally let's plot correlation plot between only numberical variables



Findings

- Billing AMT 1 to Billing AMT6 are highly correlated. These highly correlated features can undergo dimension reduction using PCA.

- PAY AMT1 to PAY AMT6 are highly correlated. These highly correlated features can undergo dimension reduction using PCA.
- PAY1 to PAY 6 needs further investigation. These features will be handeled in future.

8 Summarise by asking some questions

8.1 Defaulters are more in which Age bracket?

- There are more defaulters between Age 20 and Age 25

8.2 Any effect of Education (level) on Default?

- High School and Graduate school education holders are likely to Default more.

8.3 Did you find any any gender bias in extending credits?

- There are more FEMALE credit card holders. 20% more to be precise.

8.4 More Defaulters belong to which Gender?

- There are more FEMALE defaulters compared to MEN. FEMALE defaulters are 12.5% VS 9.6% MEN

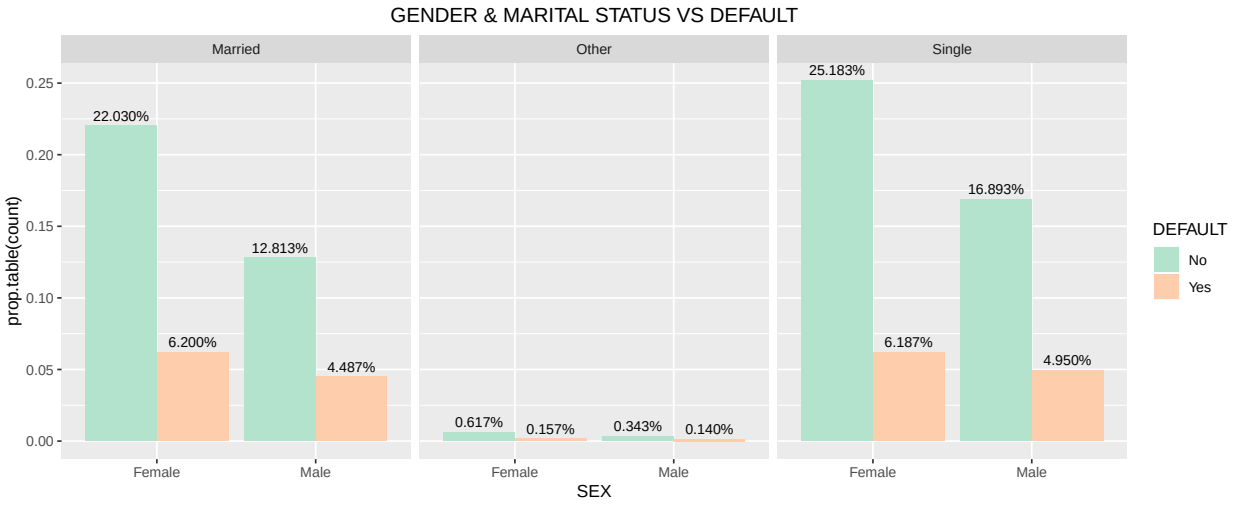
8.5 Married people taking more credits than single?

- No, it's the other way around. Single Customers are taking more loans than Married customers. MARRIED customers are 45.5% VS 53.2% SINGLE customers.

8.6 Who are more defaulters – Single or Married?

- Single Customers are more defaulters compared to Married Customers. MARRIED 10.69% VS SINGLE 11.14%

8.7 Does Gender and Marital Status has any role on Defaults?



- FEMALE customers are more likely to default regardless of marital status.

9 NEXT STEPS (Notes 2)

9.1 Outlier Treatent using winsorizing method.

9.2 Feature creation

9.3 Numerical variables AGE and Other variables are on differnt scales.Normalization or Standardization of data will be done.

9.4 Build classification model based on variable importance.

10 Notes 2 Roadmap.

10.1 Detailed EDA would include several aspects some of those are mentioned below:

1. Renaming of variables
2. Remove the variables that are not required
3. Outlier treatment
4. New features creation
5. EDA of new features
6. Binning of a particular variable (for example- Age) (as needed)
7. Merging or combining different values under one category for any variable(for example for a categorical variable-education level 4 and above can be grouped together) (as needed)
8. Any such modifications would also involve EDA
9. Identification of important variables
10. Multicollinearity
11. List the observations from the above steps
12. Dividing the data into Train and Test datasets.
13. Assess if SMOTE is required
14. Make data sets (for example actual train, smoted train, normalized actual train, normalized smoted train , dataset post PCA etc.) While you make these data sets, please add the objective(why are you doing this).
15. List out different models/algorithms you think appropriate to be used in the context of the problem statement.
16. For each method you would list the name, definition and how is it going to help in this case.
17. Please elaborate on all or any of the above-mentioned topics as you think appropriate.

CREDIT CARD DEFAULT PREDICTION

CAPSTONE PROJECT NOTES 2

REPORT BY ABHAY KULKARNI

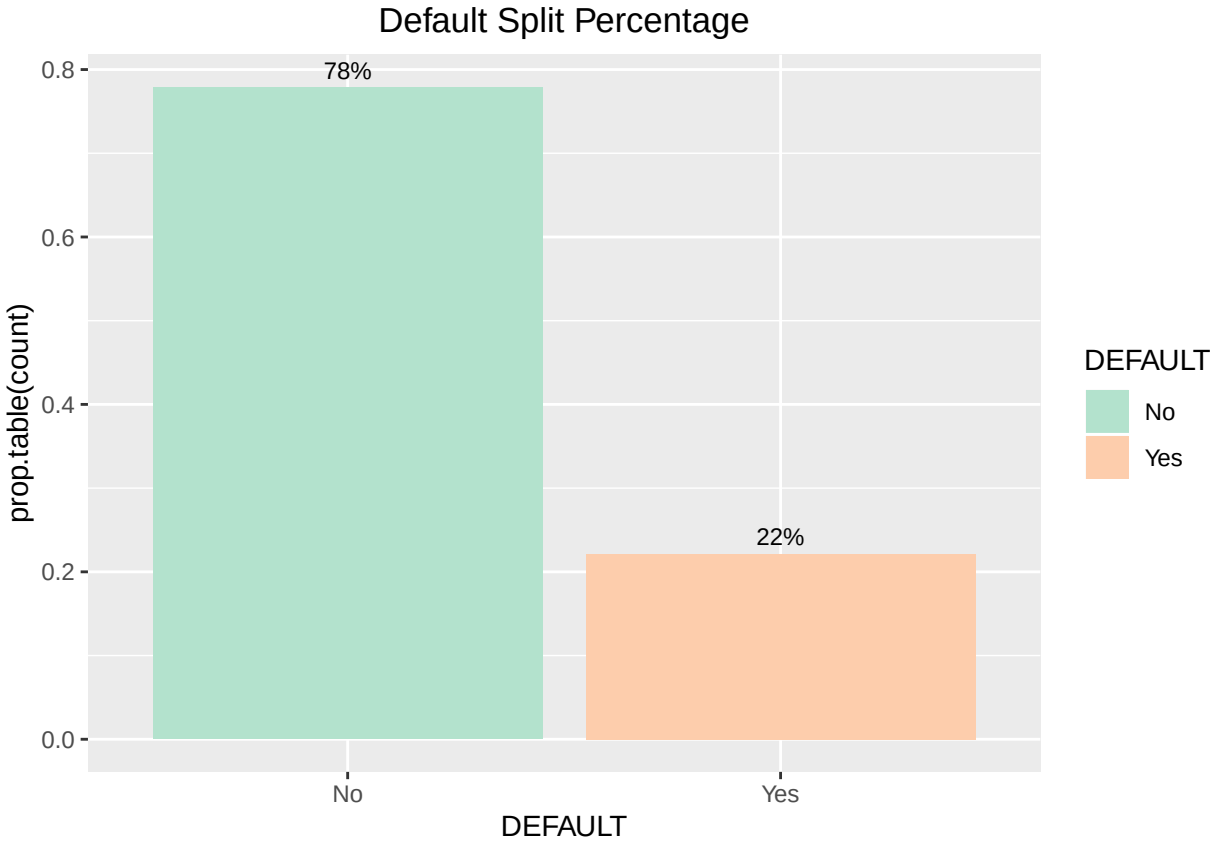
Figure 3: Taiwan Customer Defaults Notes2

11 Understanding the dataset and Data Cleaning

Findings

- There are no missing values

12 Check for dependent(DEFAULT) column split



Findings

- The dependent data is not evenly distributed. We have more of “NO 78% and 22%”YES"
- Dataset of Bank Credit Defaults are good examples of imbalanced data.
- There will be multiple datasets created. One Balanced dataset using SMOTE will also be created.

13 Check the data

```
##    rows columns discrete_columns continuous_columns all_missing_columns
## 1 30000      24               10                 14                   0
##   total_missing_values complete_rows total_observations memory_usage
## 1                    0         30000             720000      4574624
```

Findings

- Have converted ‘Sex’, ‘Education’, ‘Marriage’ and ’Default Payment’as factors.

14 Change names of few columns

Findings

- To have similar column names, changing "PAY_0" to "PAY_1"
- Column default.payment.next.month rename to DEFAULT

15 Merging or combining different values under one category. Converting data levels of category

Findings

- Converting Default factor from "0" and "1" to "No" and "Yes"
- Converting Marriage "1", "2" and "3" to "Married", "Single" and "Other"
- As there is no description for "5" and "6". Converting Education to "Graduate.School", "University", ""High.School" and "Unknown".

16 Let's check the above conversion

```
##
## Married   Other  Single
##   13659     377   15964
```

```
##
## Female   Male
##  18112  11888
```

```
##
## Graduate.School     High.School           Other      University          Unkown
##           10585            4917             123           14030             345
```

17 Before treating Outliers. Let's plot some scaatter plot

17.1 To check the reationship between numericals data. AGE VS LIMIT BAL

Findings

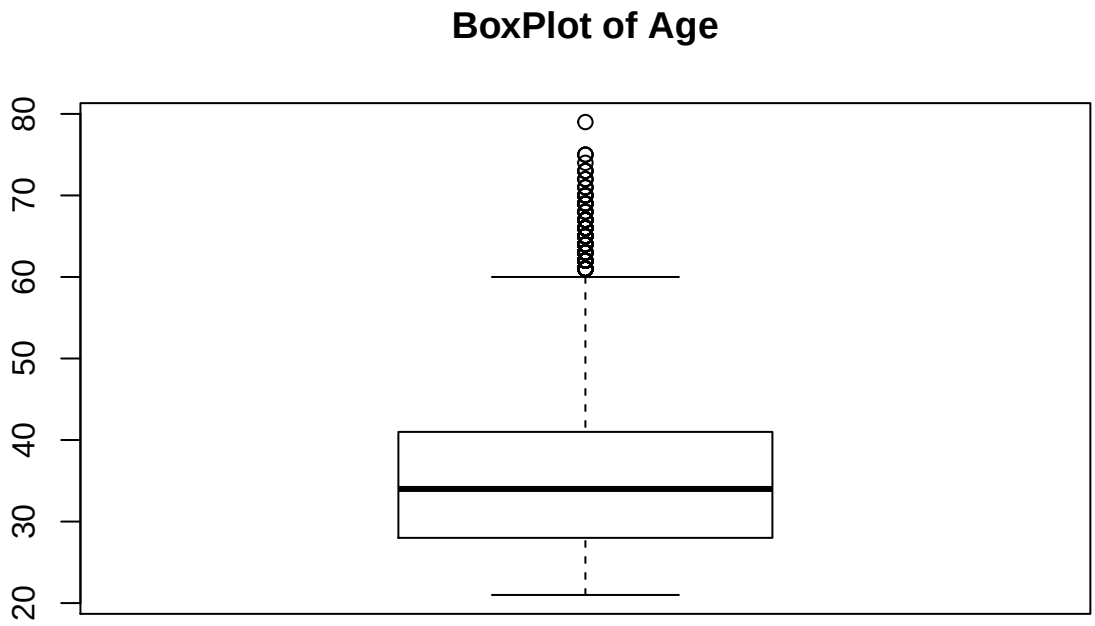
- We see that between AGE 25 and 50. There is direct relationship between AGE and LIMIT BAL. Also, we see that LIMIT BAL more than 500000 results in less default.

18 Create backup and proceed with EDA

19 Check Outliers and treat them.

Will be creating Multiple datasets. ActualDataset, Outlier treated, Standardized and Standardised SMOTE

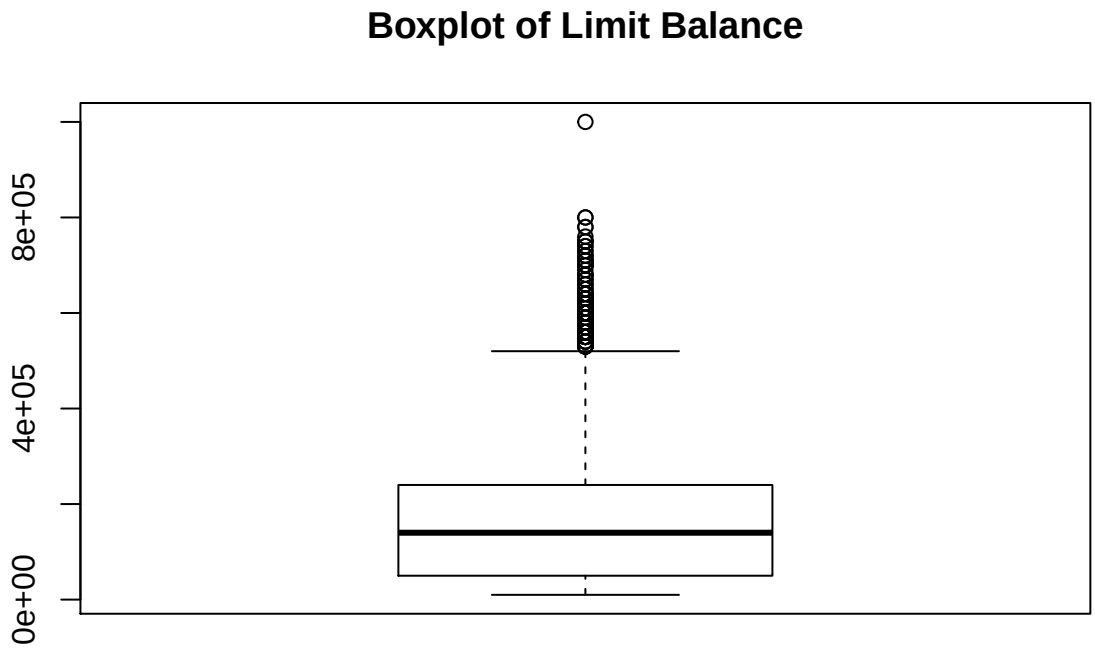
19.1 Boxplot of AGE



Findings

- We may not want to treat AGE as an outlier. It represents true distribution of customer by age.

19.2 Boxplot of Limit Balance and Outlier Treatment




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10000   50000  140000  167484  240000 1000000
```

19.2.1 Treating Outlier for Limit Balance using Winsorizing.

```
## [1] 190000
```

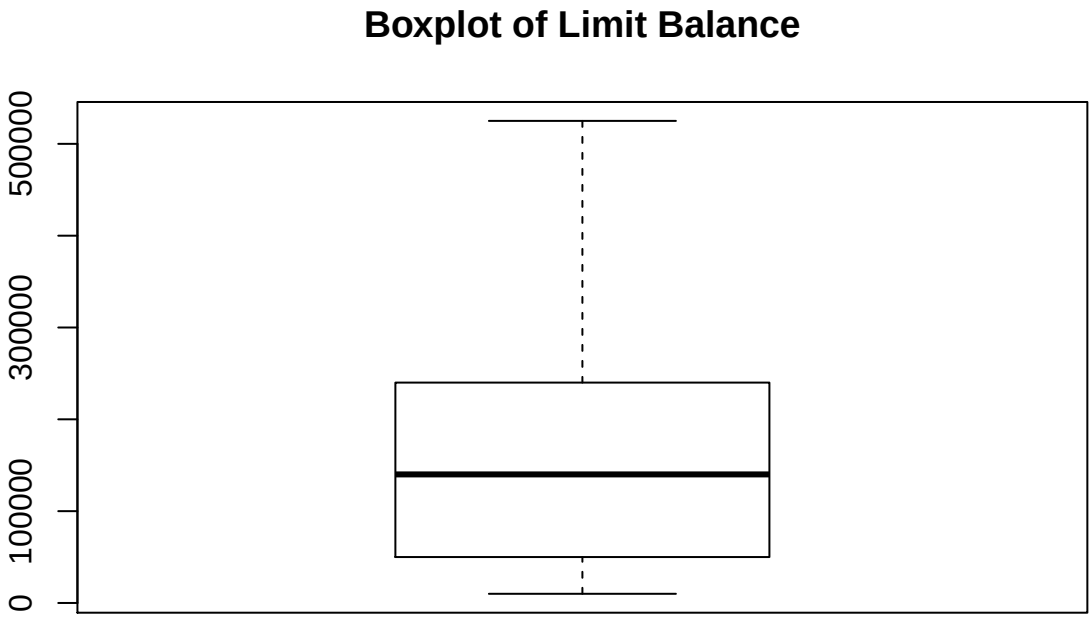
```
## [1] 525000
```

```
## [1] -235000
```

Findings

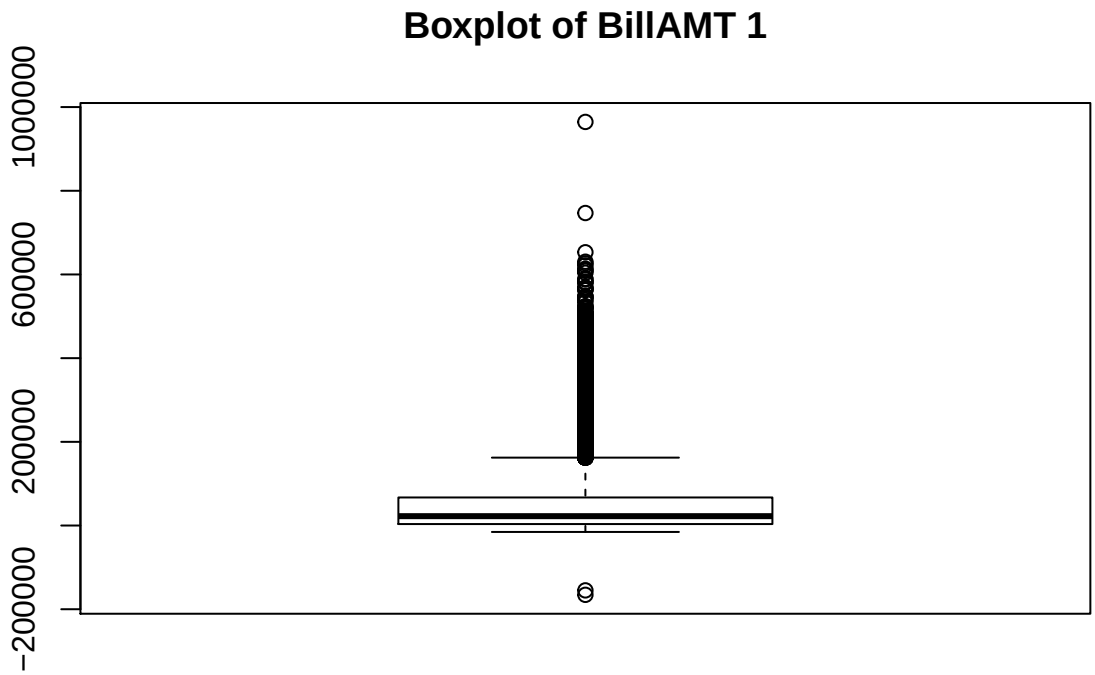
- Any observation above 525000 will be assigned the value of 525000

19.2.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10000   50000  140000  166968  240000  525000
```

19.3 Boxplot of BillAMT 1




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -165580    3559   22382   51223   67091  964511
```

19.3.1 Treating Outlier for BILLAMT1 using Winsorizing.

```
## [1] 63532.25
```

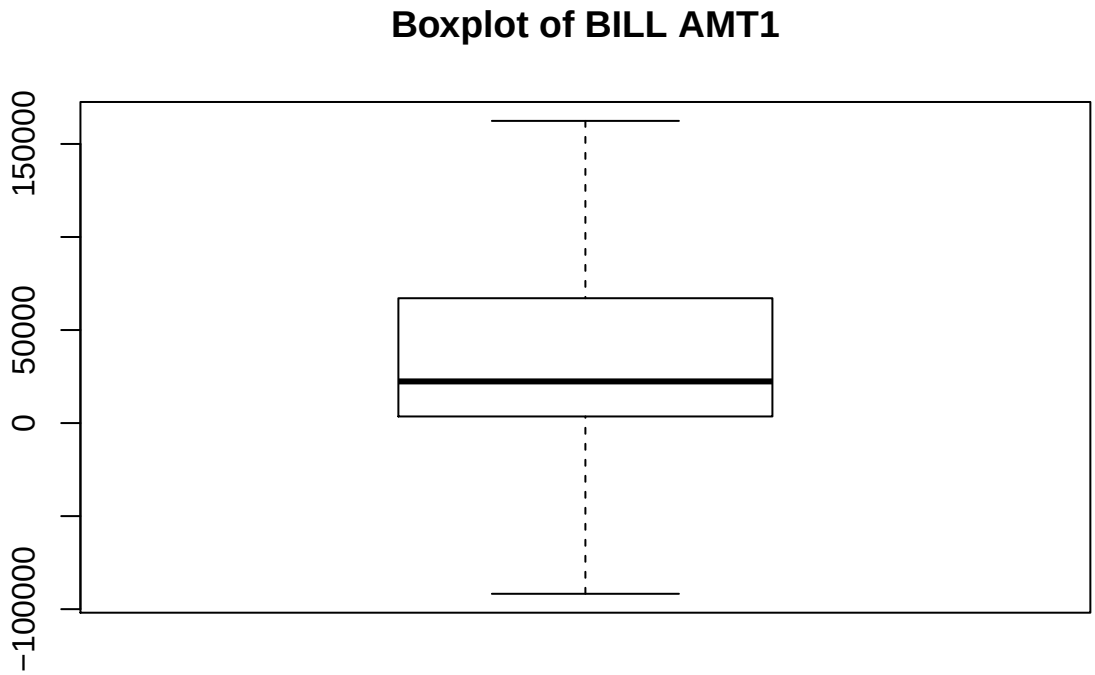
```
## [1] 162389.4
```

```
## [1] -91739.38
```

Findings

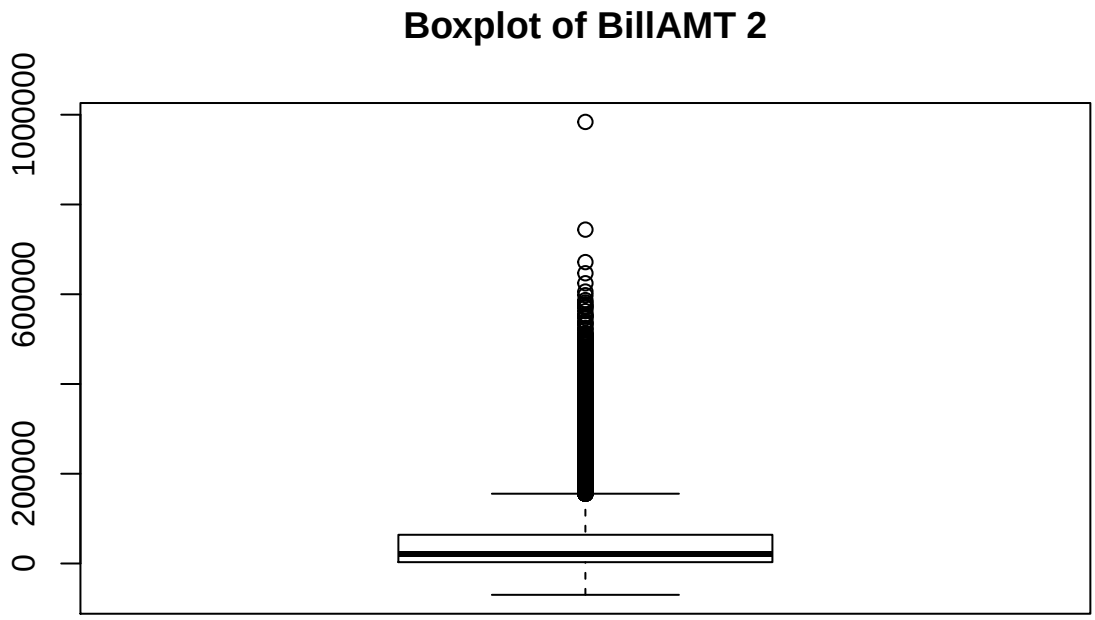
- Any observation above 162389.4 and below -91739.38 will be assigned the value of 162389.4 and -91739.38 respectively

19.3.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -91739    3559   22382   44293   67091  162389
```

19.4 Boxplot of BillAMT 2




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -69777    2985   21200   49179   64006  983931
```

19.4.1 Treating Outlier for Bill AMT 2 using Winsorizing.

```
## [1] 61021.5
```

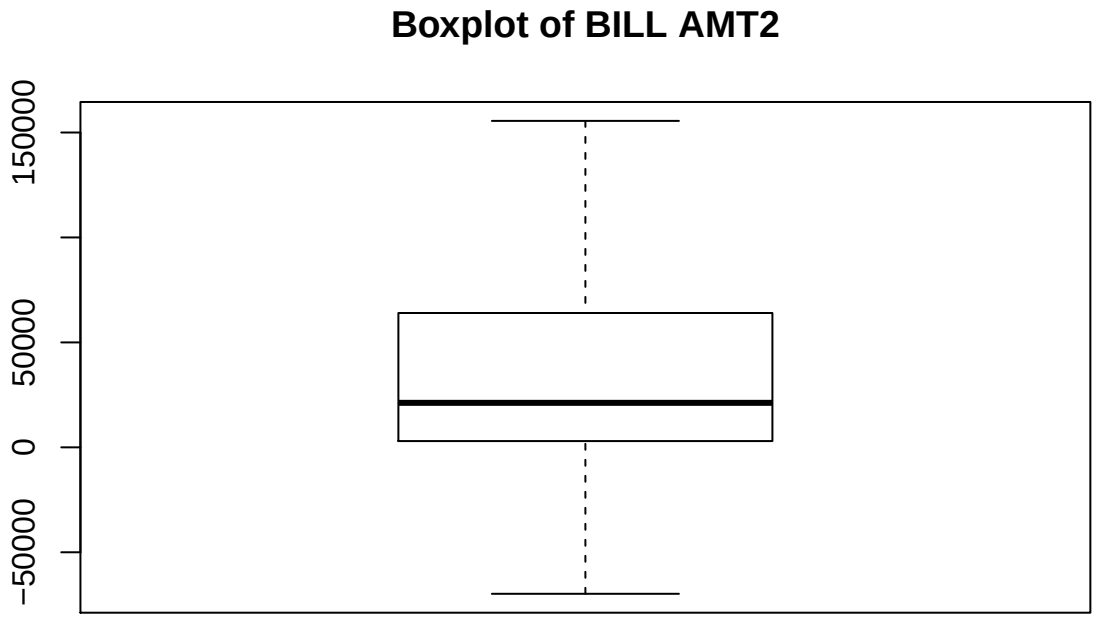
```
## [1] 155538.2
```

```
## [1] -88547.25
```

Findings

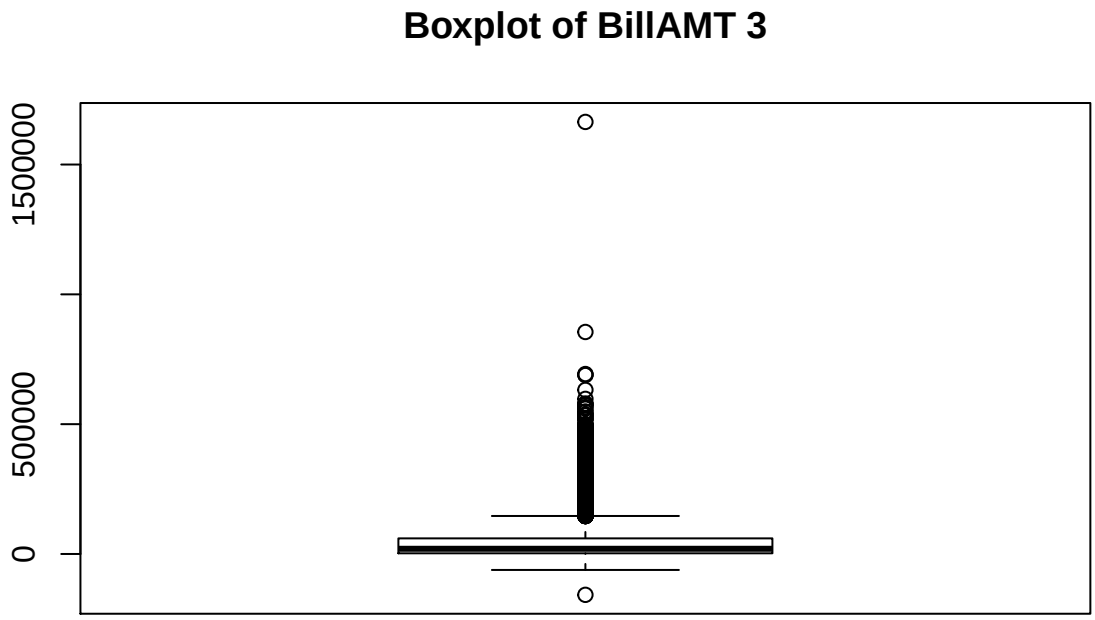
- Any observation above 155538.2 and below -88547.25 will be assigned the value of 155538.2 and -88547.25 respectively

19.4.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -69777    2985   21200   42395   64006  155538
```

19.5 Boxplot of BillAMT 3




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -157264    2666   20089   47013   60165 1664089
```

19.5.1 Treating Outlier for Bill AMT 3 using Winsorizing.

```
## [1] 57498.5
```

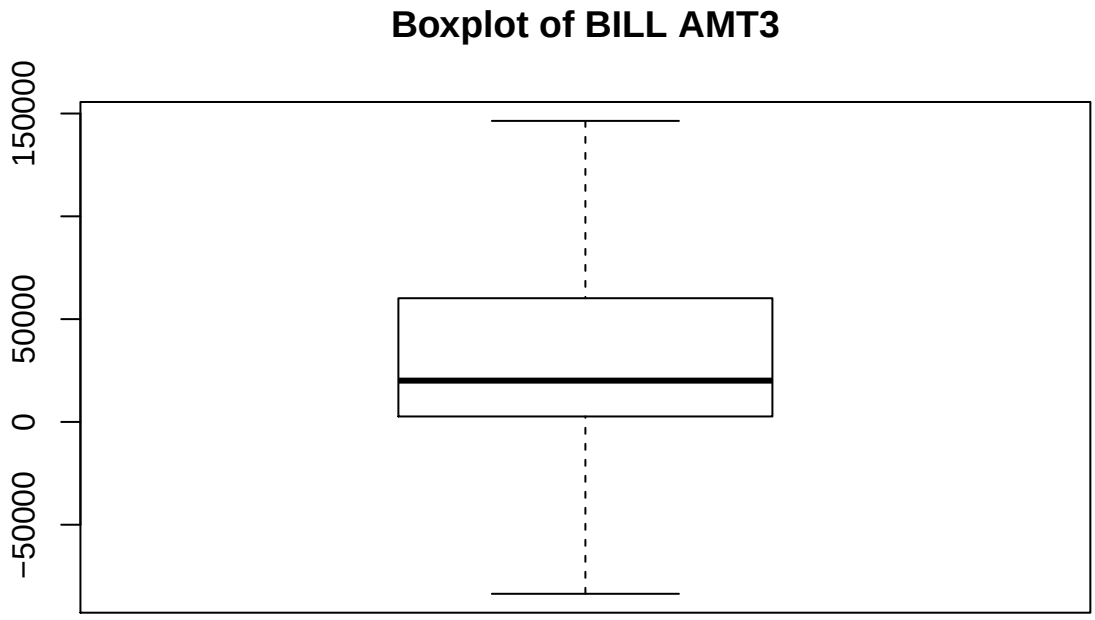
```
## [1] 146412.8
```

```
## [1] -83581.75
```

Findings

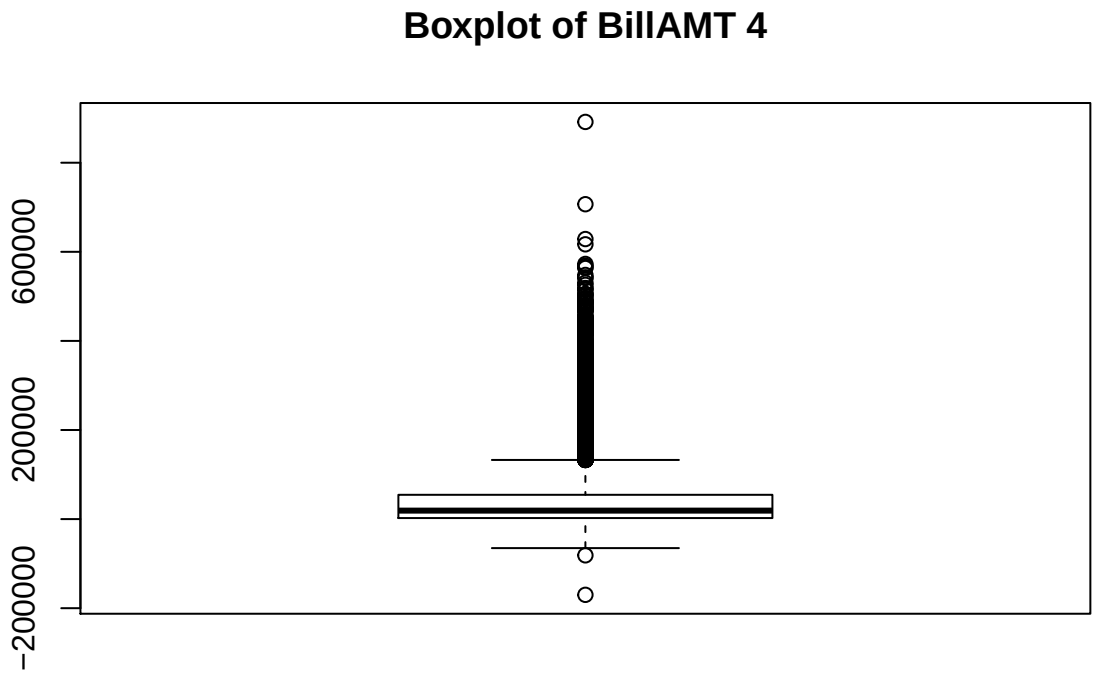
- Any observation above 146412.8 and below -83581.75 will be assigned the value of 146412.8 and -83581.75 respectively

19.5.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -83582    2666   20089   40126   60165  146413
```

19.6 Boxplot of BillAMT 4




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -170000    2327   19052   43263   54506  891586
```

19.6.1 Treating Outlier for Bill AMT 4 using Winsorizing.

```
## [1] 52179.25
```

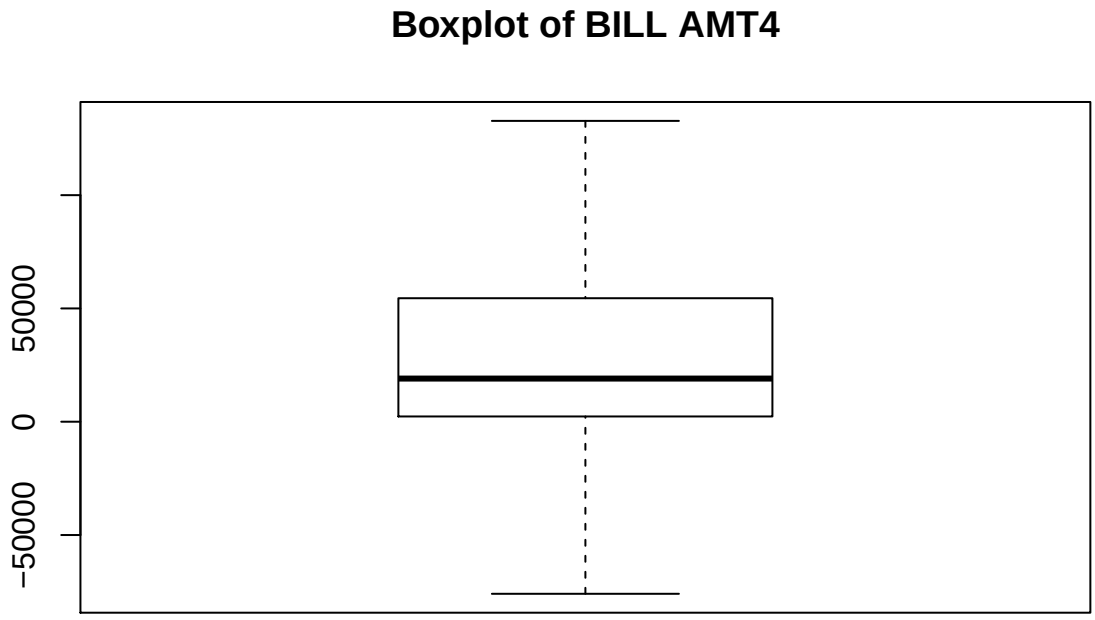
```
## [1] 132774.9
```

```
## [1] -75941.88
```

Findings

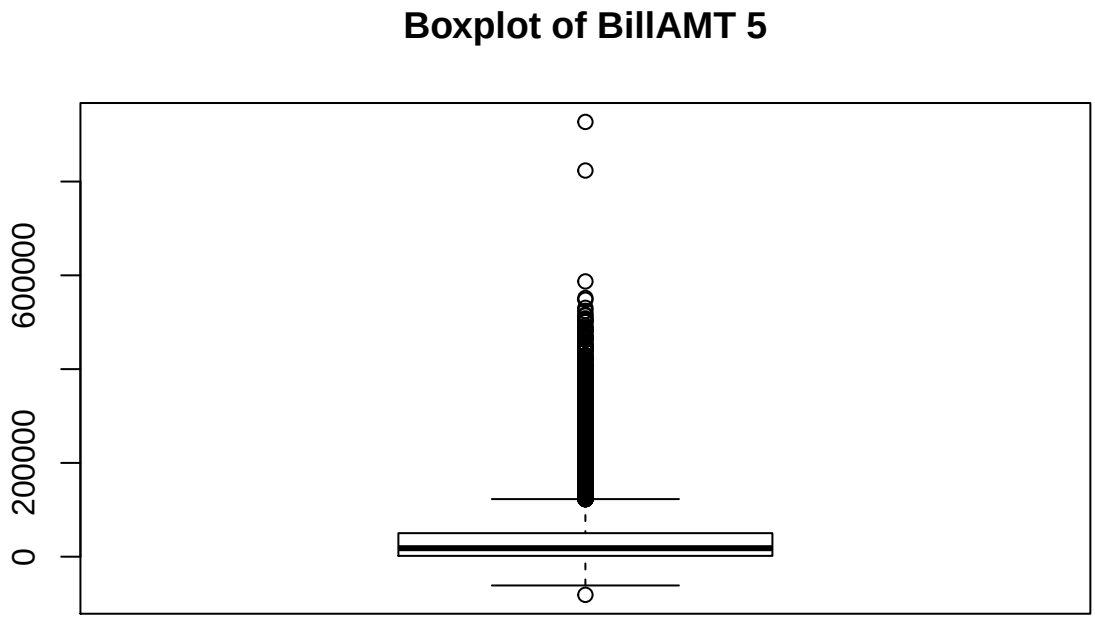
- Any observation above 132774.9 and below -75941.88 will be assigned the value of 132774.9 and -75941.88 respectively

19.6.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -75942    2327   19052   36551   54506  132775
```

19.7 Boxplot of BillAMT 5




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -81334    1763   18105   40311   50191  927171
```

19.7.1 Treating Outlier for Bill AMT 5 using Winsorizing.

```
## [1] 48427.5
```

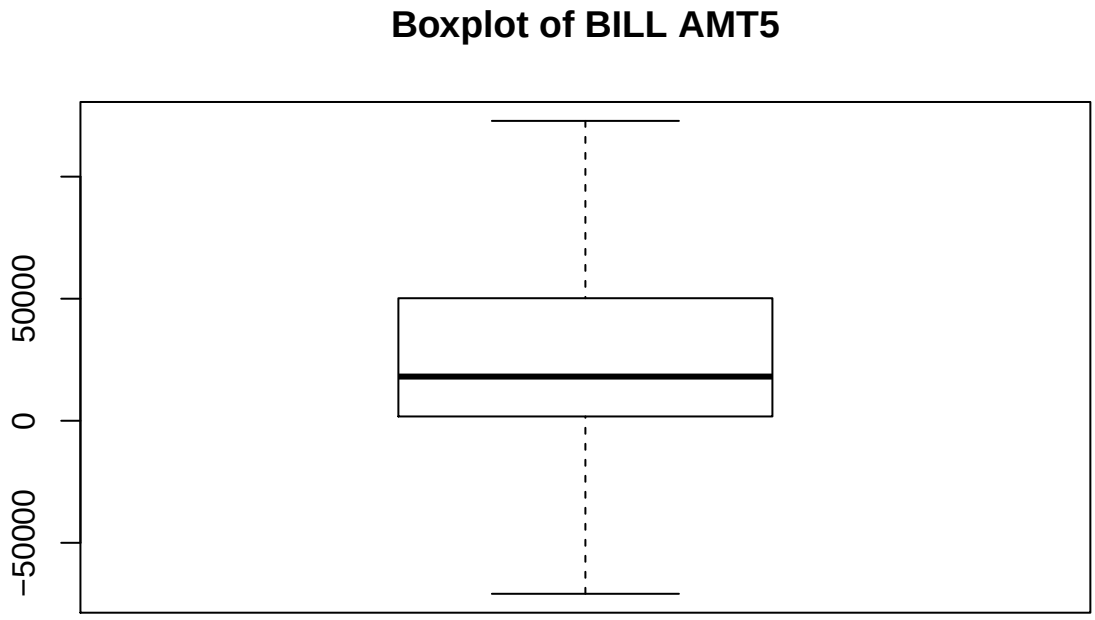
```
## [1] 122832.2
```

```
## [1] -70878.25
```

Findings

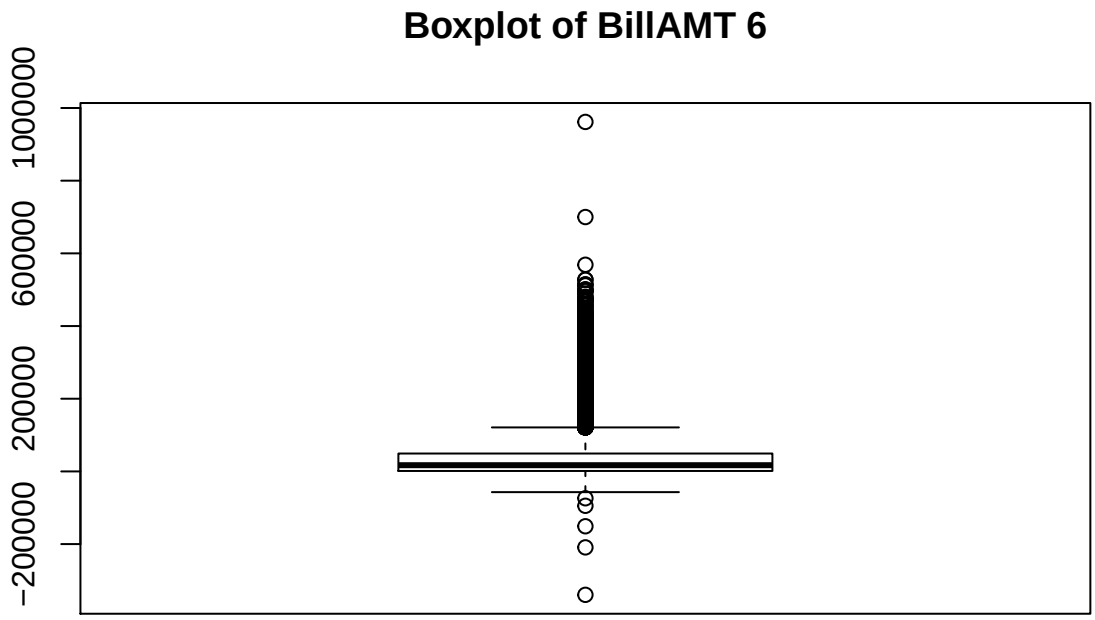
- Any observation above 122832.2 and below -70878.25 will be assigned the value of 122832.2 and -70878.25 respectively

19.7.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -70878    1763   18105   33754   50191  122832
```

19.8 Boxplot of BillAMT 6




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -339603    1256   17071   38872   49198  961664
```

19.8.1 Treating Outlier for Bill AMT 6 using Winsorizing.

```
## [1] 47942.25
```

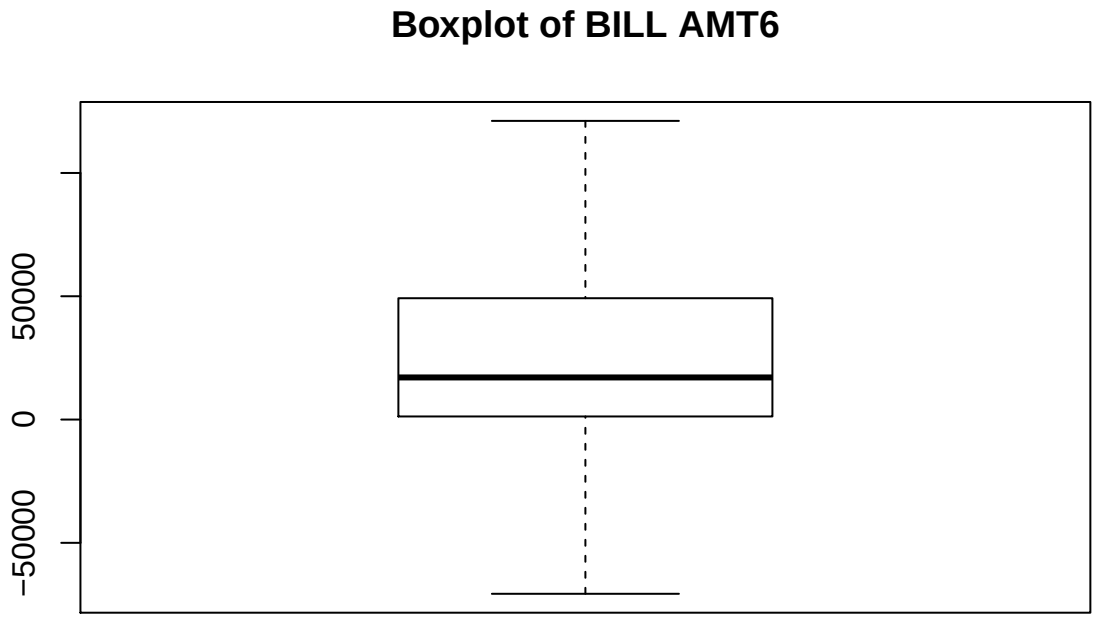
```
## [1] 121111.4
```

```
## [1] -70657.38
```

Findings

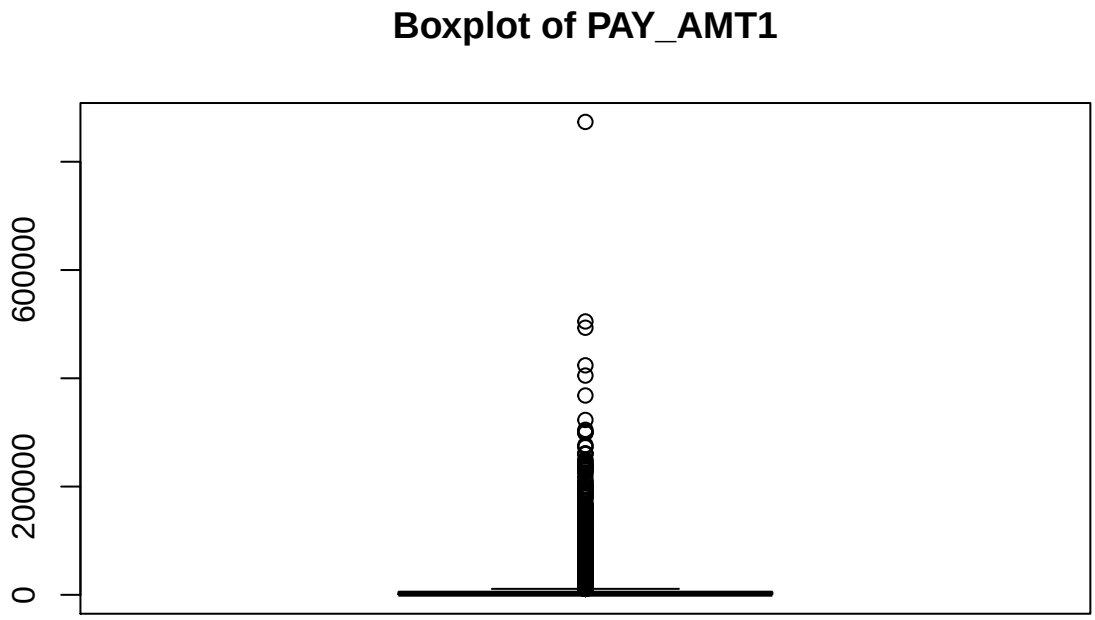
- Any observation above 121111.4 and below -70657.38 will be assigned the value of 121111.4 and -70657.38 respectively

19.8.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -70657    1256   17071   32595   49198  121111
```

19.9 Boxplot of PAY_AMT1




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1000    2100    5664    5006  873552
```

19.9.1 Treating Outlier for PAY_AMT1 using Winsorizing.

```
## [1] 4006
```

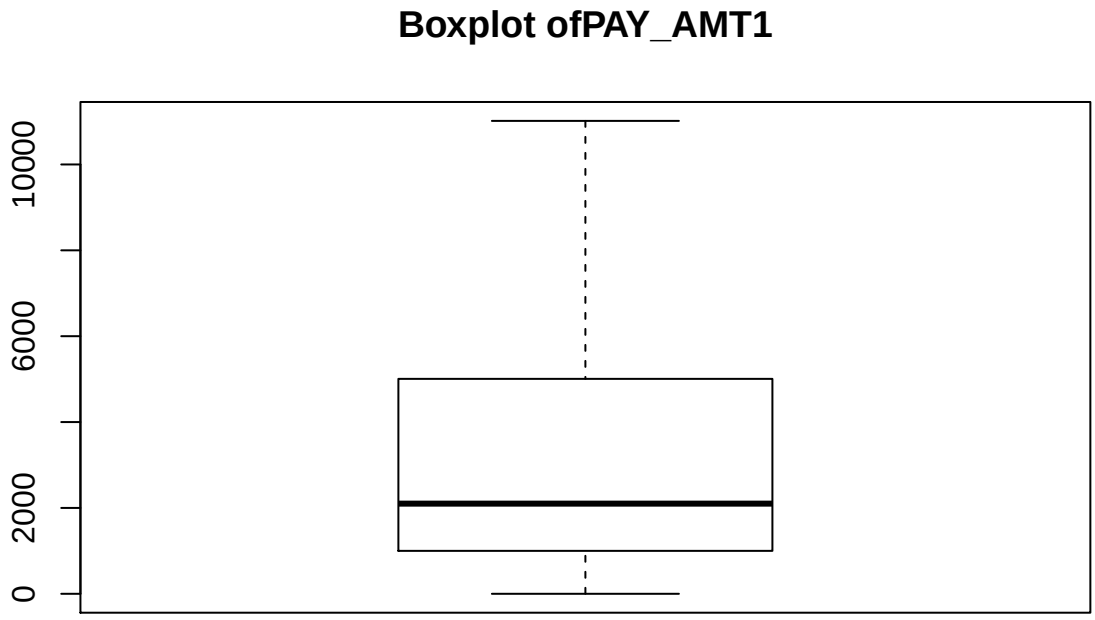
```
## [1] 11015
```

```
## [1] -5009
```

Findings

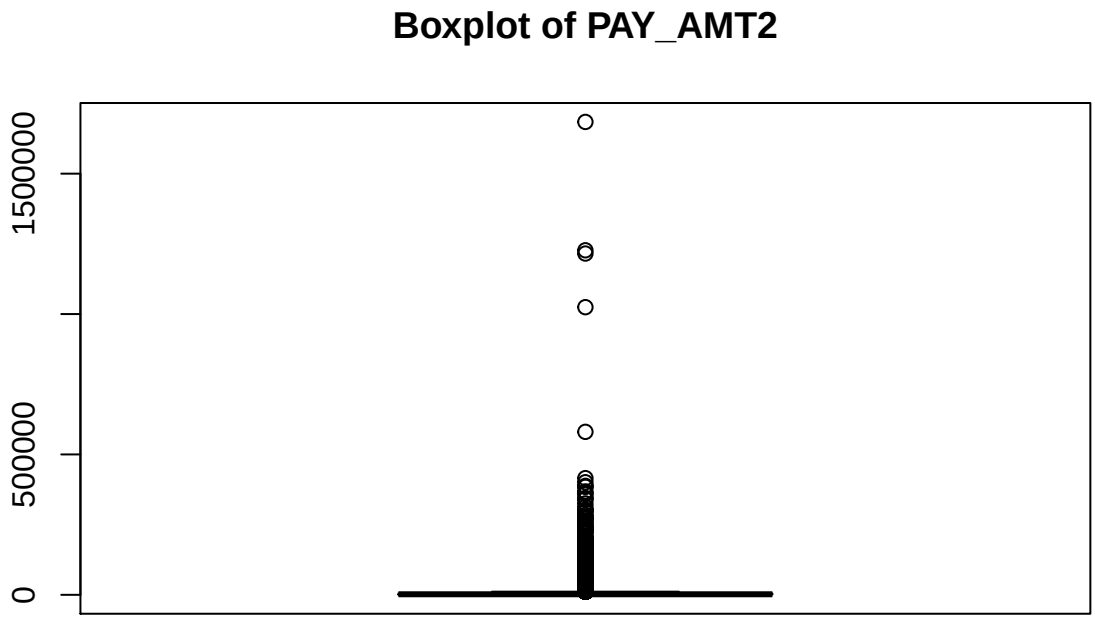
- Any observation above 11015 and below -5009 will be assigned the value of 11015 and -5009

19.9.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1000    2100    3497    5006   11015
```

19.10 Boxplot of PAY_AMT2




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     833    2009    5921    5000 1684259
```

19.10.1 Treating Outlier for PAY_AMT2 using Winsorizing.

```
## [1] 4167
```

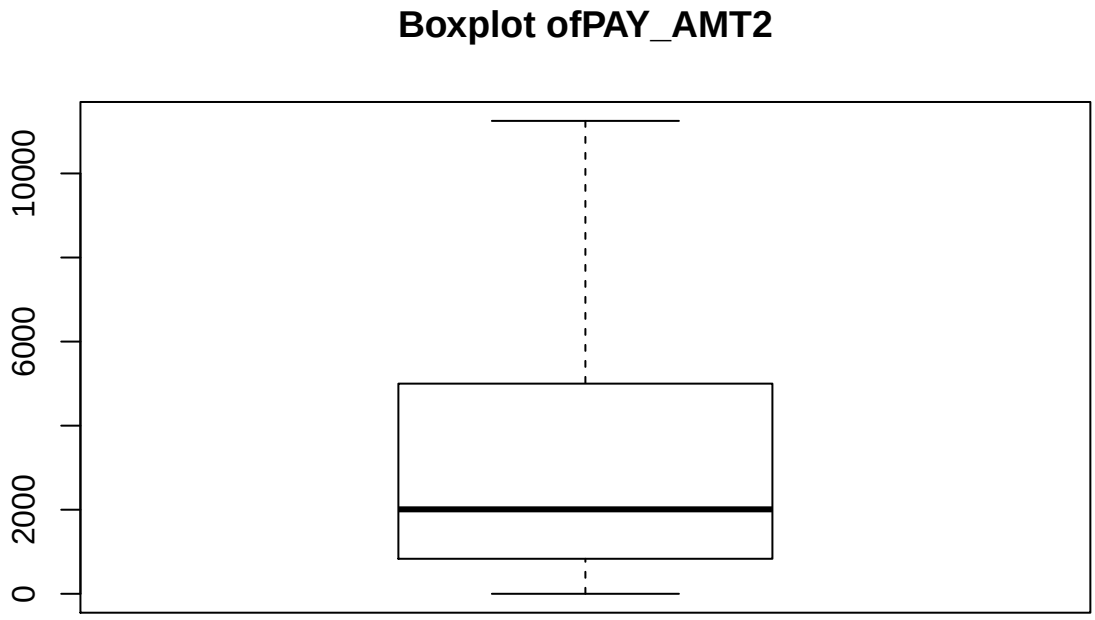
```
## [1] 11250.5
```

```
## [1] -5417.5
```

Findings

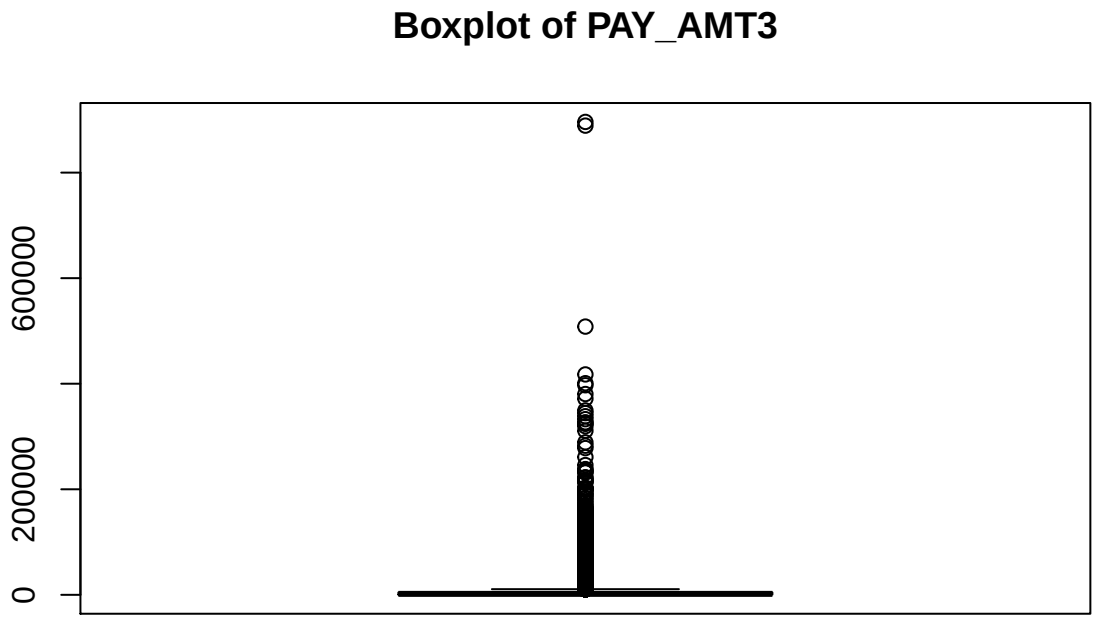
- Any observation above 11250.5 and below -5417.5 will be assigned the value of 11250.5 and -5417.5

19.10.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     833    2009    3422    5000   11250
```

19.11 Boxplot of PAY_AMT3




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     390    1800    5226    4505  896040
```

19.11.1 Treating Outlier for PAY_AMT3 using Winsorizing.

```
## [1] 4115
```

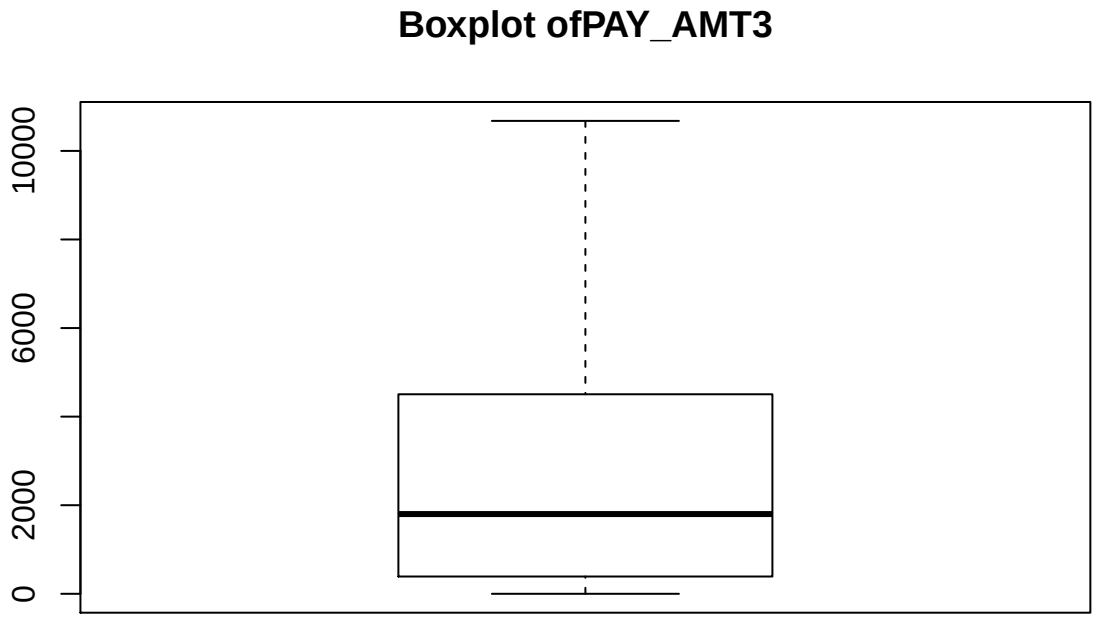
```
## [1] 10677.5
```

```
## [1] -5782.5
```

Findings

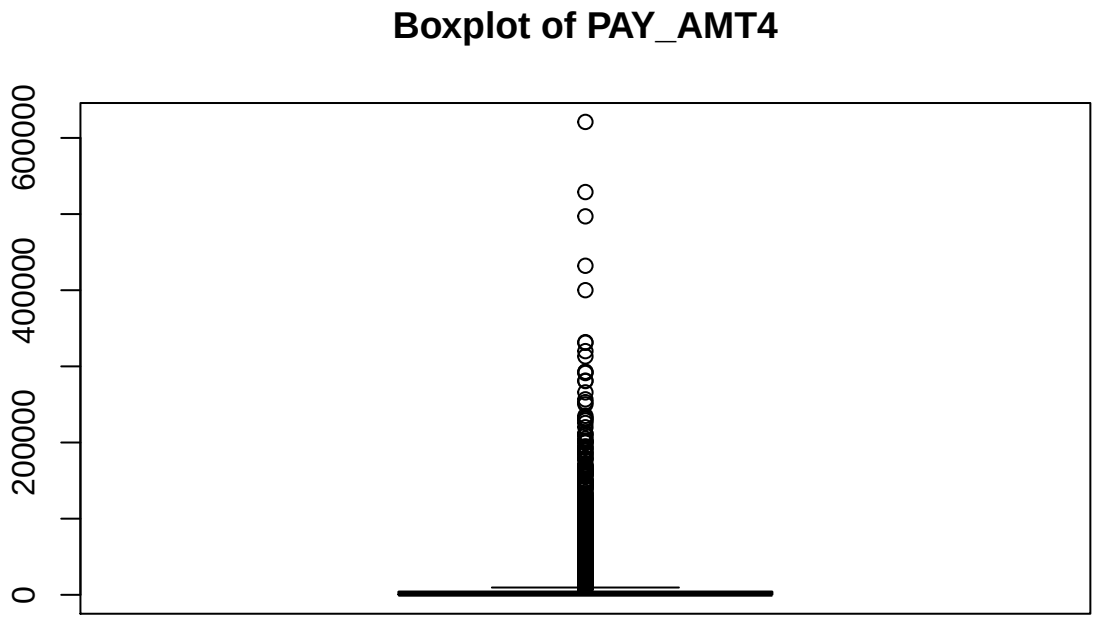
- Any observation above 10677.5 and below -5782.5 will be assigned the value of 10677.5 and -5782.5

19.11.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     390    1800    3036    4505   10678
```

19.12 Boxplot of PAY_AMT4




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     296    1500    4826    4013  621000
```

19.12.1 Treating Outlier for PAY_AMT4 using Winsorizing.

```
## [1] 3717.25
```

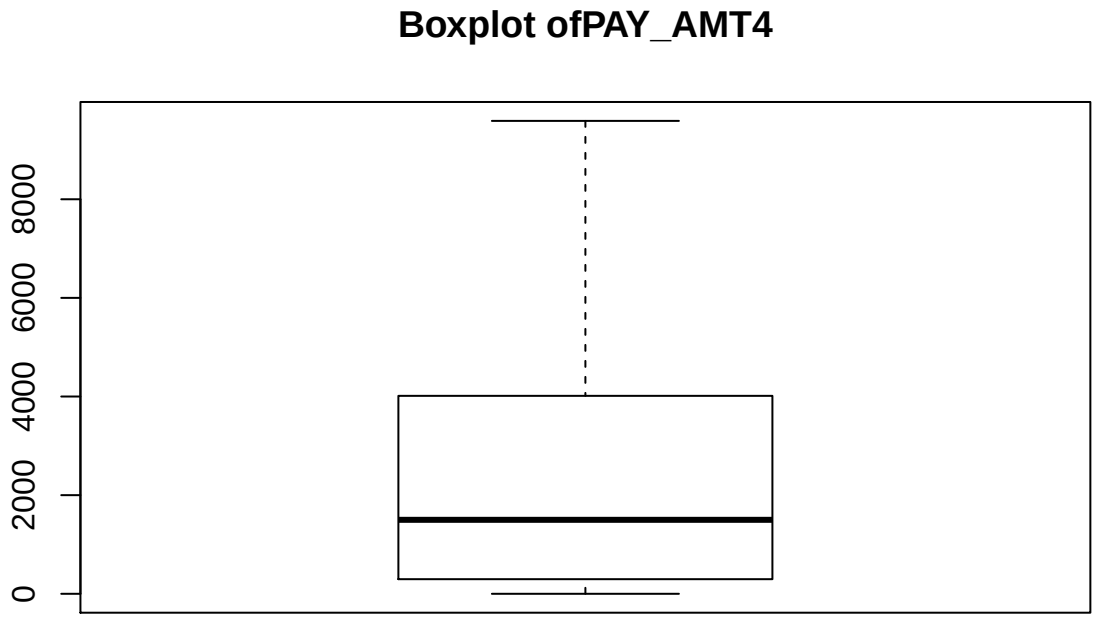
```
## [1] 9588.875
```

```
## [1] -5279.875
```

Findings

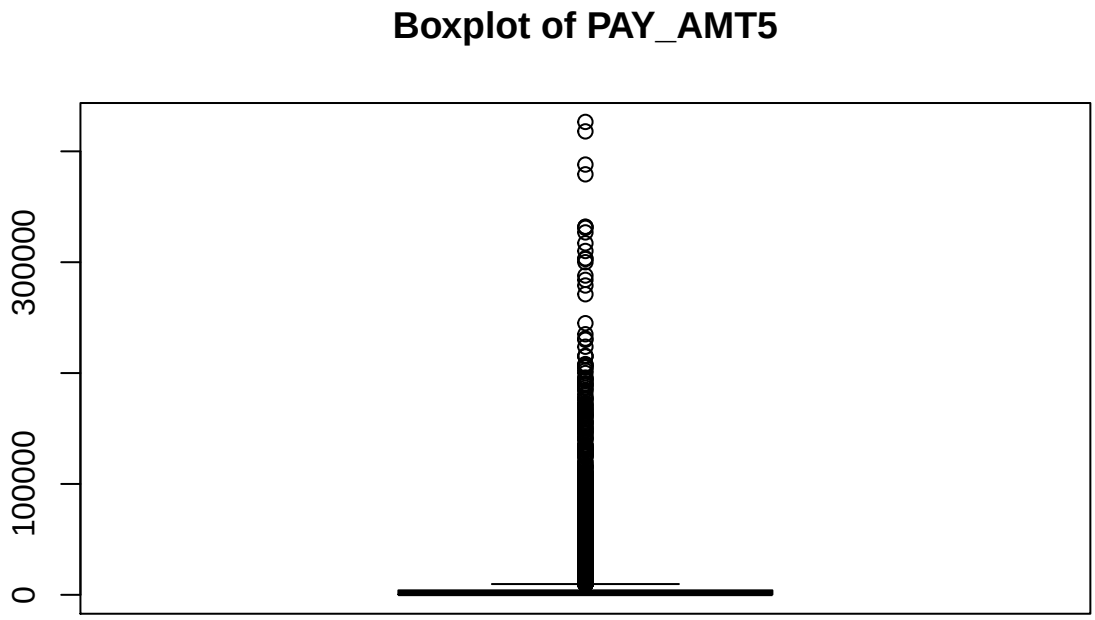
- Any observation above 9588.875 and below -5279.875 will be assigned the value of 9588.875 and -5279.875

19.12.2 Let's plot and check if Outliers have reduced




```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0     296    1500    2718    4013    9589
```

19.13 Boxplot of PAY_AMT5




```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##      0.0    252.5   1500.0   4799.4   4031.5 426529.0
```

19.13.1 Treating Outlier for PAY_AMT5 using Winsorizing.

```
## [1] 3779
```

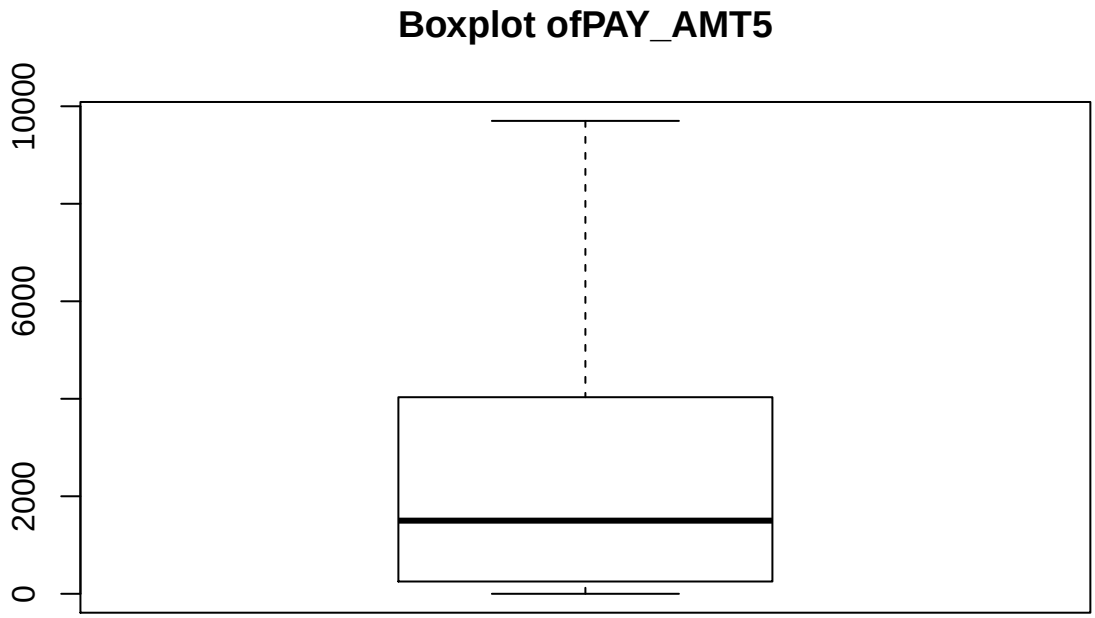
```
## [1] 9700
```

```
## [1] -5416
```

Findings

- Any observation above 9700 and below -5416 will be assigned the value of 9700 and -5416

19.13.2 Let's plot and check if Outliers have reduced

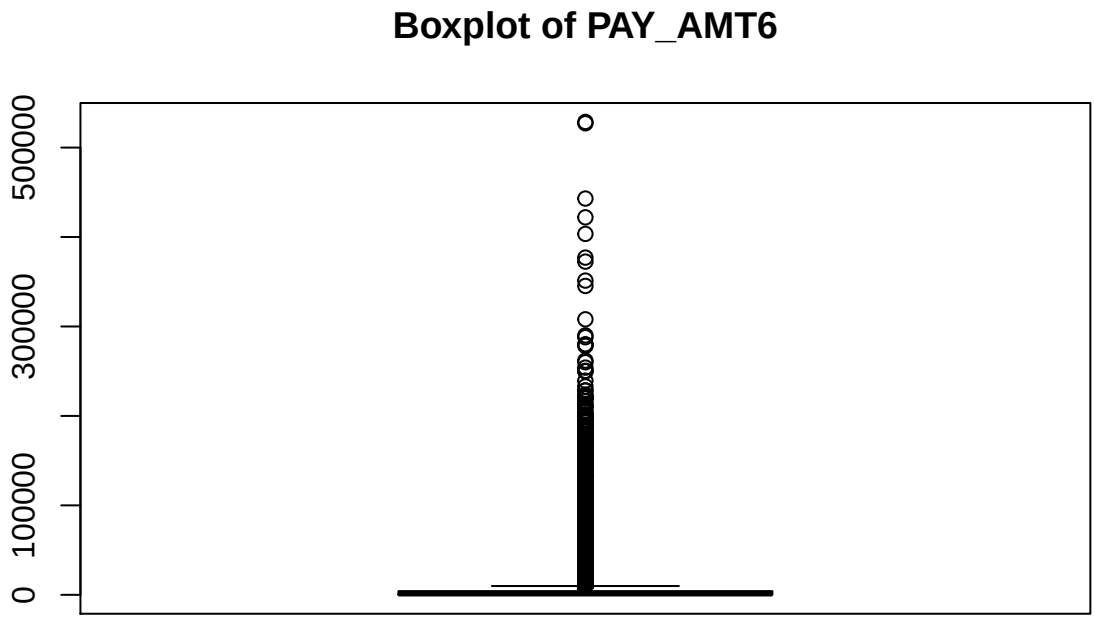



```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   252.5  1500.0  2731.5  4031.5  9700.0
```

Findings

Created dataset called 'outliersdata' with outliers treated.

19.14 Boxplot of PAY_AMT6




```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##      0.0    117.8   1500.0   5215.5   4000.0 528666.0
```

19.14.1 Treating Outlier for PAY_AMT6 using Winsorizing.

```
## [1] 3882.25
```

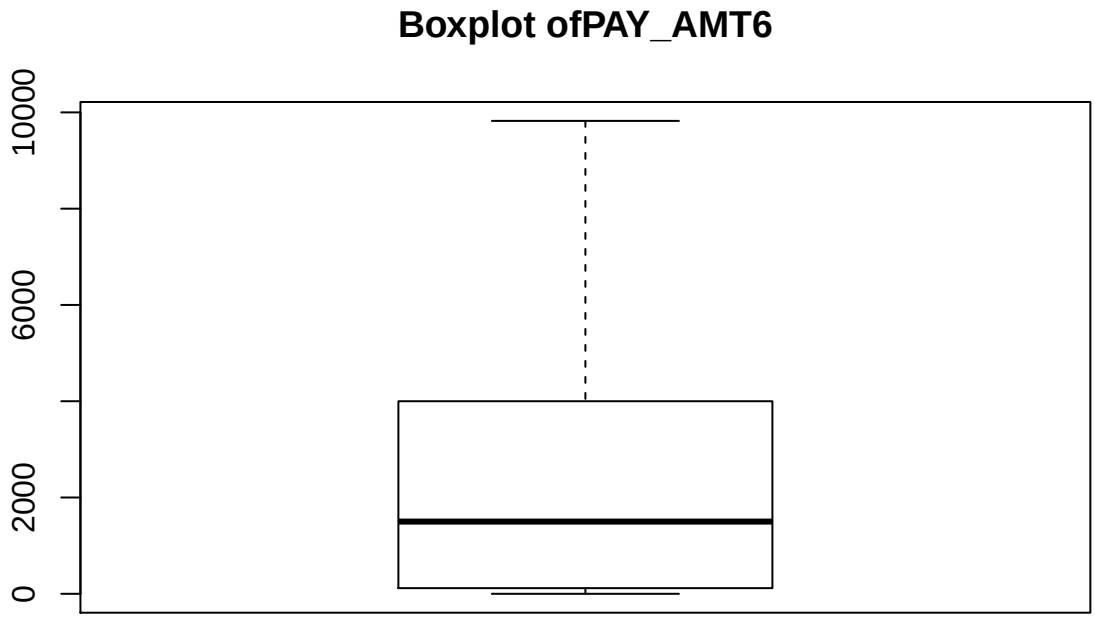
```
## [1] 9823.375
```

```
## [1] -5705.575
```

Findings

- Any observation above 9823.37 and below -5705.575 will be assigned the value of 9823.37 and -5705.575

19.14.2 Let's plot and check if Outliers have reduced



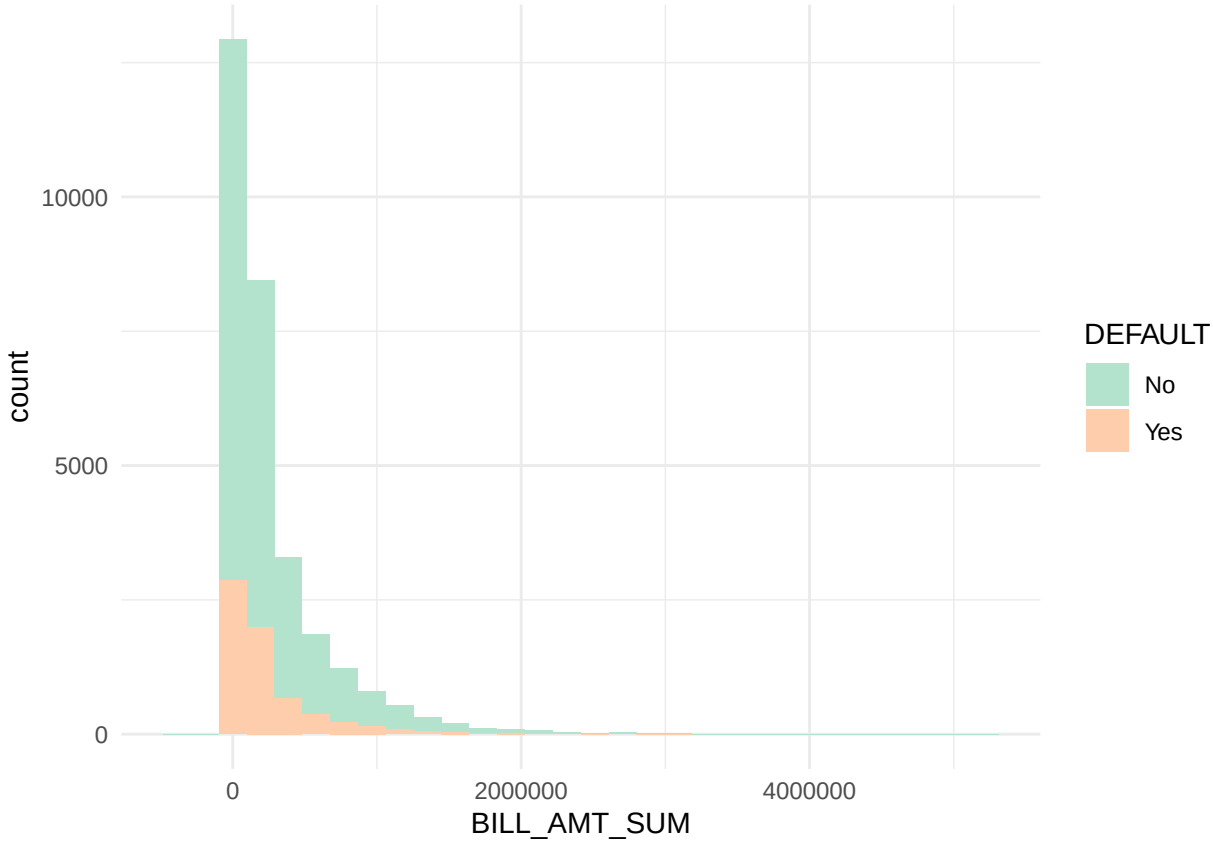

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   117.8  1500.0  2714.4  4000.0  9823.4
```

20 Feature Engineering. Creating New Variables

Findings

- Created TWO NEW Variables 'BILL_AMT_SUM' and 'PAY_AMT_SUM'

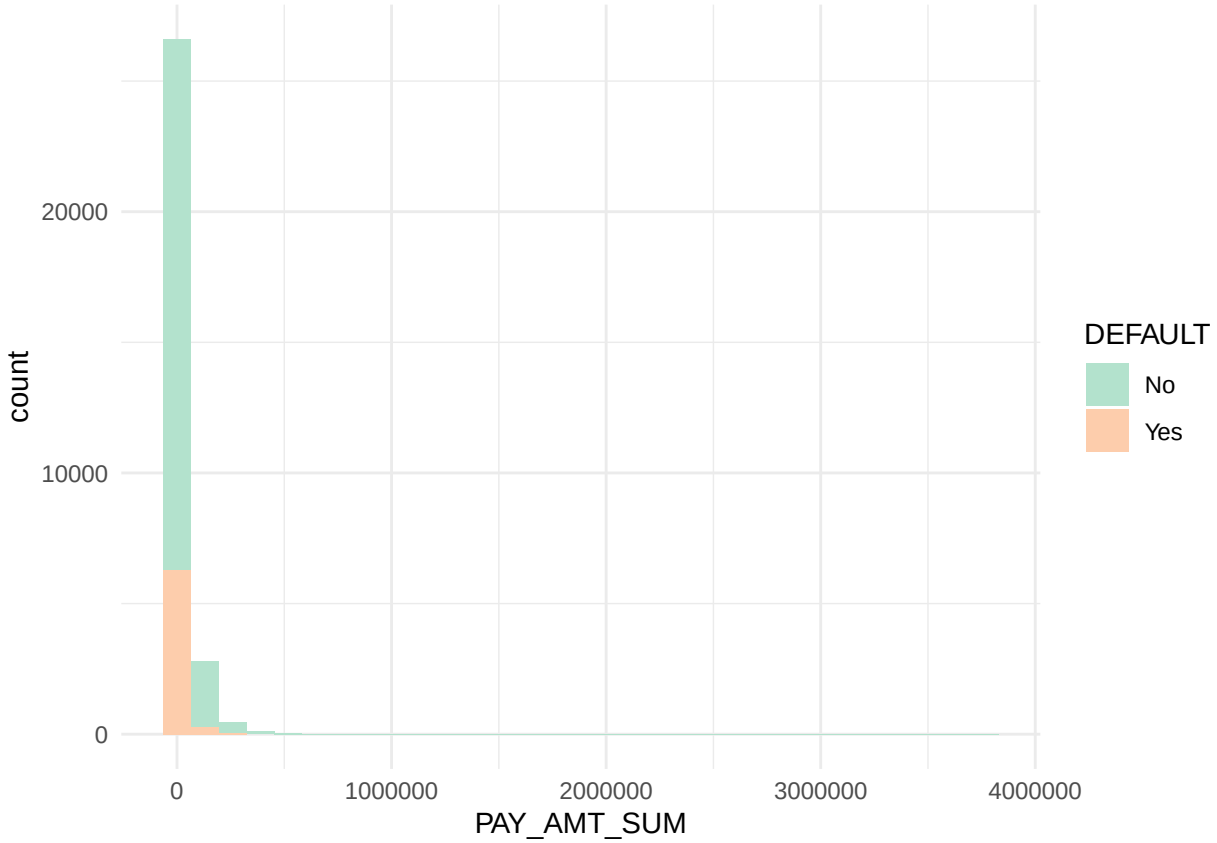
20.1 EDA on the new Variable 'BILL_AMT_SUM'



Findings

- This is intersting! The more the Sum of Billing Amount lesser are the chances of Default

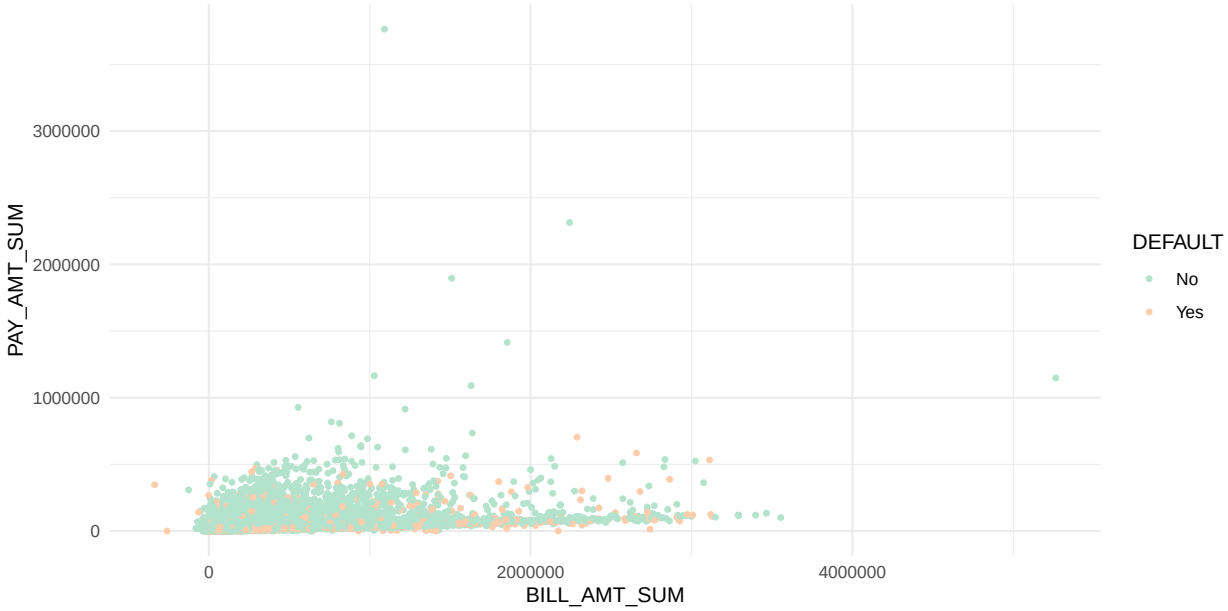
20.2 EDA on the new Variable 'PAY_AMT_SUM'



Findings

- More the 'PAY_AMT_SUM' the lesser the Default Rate

20.3 Let's see how they interact with each other



Findings

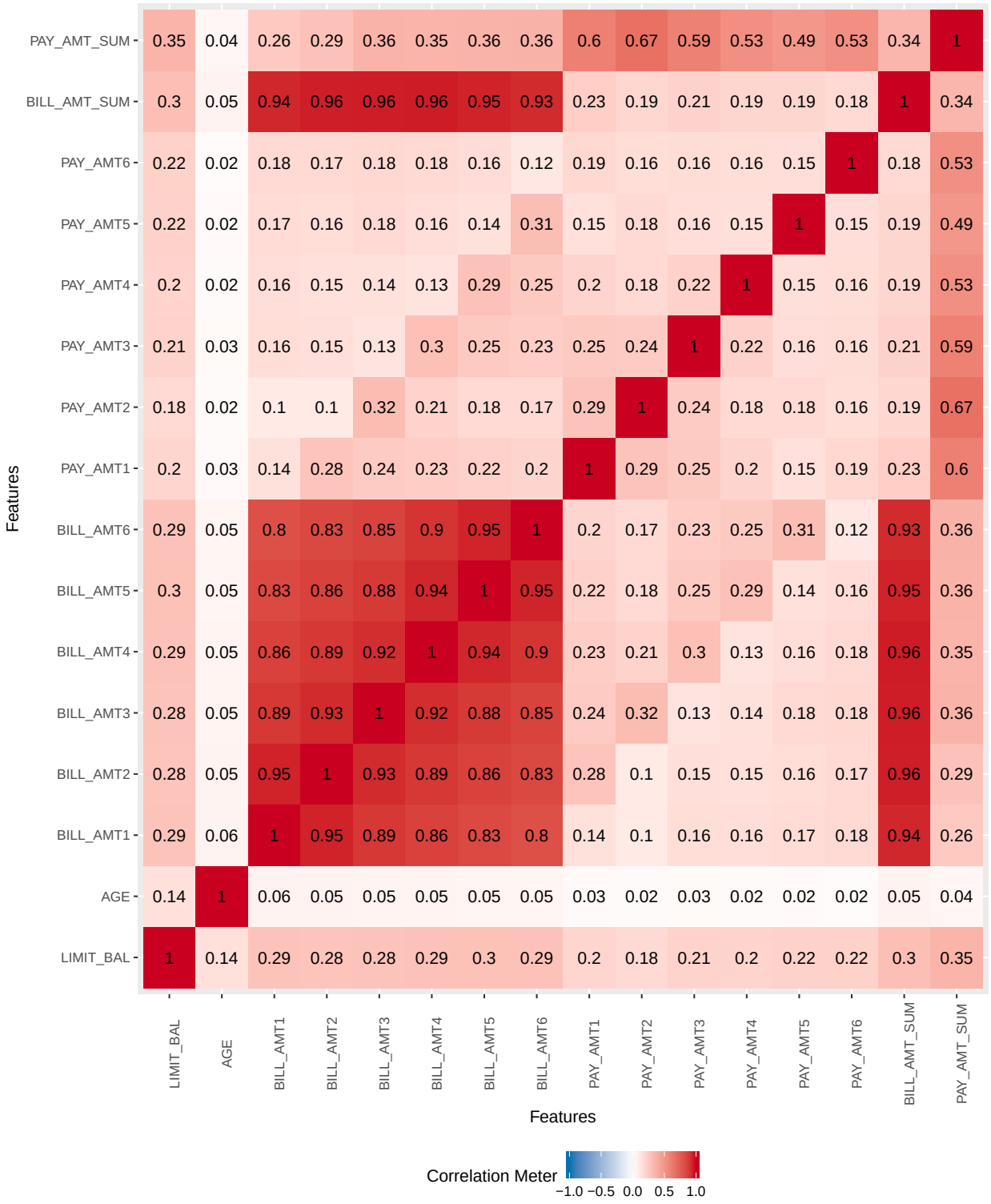
- There definitely is some pattern. We find Defaulters more towards bottom of X axis.

21 Identification of Important Variables as per Submission 1 EDA

As per EDA 1. The imporatant variables are Gender, Age, Education, Limit Balance and the two new variabes 'BILL_AMT_SUM' and 'PAY_AMT_SUM' we created.

22 Check Correlation

let's plot correlation plot between only numberical variables



Findings

- Will replace (Pay_AMT 1 to 6) and (BILL_AMT 1 to 6) with To ('BILL_AMT_SUM' and 'PAY_AMT_SUM')

23 Multicollinearity

Multicollinearity will be dealth by replacing (Pay_AMT 1 to 6) and (BILL_AMT 1 to 6) with To ('BILL_AMT_SUM' and 'PAY_AMT_SUM')

24 Assess if SMOTE is required

```
##
##    No   Yes
## 77.88 22.12
```

```
##
##    No   Yes
## 23364  6636
```

Findings

- Machine learning classifiers such as Random Forests fail to cope with imbalanced training datasets as they are sensitive to the proportions of the different classes. As a consequence, these algorithms tend to favor the class with the largest proportion of observations (known as majority class), which may lead to misleading accuracies.
- We'll create a balanced dataset. However, we'll retain the actual dataset aswell(unbalanced)

24.1 now using SMOTE to create a more "balanced problem"

```
##
##    No   Yes
## 26544 33180
```

```
##
##       No      Yes
## 44.44444 55.55556
```

24.2 Let's create another SMOTE dataset with Outlier treated

24.3 now using SMOTE to create a more "balanced problem"

```
##
##    No   Yes
## 26544 33180
```

```
##
##       No      Yes
## 44.44444 55.55556
```

24.4 Create Standarized dataset. Data standardization is about making sure that data is internally consistent; that is, each data type has the same content and format

24.4.1 Standarized Dataet

```
standardizeddaaset$PAY_6<- as.numeric(as.character(standardizeddaaset$PAY_6))
```

```
##   LIMIT_BAL.V1      SEX                  EDUCATION        MARRIAGE
##  Min.   :-1.2   Female:18112   Graduate.School:10585   Married:13659
##  1st Qu.:-0.9   Male  :11888   High.School    : 4917   Other  :  377
```

```
##  Median :-0.2                 Other          :  123   Single :15964
##  Mean   : 0.0                 University     :14030
##  3rd Qu.: 0.6                 Unkown         :  345
##  Max.   : 6.4
##      AGE.V1        PAY_1.V1       PAY_2.V1       PAY_3.V1       PAY_4.V1
##  Min.   :-1.6   Min.   :-1.8   Min.   :-1.6   Min.   :-1.5   Min.   :-1.5
##  1st Qu.:-0.8   1st Qu.:-0.9   1st Qu.:-0.7   1st Qu.:-0.7   1st Qu.:-0.7
##  Median :-0.2   Median : 0.0   Median : 0.1   Median : 0.1   Median : 0.2
##  Mean   : 0.0   Mean   : 0.0   Mean   : 0.0   Mean   : 0.0   Mean   : 0.0
##  3rd Qu.: 0.6   3rd Qu.: 0.0   3rd Qu.: 0.1   3rd Qu.: 0.1   3rd Qu.: 0.2
##  Max.   : 4.7   Max.   : 7.1   Max.   : 6.8   Max.   : 6.8   Max.   : 7.0
##     PAY_5.V1       PAY_6.V1     BILL_AMT1.V1   BILL_AMT2.V1   BILL_AMT3.V1
##  Min.   :-1.5   Min.   :-1.5   Min.   :-2.9   Min.   :-1.7   Min.   :-2.9
##  1st Qu.:-0.6   1st Qu.:-0.6   1st Qu.:-0.6   1st Qu.:-0.6   1st Qu.:-0.6
##  Median : 0.2   Median : 0.3   Median :-0.4   Median :-0.4   Median :-0.4
##  Mean   : 0.0   Mean   : 0.0   Mean   : 0.0   Mean   : 0.0   Mean   : 0.0
##  3rd Qu.: 0.2   3rd Qu.: 0.3   3rd Qu.: 0.2   3rd Qu.: 0.2   3rd Qu.: 0.2
##  Max.   : 7.3   Max.   : 7.2   Max.   :12.4   Max.   :13.1   Max.   :23.3
##   BILL_AMT4.V1   BILL_AMT5.V1   BILL_AMT6.V1  PAY_AMT1.V1  PAY_AMT2.V1
##  Min.   :-3.3   Min.   :-2.0   Min.   :-6.4   Min.   : 0   Min.   : 0
##  1st Qu.:-0.6   1st Qu.:-0.6   1st Qu.:-0.6   1st Qu.: 0   1st Qu.: 0
##  Median :-0.4   Median :-0.4   Median :-0.4   Median : 0   Median : 0
##  Mean   : 0.0   Mean   : 0.0   Mean   : 0.0   Mean   : 0   Mean   : 0
##  3rd Qu.: 0.2   3rd Qu.: 0.2   3rd Qu.: 0.2   3rd Qu.: 0   3rd Qu.: 0
##  Max.   :13.2   Max.   :14.6   Max.   :15.5   Max.   :52   Max.   :73
##  PAY_AMT3.V1  PAY_AMT4.V1   PAY_AMT5.V1    PAY_AMT6.V1   DEFAULT
##  Min.   : 0   Min.   : 0   Min.   :-0.3   Min.   :-0.3   No :23364
##  1st Qu.: 0   1st Qu.: 0   1st Qu.:-0.3   1st Qu.:-0.3   Yes: 6636
##  Median : 0   Median : 0   Median :-0.2   Median :-0.2
##  Mean   : 0   Mean   : 0   Mean   : 0.0   Mean   : 0.0
##  3rd Qu.: 0   3rd Qu.: 0   3rd Qu.:-0.1   3rd Qu.:-0.1
##  Max.   :51   Max.   :39   Max.   :27.6   Max.   :29.4
##  BILL_AMT_SUM.V1 PAY_AMT_SUM.V1
##  Min.   :-1.6    Min.   :-1
##  1st Qu.:-0.6    1st Qu.: 0
##  Median :-0.4    Median : 0
##  Mean   : 0.0    Mean   : 0
##  3rd Qu.: 0.2    3rd Qu.: 0
##  Max.   :13.2    Max.   :61
```

25 List out different models/algorithms

It is a Classification problem. We would like to predict DEFAULTERS. Models used are :

1) **Logistic Regression**

Definition: Logistic regression is a machine learning algorithm for classification. In this algorithm, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function

Advantages: Logistic regression is designed for this purpose (classification), and is most useful for understanding the influence of several independent variables on a single outcome variable

2) **Naïve Bayes**

Definition: Naive Bayes algorithm based on Bayes' theorem with the assumption of independence between every pair of features. Naive Bayes classifiers work well in many real-world situations such as document

classification and spam filtering.

Advantages: This algorithm requires a small amount of training data to estimate the necessary parameters. Naive Bayes classifiers are extremely fast compared to more sophisticated methods.

3) **K-Nearest Neighbours**

Definition: Neighbours based classification is a type of lazy learning as it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the k nearest neighbours of each point.

Advantages: This algorithm is simple to implement, robust to noisy training data, and effective if training data is large.

4) **Decision Tree**

Definition: Given a data of attributes together with its classes, a decision tree produces a sequence of rules that can be used to classify the data.

Advantages: Decision Tree is simple to understand and visualise, requires little data preparation, and can handle both numerical and categorical data.

5) **Random Forest**

Definition: Random forest classifier is a meta-estimator that fits a number of decision trees on various sub-samples of datasets and uses average to improve the predictive accuracy of the model and controls over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.

Advantages: Reduction in over-fitting and random forest classifier is more accurate than decision trees in most cases.

26 SUMMARY OF NOTES 2

1) Created New Variables BILL_AMT_SUM and PAY_AMT_SUM
2) EDA of NEW VARIABLES
3) Outliers Treated
4) Multiple Datasets created. Actual, outlier treated, SMOTE OUTLIER, Standardization SMOTE
5) Clearly mentioned models and their advantages to be used to predict DEFAULTERS
6) DATASET WILL BE SPLIT IN 70 - 30 Ratio for all models.

26.1 Datasets to be used

1) Feature Engineered but not outlier treated ('BILL_AMT_SUM' and 'PAY_AMT_SUM')
2) Feature Engineered and Outlier Treated
3) SMOTE Dataset(Outlier Not treated)
4) SMOTE Outlier Treated
5) Standardized

26.2 Steps

1) Create 5 models on 5 different datasets (Training Set)
2) Tune Model (Adaboost)
3) Use the best one on TEST set and Final model

27 Evaluation Matrix

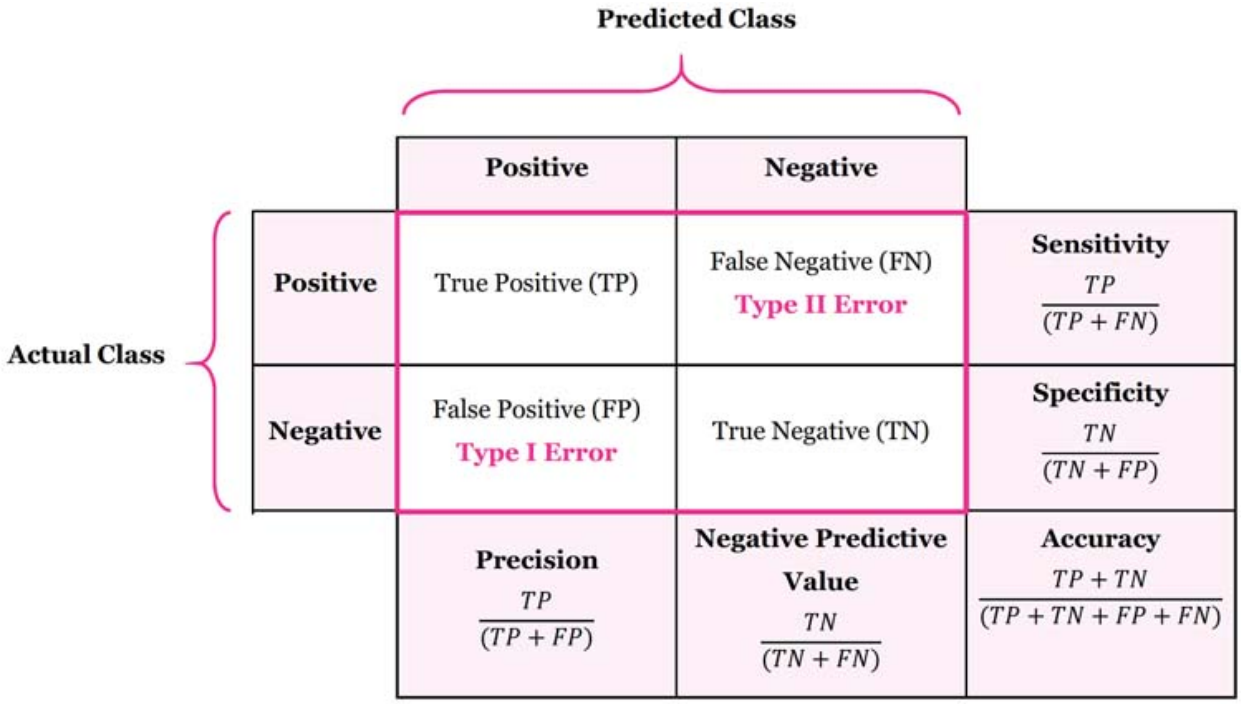


Figure 4: ConMatrix

We'll look into Sensitivity and Accuracy more closely as we are more bothered of DEFAULTERS than NON DEFAULTERS as they can bring the whole banking system down if Credit or Loan is given to wrong people.

27.1 Datasets

27.1.1 Feature Engineered

27.1.2 Feature Engineered with Outlier Treated

```
outliersdata$PAY_2<- as.numeric(as.character(outliersdata$PAY_2))
```

27.1.3 SMOTE

27.1.4 SMOTE Outlier treated

28 Model Evaluation

MODEL WILL BE EVALUATED USING SENSITIVITY AND ACCURACY

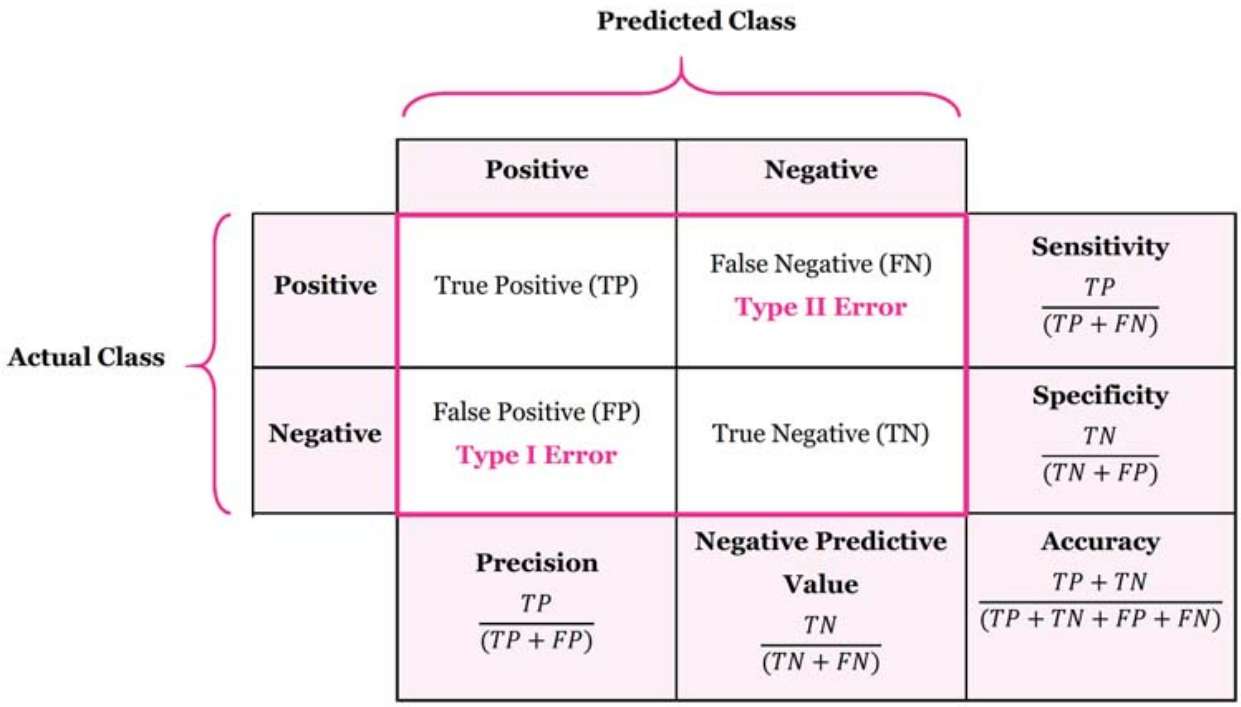


Figure 5: CMATRIX

29 Splitting Datasets 70:30 ratio

29.1 Splittig Feature Engineered but not outlier treated

29.2 Splittig Feature Engineered outlier treated

29.3 Splitting SMOTE without outlier treat

29.4 Splitting OT SMOTE Dataset

29.5 Splitting Standarized Dataset

30 Building Models

30.1 Logistic Regression on Feature Engineered Dataset

30.1.1 Full Model Stepwise

30.1.2 Empty Model

30.1.3 Backward Selection of significant variables

```
## Start:  AIC=19584
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_4 + PAY_5 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## - PAY_4         1    19548 19582
## - PAY_5         1    19548 19582
## - PAY_6         1    19549 19583
## <none>               19548 19584
```

```
## - BILL_AMT_SUM  1    19556 19590
## - PAY_3         1    19556 19590
## - SEX           1    19556 19590
## - MARRIAGE      2    19561 19593
## - AGE           1    19560 19594
## - LIMIT_BAL     1    19561 19595
## - PAY_2         1    19569 19603
## - EDUCATION     4    19589 19617
## - PAY_AMT_SUM   1    19631 19665
## - PAY_1         1    20266 20300
##
## Step:  AIC=19582
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_5 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## - PAY_5         1    19548 19580
## - PAY_6         1    19549 19581
## <none>               19548 19582
## - BILL_AMT_SUM  1    19556 19588
## - SEX           1    19556 19588
## - PAY_3         1    19558 19590
## - MARRIAGE      2    19561 19591
## - AGE           1    19560 19592
## - LIMIT_BAL     1    19561 19593
## - PAY_2         1    19569 19601
## - EDUCATION     4    19589 19615
## - PAY_AMT_SUM   1    19631 19663
## - PAY_1         1    20269 20301
##
## Step:  AIC=19580
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## <none>               19548 19580
## - PAY_6         1    19552 19582
## - BILL_AMT_SUM  1    19556 19586
## - SEX           1    19557 19587
## - MARRIAGE      2    19561 19589
## - PAY_3         1    19560 19590
## - AGE           1    19560 19590
## - LIMIT_BAL     1    19562 19592
## - PAY_2         1    19570 19600
## - EDUCATION     4    19590 19614
## - PAY_AMT_SUM   1    19631 19661
## - PAY_1         1    20273 20303
```

30.1.4 Variable Selection using direction BOTH

```
## Start:  AIC=22195
## DEFAULT ~ 1

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
##                Df Deviance   AIC
## + PAY_1         1    19994 19998
## + PAY_2         1    20735 20739
## + PAY_3         1    21086 21090
## + PAY_4         1    21300 21304
## + PAY_5         1    21398 21402
## + PAY_6         1    21511 21515
## + LIMIT_BAL     1    21680 21684
## + PAY_AMT_SUM   1    21782 21786
## + EDUCATION     4    22059 22069
## + SEX           1    22158 22162
## + MARRIAGE      2    22175 22181
## + AGE           1    22185 22189
## <none>              22193 22195
## + BILL_AMT_SUM  1    22192 22196
##
## Step:  AIC=19998
## DEFAULT ~ PAY_1
##
##                Df Deviance   AIC
## + PAY_AMT_SUM   1    19796 19802
## + LIMIT_BAL     1    19866 19872
## + PAY_2         1    19882 19888
## + PAY_3         1    19889 19895
## + PAY_4         1    19928 19934
## + BILL_AMT_SUM  1    19933 19939
## + PAY_5         1    19938 19944
## + EDUCATION     4    19938 19950
## + PAY_6         1    19948 19954
## + MARRIAGE      2    19972 19980
## + AGE           1    19974 19980
## + SEX           1    19980 19986
## <none>              19994 19998
## - PAY_1         1    22193 22195
##
## Step:  AIC=19802
## DEFAULT ~ PAY_1 + PAY_AMT_SUM

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##                Df Deviance   AIC
## + PAY_2         1    19688 19696
## + PAY_3         1    19693 19701
## + PAY_4         1    19727 19735
## + PAY_5         1    19734 19742
## + PAY_6         1    19741 19749
## + EDUCATION     4    19750 19764
## + LIMIT_BAL     1    19758 19766
## + AGE           1    19773 19781
## + MARRIAGE      2    19774 19784
## + SEX           1    19781 19789
## <none>              19796 19802
## + BILL_AMT_SUM  1    19794 19802
## - PAY_AMT_SUM   1    19994 19998
## - PAY_1         1    21782 21786
```

```
##
## Step:  AIC=19696
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##                Df Deviance   AIC
## + EDUCATION     4    19647 19663
## + AGE           1    19661 19671
## + MARRIAGE      2    19665 19677
## + PAY_3         1    19668 19678
## + LIMIT_BAL     1    19671 19681
## + PAY_4         1    19675 19685
## + PAY_5         1    19675 19685
## + PAY_6         1    19676 19686
## + SEX           1    19677 19687
## + BILL_AMT_SUM  1    19677 19687
## <none>               19688 19696
## - PAY_2         1    19796 19802
## - PAY_AMT_SUM   1    19882 19888
## - PAY_1         1    20474 20480
##
## Step:  AIC=19663
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##                Df Deviance   AIC
## + AGE           1    19622 19640
## + MARRIAGE      2    19624 19644
## + PAY_3         1    19627 19645
## + LIMIT_BAL     1    19629 19647
## + PAY_4         1    19634 19652
## + PAY_5         1    19634 19652
## + SEX           1    19635 19653
## + PAY_6         1    19636 19654
## + BILL_AMT_SUM  1    19637 19655
## <none>               19647 19663
## - EDUCATION     4    19688 19696
## - PAY_2         1    19750 19764
## - PAY_AMT_SUM   1    19833 19847
## - PAY_1         1    20432 20446
##
## Step:  AIC=19640
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

##                Df Deviance   AIC
## + LIMIT_BAL     1    19595 19615
## + PAY_3         1    19601 19621
## + PAY_4         1    19608 19628
## + PAY_5         1    19608 19628
## + BILL_AMT_SUM  1    19610 19630
## + PAY_6         1    19610 19630
## + SEX           1    19613 19633
## + MARRIAGE      2    19613 19635
```

```
## <none>                19622 19640
## - AGE           1     19647 19663
## - EDUCATION     4     19661 19671
## - PAY_2         1     19729 19745
## - PAY_AMT_SUM   1     19813 19829
## - PAY_1         1     20408 20424
##
## Step:  AIC=19615
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL
##
##                Df Deviance   AIC
## + PAY_3         1    19578 19600
## + PAY_4         1    19585 19607
## + PAY_5         1    19586 19608
## + MARRIAGE      2    19584 19608
## + PAY_6         1    19587 19609
## + SEX           1    19588 19610
## + BILL_AMT_SUM  1    19591 19613
## <none>                19595 19615
## - LIMIT_BAL     1    19622 19640
## - AGE           1    19629 19647
## - EDUCATION     4    19638 19650
## - PAY_2         1    19682 19700
## - PAY_AMT_SUM   1    19708 19726
## - PAY_1         1    20362 20380
##
## Step:  AIC=19600
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3
##
##                Df Deviance   AIC
## + MARRIAGE      2    19567 19593
## + SEX           1    19572 19596
## + BILL_AMT_SUM  1    19572 19596
## + PAY_6         1    19576 19600
## + PAY_5         1    19576 19600
## <none>                19578 19600
## + PAY_4         1    19577 19601
## - PAY_3         1    19595 19615
## - PAY_2         1    19600 19620
## - LIMIT_BAL     1    19601 19621
## - AGE           1    19613 19633
## - EDUCATION     4    19621 19635
## - PAY_AMT_SUM   1    19694 19714
## - PAY_1         1    20318 20338
##
## Step:  AIC=19593
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3 + MARRIAGE
##
##                Df Deviance   AIC
## + SEX           1    19558 19586
## + BILL_AMT_SUM  1    19561 19589
## + PAY_6         1    19564 19592
```

```
## + PAY_5         1    19564 19592
## <none>              19567 19593
## + PAY_4         1    19566 19594
## - MARRIAGE      2    19578 19600
## - AGE           1    19581 19605
## - PAY_3         1    19584 19608
## - PAY_2         1    19588 19612
## - LIMIT_BAL     1    19592 19616
## - EDUCATION     4    19611 19629
## - PAY_AMT_SUM   1    19680 19704
## - PAY_1         1    20305 20329
##
## Step:  AIC=19586
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3 + MARRIAGE + SEX
##
##                Df Deviance   AIC
## + BILL_AMT_SUM  1    19552 19582
## + PAY_6         1    19556 19586
## + PAY_5         1    19556 19586
## <none>              19558 19586
## + PAY_4         1    19557 19587
## - SEX           1    19567 19593
## - MARRIAGE      2    19572 19596
## - AGE           1    19570 19596
## - PAY_3         1    19575 19601
## - PAY_2         1    19579 19605
## - LIMIT_BAL     1    19583 19609
## - EDUCATION     4    19602 19622
## - PAY_AMT_SUM   1    19673 19699
## - PAY_1         1    20296 20322
##
## Step:  AIC=19582
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM
##
##                Df Deviance   AIC
## + PAY_6         1    19548 19580
## + PAY_5         1    19549 19581
## <none>              19552 19582
## + PAY_4         1    19551 19583
## - BILL_AMT_SUM  1    19558 19586
## - SEX           1    19561 19589
## - MARRIAGE      2    19565 19591
## - AGE           1    19564 19592
## - LIMIT_BAL     1    19567 19595
## - PAY_3         1    19571 19599
## - PAY_2         1    19575 19603
## - EDUCATION     4    19594 19616
## - PAY_AMT_SUM   1    19634 19662
## - PAY_1         1    20296 20324
##
## Step:  AIC=19580
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
```

```
##     PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM + PAY_6
##
##                Df Deviance   AIC
## <none>               19548 19580
## + PAY_5         1    19548 19582
## - PAY_6         1    19552 19582
## + PAY_4         1    19548 19582
## - BILL_AMT_SUM  1    19556 19586
## - SEX           1    19557 19587
## - MARRIAGE      2    19561 19589
## - PAY_3         1    19560 19590
## - AGE           1    19560 19590
## - LIMIT_BAL     1    19562 19592
## - PAY_2         1    19570 19600
## - EDUCATION     4    19590 19614
## - PAY_AMT_SUM   1    19631 19661
## - PAY_1         1    20273 20303
##
## Call:
## glm(formula = DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION +
##     AGE + LIMIT_BAL + PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM +
##     PAY_6, family = "binomial", data = FeatEngineered.train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -3.141  -0.700  -0.553  -0.291   3.125
##
## Coefficients:
##                          Estimate    Std. Error z value             Pr(>|z|)
## (Intercept)          -1.2527736331  0.0984365227  -12.73 < 0.0000000000000002
## PAY_1                 0.5696370461  0.0210865776   27.01 < 0.0000000000000002
## PAY_AMT_SUM          -0.0000054294  0.0000006723   -8.08  0.0000000000000067
## PAY_2                 0.1087484591  0.0235254795    4.62  0.00000378992846073
## EDUCATIONHigh.School -0.0816676226  0.0565045846   -1.45              0.14837
## EDUCATIONOther       -1.6768306367  0.5957176440   -2.81              0.00488
## EDUCATIONUniversity  -0.0847284679  0.0423836732   -2.00              0.04560
## EDUCATIONUnkown      -1.2117213497  0.2676251866   -4.53  0.00000596346444830
## AGE                   0.0077263028  0.0022229247    3.48              0.00051
## LIMIT_BAL            -0.0000006791  0.0000001874   -3.62              0.00029
## PAY_3                 0.0797098534  0.0231846564    3.44              0.00059
## MARRIAGEOther        -0.1603253576  0.1554208779   -1.03              0.30228
## MARRIAGESingle       -0.1471543022  0.0414430513   -3.55              0.00038
## SEXMale               0.1073053223  0.0366781367    2.93              0.00344
## BILL_AMT_SUM         -0.0000001708  0.0000000625   -2.73              0.00625
## PAY_6                 0.0373819542  0.0194584199    1.92              0.05472
##
## (Intercept)          ***
## PAY_1                ***
## PAY_AMT_SUM          ***
## PAY_2                ***
## EDUCATIONHigh.School
## EDUCATIONOther       **
## EDUCATIONUniversity  *
```

```
## EDUCATIONUnkown      ***
## AGE                  ***
## LIMIT_BAL            ***
## PAY_3                ***
## MARRIAGEOther
## MARRIAGESingle       ***
## SEXMale              **
## BILL_AMT_SUM         **
## PAY_6                .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 22193  on 20999  degrees of freedom
## Residual deviance: 19548  on 20984  degrees of freedom
## AIC: 19580
##
## Number of Fisher Scoring iterations: 5
```

30.1.5 Predict Test Set

30.1.6 Converting Prob to Classes

30.1.7 Confusion Matrix

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6852  157
##        Yes 1515  476
##
##                Accuracy : 0.814
##                  95% CI : (0.806, 0.822)
##     No Information Rate : 0.93
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.287
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.7520
##             Specificity : 0.8189
##          Pos Pred Value : 0.2391
##          Neg Pred Value : 0.9776
##               Precision : 0.2391
##                  Recall : 0.7520
##                      F1 : 0.3628
##              Prevalence : 0.0703
##          Detection Rate : 0.0529
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7855
##
##        'Positive' Class : Yes
```

##

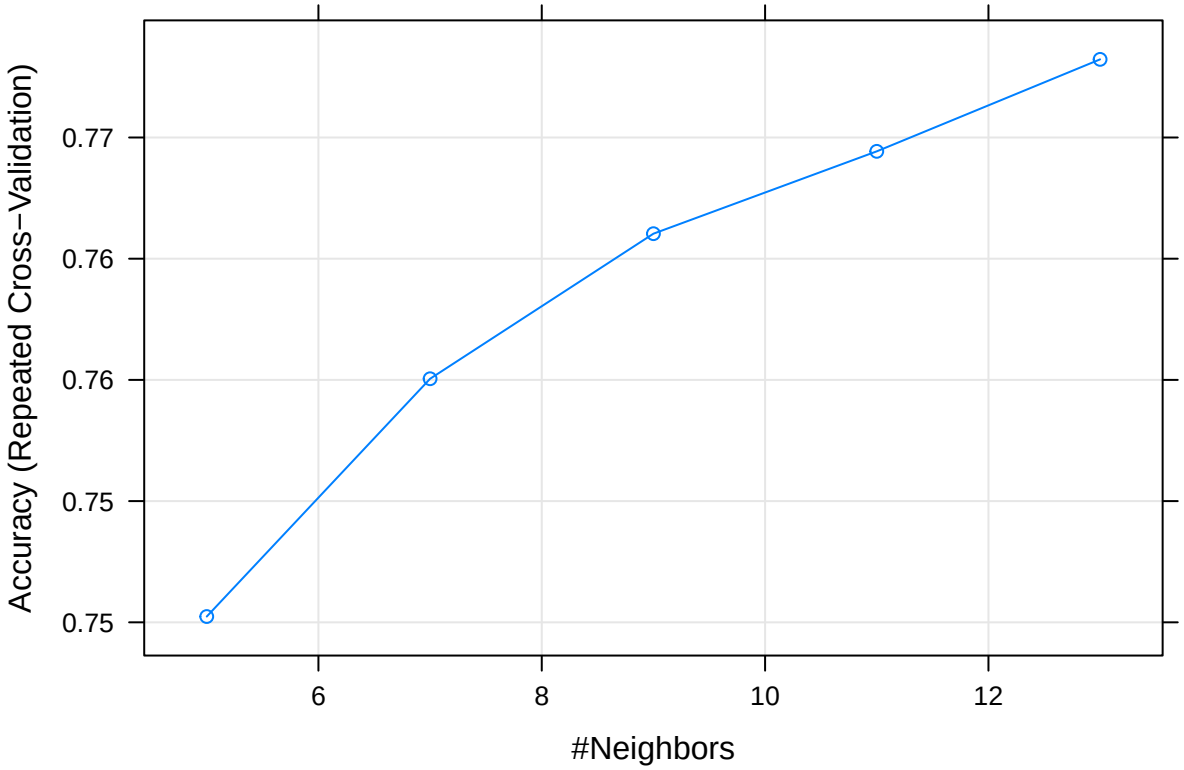
Findings

- Accuracy is good. However, Sensitivity can be improved

30.2 KNN Model

```
## k-Nearest Neighbors
##
## 21000 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 18901, 18900, 18901, 18900, 18899, 18899, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy  Kappa
##    5  0.75      0.083
##    7  0.76      0.079
##    9  0.76      0.073
##   11  0.76      0.068
##   13  0.77      0.066
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 13.
```

30.2.1 #Plotting Number of Neighbours Vs accuracy (based on repeated cross validation)



Findings

- 33 is the best K value

30.2.2 Let's predict for Test Data

30.2.3 Confusion Matrix

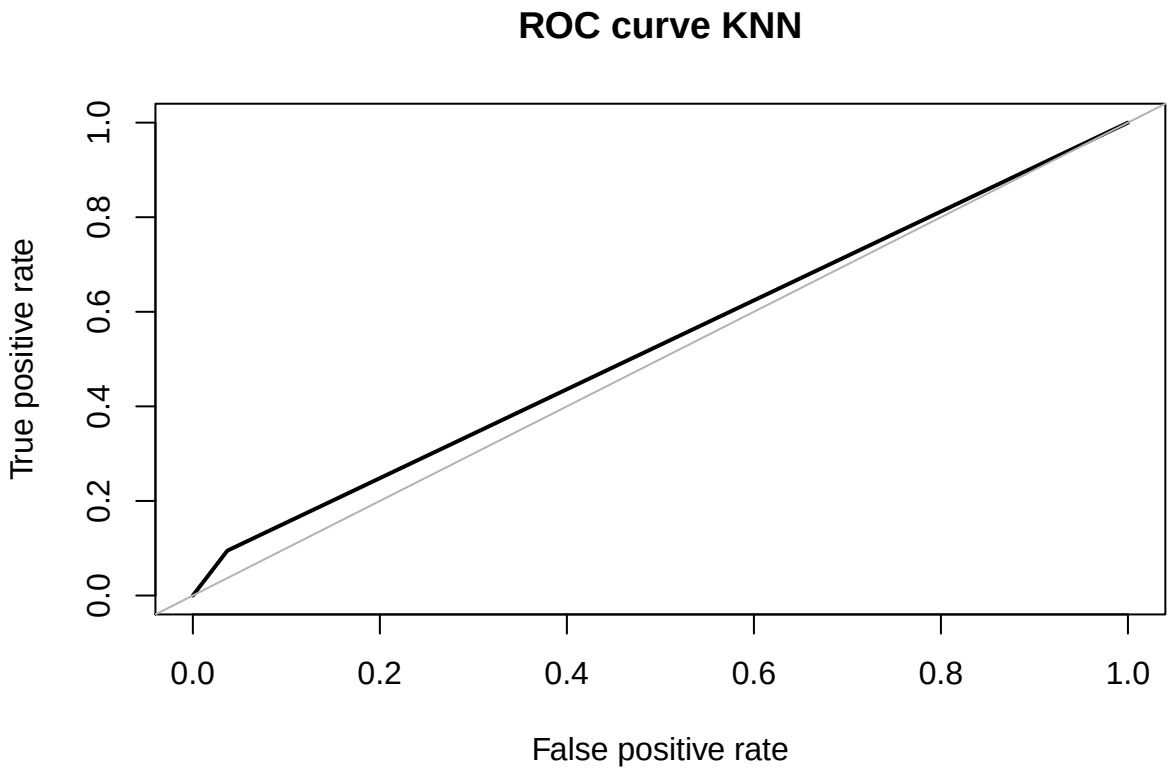
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6750  259
##        Yes 1802  189
##
##                Accuracy : 0.771
##                  95% CI : (0.762, 0.78)
##     No Information Rate : 0.95
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.08
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.4219
##             Specificity : 0.7893
```

```
##          Pos Pred Value : 0.0949
##          Neg Pred Value : 0.9630
##               Precision : 0.0949
##                  Recall : 0.4219
##                      F1 : 0.1550
##              Prevalence : 0.0498
##          Detection Rate : 0.0210
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.6056
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is Very LOW. We'll buld multiple models with SMOTE dataset, Standardized and SMOTE Standardized

30.2.4 Receiver Operating Characteristic Curve (ROC)



30.3 Naive Bayes Model

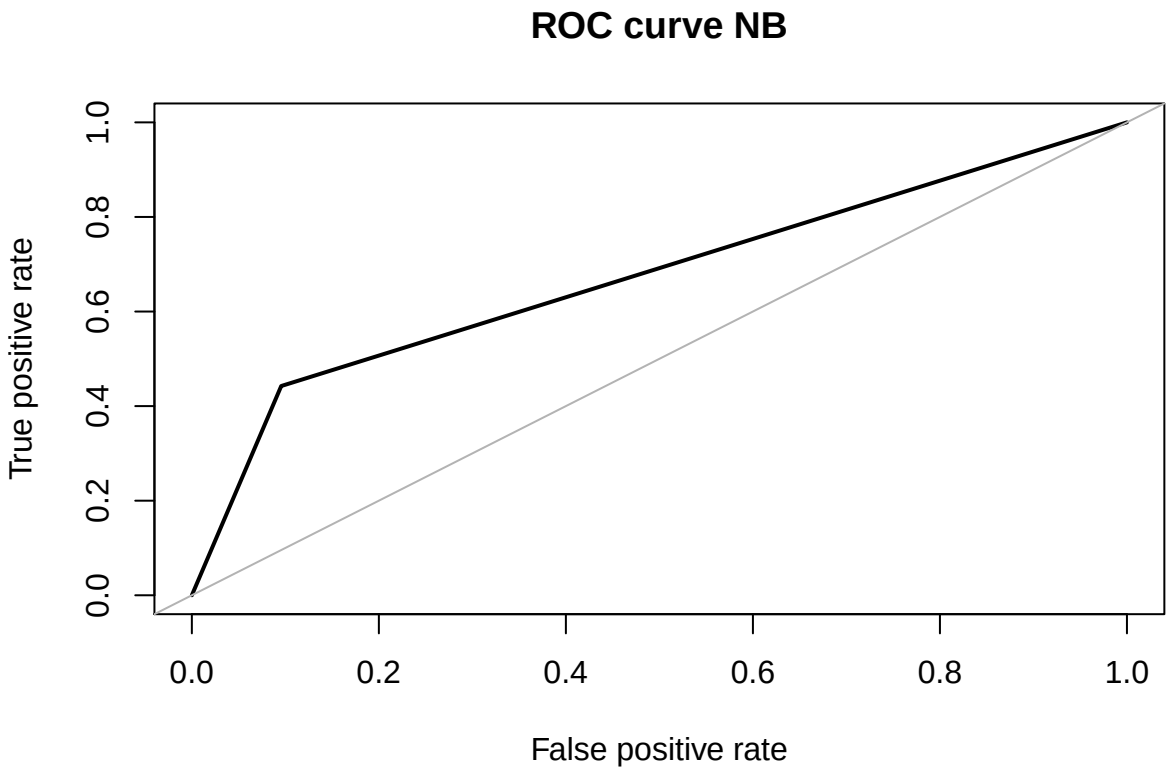
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6339  670
```

```
##        Yes 1110  881
##
##               Accuracy : 0.802
##                 95% CI : (0.794, 0.81)
##    No Information Rate : 0.828
##    P-Value [Acc > NIR] : 1
##
##                  Kappa : 0.377
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.5680
##             Specificity : 0.8510
##          Pos Pred Value : 0.4425
##          Neg Pred Value : 0.9044
##               Precision : 0.4425
##                  Recall : 0.5680
##                      F1 : 0.4975
##              Prevalence : 0.1723
##          Detection Rate : 0.0979
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7095
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is LOW. We'll try and improve using CART, Random Forest, Bagging and Boosting Model

30.3.1 Receiver Operating Characteristic Curve (ROC)



31 CART with Tuning Parameters

```
## CART
##
## 21000 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 16800, 16800, 16800, 16800, 16800
## Resampling results across tuning parameters:
##
##   cp       ROC   Sens  Spec
##   0.00048  0.73  0.94  0.37
##   0.00050  0.73  0.94  0.37
##   0.00054  0.73  0.94  0.37
##   0.00065  0.73  0.94  0.37
##   0.00069  0.73  0.94  0.37
##   0.00075  0.73  0.94  0.37
##   0.00086  0.72  0.95  0.35
##   0.00108  0.71  0.95  0.36
##   0.00115  0.71  0.95  0.36
```

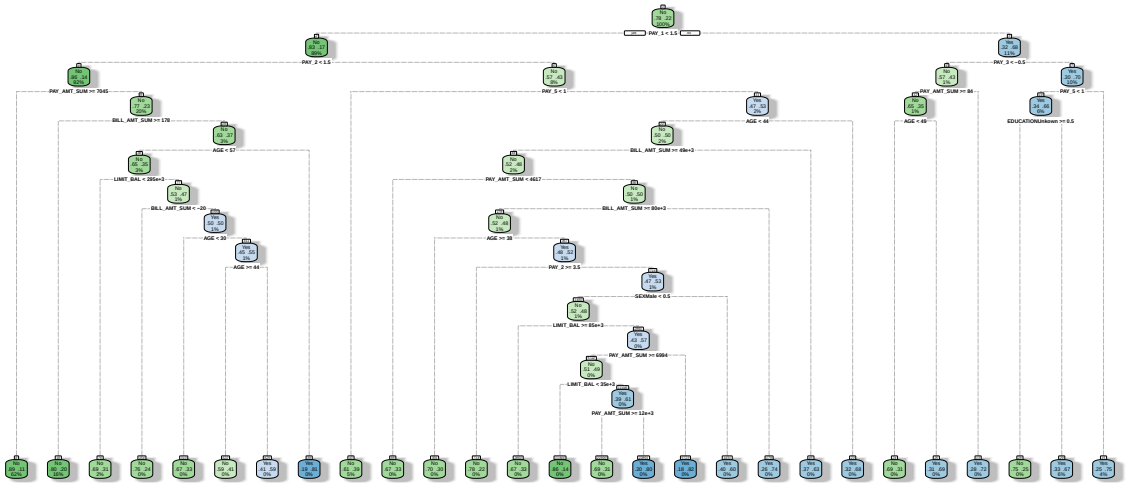
```
##   0.00129  0.70  0.95  0.36
##   0.00172  0.68  0.95  0.34
##   0.00280  0.68  0.95  0.35
##   0.00334  0.66  0.95  0.34
##   0.00409  0.66  0.95  0.34
##   0.17696  0.58  0.97  0.20
##
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.00075.
```

Findings

- nprune 18 is optimal prune

31.0.1 Plotting Tree

```
fancyRpartPlot(model_marsfeatenginerred$finalModel)
```



31.0.2 Predict on testData and Compute the confusion matrix

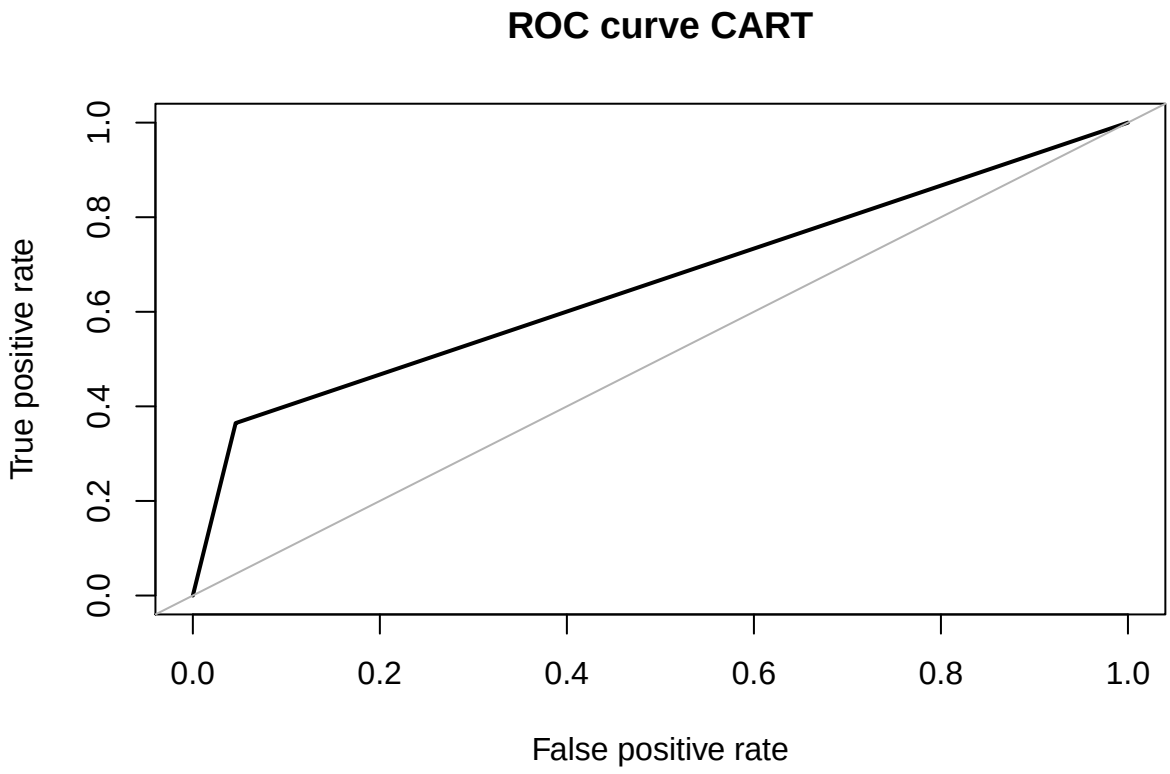
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6688 1265
##        Yes  321  726
##
##                Accuracy : 0.824
##                  95% CI : (0.816, 0.832)
##     No Information Rate : 0.779
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.384
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.3646
```

```
##             Specificity : 0.9542
##          Pos Pred Value : 0.6934
##          Neg Pred Value : 0.8409
##               Precision : 0.6934
##                  Recall : 0.3646
##                      F1 : 0.4779
##              Prevalence : 0.2212
##          Detection Rate : 0.0807
##    Detection Prevalence : 0.1163
##       Balanced Accuracy : 0.6594
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is LOW

31.0.3 Receiver Operating Characteristic Curve (ROC)



31.1 Random FOrest with tuning Parameters

```
## Random Forest
##
## 21000 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
```

```
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 16800, 16800, 16800, 16800, 16800
## Resampling results across tuning parameters:
##
##   mtry  ROC   Sens  Spec
##    2    0.77  0.95  0.36
##    5    0.76  0.94  0.37
##    9    0.76  0.93  0.38
##   13    0.76  0.93  0.38
##   17    0.75  0.93  0.38
##
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 2.
```

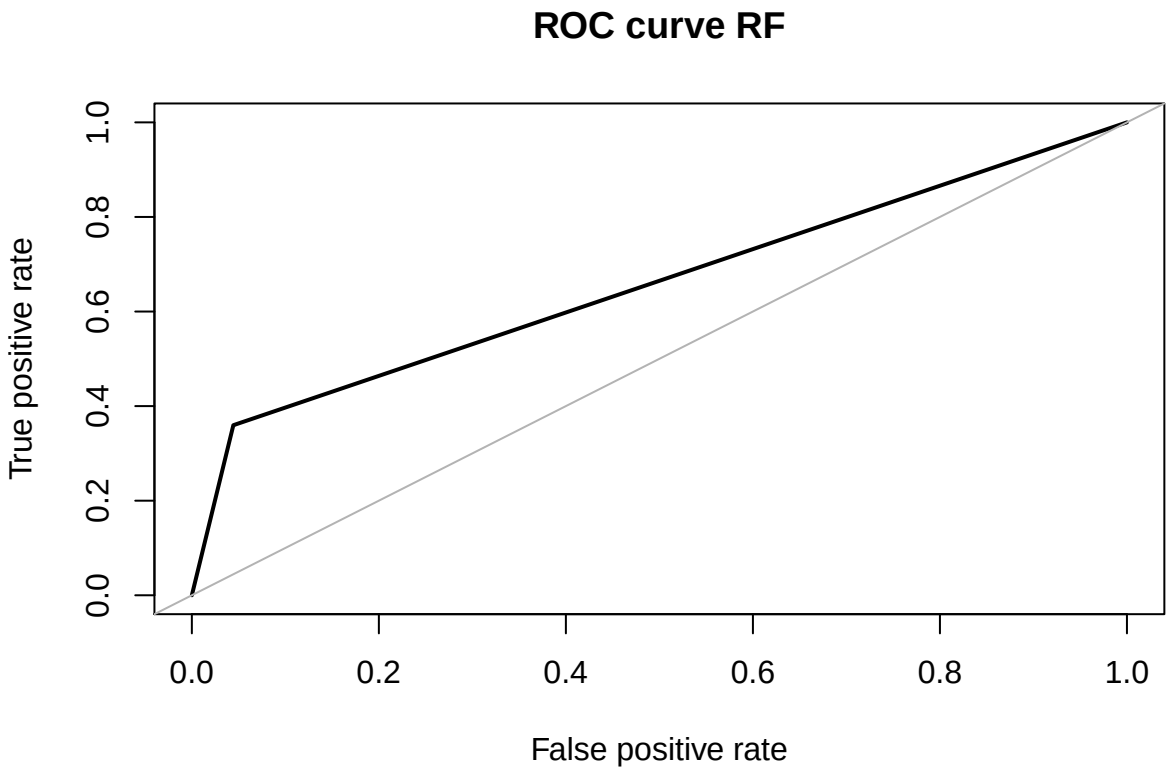
Findings

- mtry = 2 is optimal selected by the model

31.1.1 Predict RF

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6698 1275
##        Yes  311  716
##
##                Accuracy : 0.824
##                  95% CI : (0.816, 0.832)
##     No Information Rate : 0.779
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.381
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.3596
##             Specificity : 0.9556
##          Pos Pred Value : 0.6972
##          Neg Pred Value : 0.8401
##               Precision : 0.6972
##                  Recall : 0.3596
##                      F1 : 0.4745
##              Prevalence : 0.2212
##          Detection Rate : 0.0796
##    Detection Prevalence : 0.1141
##       Balanced Accuracy : 0.6576
##
##        'Positive' Class : Yes
##
```

31.1.2 Receiver Operating Characteristic Curve (ROC)



31.2 BAGGING

31.2.1 BAGGING

31.2.2 Predicting Bagging

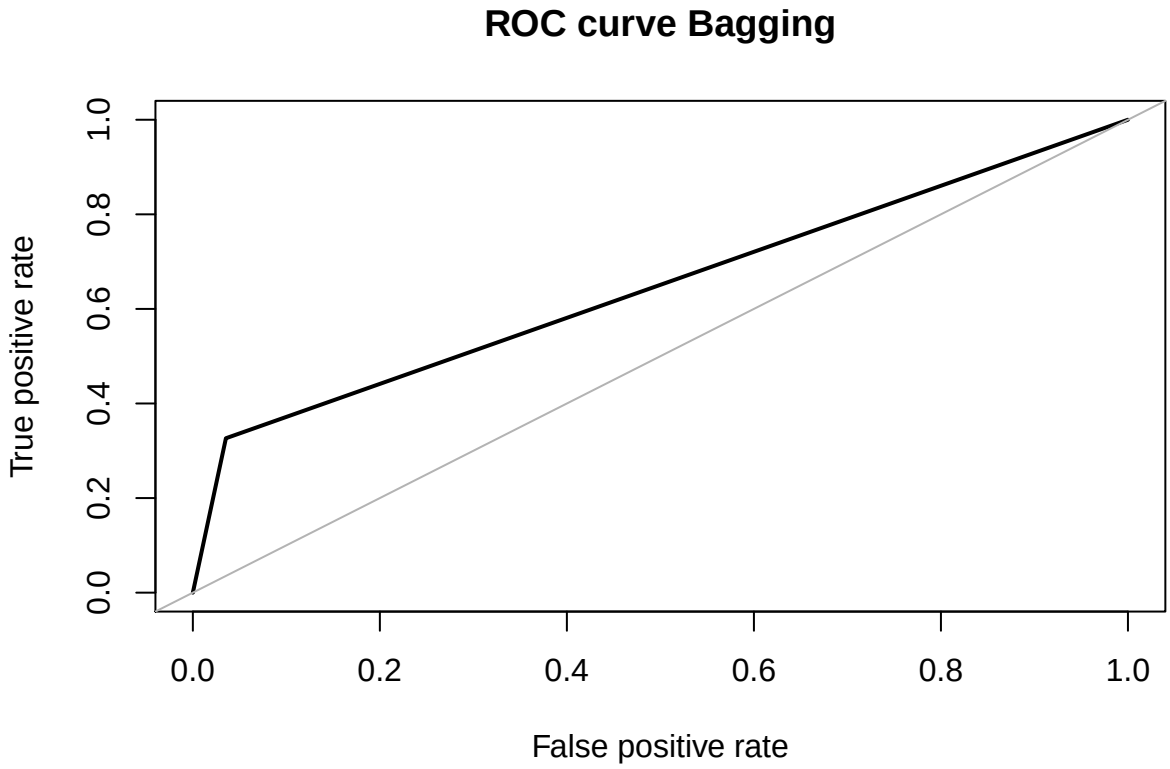
```
## [1] "factor"
```

31.2.3 Confusion Matrix

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6761  248
##        Yes 1341  650
##
##                Accuracy : 0.823
##                  95% CI : (0.815, 0.831)
##     No Information Rate : 0.9
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.362
##
##  Mcnemar's Test P-Value : <0.0000000000000002
```

```
##
##             Sensitivity : 0.7238
##             Specificity : 0.8345
##          Pos Pred Value : 0.3265
##          Neg Pred Value : 0.9646
##               Precision : 0.3265
##                  Recall : 0.7238
##                      F1 : 0.4500
##              Prevalence : 0.0998
##          Detection Rate : 0.0722
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7792
##
##        'Positive' Class : Yes
##
```

31.2.4 Receiver Operating Characteristic Curve (ROC)



31.3 Boosting

31.3.1 adaBoost with adabag

31.3.2 Print Tree

```
## [1] "formula"    "trees"      "weights"    "votes"      "prob"
## [6] "class"      "importance" "terms"      "call"
```

```
## [[1]]
## n= 21000
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
## 1) root 21000 4500 No (0.78 0.22)
##   2) PAY_1< 1.5 18775 3000 No (0.84 0.16) *
##   3) PAY_1>=1.5 2225  720 Yes (0.32 0.68) *
```

Predict on test set

```
##                Observed Class
## Predicted Class   No  Yes
##             No  6672 1262
##             Yes  337  729
```

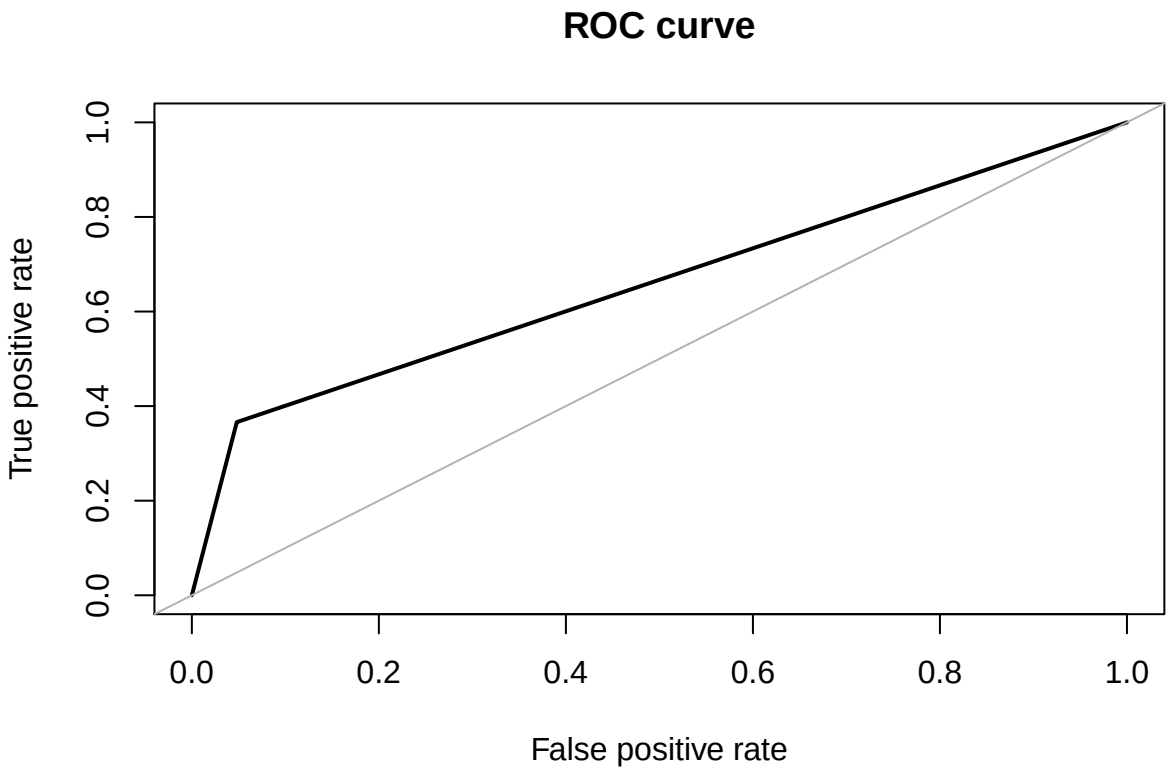
```
## [1] 0.18
```

Converting to factors

31.3.3 Confusion Matrix and evaluation

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6672  337
##        Yes 1262  729
##
##                Accuracy : 0.822
##                  95% CI : (0.814, 0.83)
##     No Information Rate : 0.882
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.382
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.684
##             Specificity : 0.841
##          Pos Pred Value : 0.366
##          Neg Pred Value : 0.952
##               Precision : 0.366
##                  Recall : 0.684
##                      F1 : 0.477
##              Prevalence : 0.118
##          Detection Rate : 0.081
##    Detection Prevalence : 0.221
##       Balanced Accuracy : 0.762
##
##        'Positive' Class : Yes
##
```

31.3.4 Receiver Operating Characteristic Curve (ROC)



32 Building Models with SMOTE Dataset

32.1 Logistic Regression on Feature Engineered Dataset

32.1.1 Full Model Stepwise

```
fullmodSMOTE <- glm(DEFAULT ~. ,family=binomial,data=SMOTE.train)
```

32.1.2 Empty Model

32.1.3 Backward Selection of significant variables

```
## Start:  AIC=48454
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_4 + PAY_5 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## <none>              48418 48454
## - LIMIT_BAL     1    48420 48454
## - AGE           1    48461 48495
## - PAY_6         1    48479 48513
## - PAY_5         1    48479 48513
## - PAY_4         1    48495 48529
## - PAY_3         1    48518 48552
```

```
## - MARRIAGE      2    48522 48554
## - EDUCATION     4    48592 48620
## - PAY_2         1    48658 48692
## - SEX           1    48669 48703
## - BILL_AMT_SUM  1    48717 48751
## - PAY_AMT_SUM   1    49082 49116
## - PAY_1         1    49793 49827
```

32.1.4 Variable Selection using direction BOTH

```
## Start:  AIC=57442
## DEFAULT ~ 1
##
##                Df Deviance   AIC
## + PAY_1         1    52595 52599
## + PAY_2         1    53935 53939
## + PAY_3         1    54451 54455
## + PAY_4         1    54677 54681
## + PAY_5         1    54902 54906
## + PAY_6         1    55230 55234
## + PAY_AMT_SUM   1    55245 55249
## + LIMIT_BAL     1    55427 55431
## + EDUCATION     4    56822 56832
## + SEX           1    57044 57048
## + MARRIAGE      2    57272 57278
## + BILL_AMT_SUM  1    57343 57347
## + AGE           1    57421 57425
## <none>               57440 57442
##
## Step:  AIC=52599
## DEFAULT ~ PAY_1
##
##                Df Deviance   AIC
## + PAY_AMT_SUM   1    50920 50926
## + PAY_2         1    51658 51664
## + LIMIT_BAL     1    51727 51733
## + PAY_3         1    51743 51749
## + PAY_4         1    51814 51820
## + PAY_5         1    51899 51905
## + BILL_AMT_SUM  1    51966 51972
## + PAY_6         1    52028 52034
## + EDUCATION     4    52189 52201
## + SEX           1    52277 52283
## + MARRIAGE      2    52432 52440
## + AGE           1    52539 52545
## <none>               52595 52599
## - PAY_1         1    57440 57442
##
## Step:  AIC=50926
## DEFAULT ~ PAY_1 + PAY_AMT_SUM
##
##                Df Deviance   AIC
## + PAY_2         1    50035 50043
## + PAY_3         1    50089 50097
```

```
## + PAY_4         1    50126 50134
## + PAY_5         1    50169 50177
## + PAY_6         1    50276 50284
## + SEX           1    50610 50618
## + EDUCATION     4    50629 50643
## + LIMIT_BAL     1    50718 50726
## + MARRIAGE      2    50764 50774
## + AGE           1    50836 50844
## + BILL_AMT_SUM  1    50885 50893
## <none>               50920 50926
## - PAY_AMT_SUM   1    52595 52599
## - PAY_1         1    55245 55249
##
## Step:  AIC=50043
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2
##
##                Df Deviance   AIC
## + PAY_4         1    49680 49690
## + PAY_5         1    49701 49711
## + PAY_3         1    49724 49734
## + SEX           1    49752 49762
## + PAY_6         1    49754 49764
## + EDUCATION     4    49778 49794
## + BILL_AMT_SUM  1    49871 49881
## + MARRIAGE      2    49879 49891
## + AGE           1    49928 49938
## + LIMIT_BAL     1    49968 49978
## <none>               50035 50043
## - PAY_2         1    50920 50926
## - PAY_AMT_SUM   1    51658 51664
## - PAY_1         1    52047 52053
##
## Step:  AIC=49690
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4
##
##                Df Deviance   AIC
## + SEX           1    49407 49419
## + BILL_AMT_SUM  1    49439 49451
## + EDUCATION     4    49436 49454
## + MARRIAGE      2    49519 49533
## + PAY_3         1    49545 49557
## + PAY_5         1    49550 49562
## + AGE           1    49565 49577
## + PAY_6         1    49565 49577
## + LIMIT_BAL     1    49651 49663
## <none>               49680 49690
## - PAY_4         1    50035 50043
## - PAY_2         1    50126 50134
## - PAY_1         1    51240 51248
## - PAY_AMT_SUM   1    51324 51332
##
## Step:  AIC=49419
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX
##
```

```
##                Df Deviance   AIC
## + BILL_AMT_SUM  1    49169 49183
## + EDUCATION     4    49168 49188
## + MARRIAGE      2    49241 49257
## + PAY_3         1    49275 49289
## + PAY_5         1    49280 49294
## + PAY_6         1    49293 49307
## + AGE           1    49310 49324
## + LIMIT_BAL     1    49380 49394
## <none>              49407 49419
## - SEX           1    49680 49690
## - PAY_4         1    49752 49762
## - PAY_2         1    49839 49849
## - PAY_1         1    50954 50964
## - PAY_AMT_SUM   1    51045 51055
##
## Step:  AIC=49183
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM
##
##                Df Deviance   AIC
## + EDUCATION     4    48952 48974
## + MARRIAGE      2    48999 49017
## + PAY_3         1    49003 49019
## + PAY_5         1    49007 49023
## + PAY_6         1    49017 49033
## + AGE           1    49049 49065
## <none>              49169 49183
## + LIMIT_BAL     1    49168 49184
## - BILL_AMT_SUM  1    49407 49419
## - SEX           1    49439 49451
## - PAY_4         1    49590 49602
## - PAY_2         1    49697 49709
## - PAY_AMT_SUM   1    49911 49923
## - PAY_1         1    50850 50862
##
## Step:  AIC=48974
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION
##
##                Df Deviance   AIC
## + PAY_3         1    48790 48814
## + PAY_5         1    48795 48819
## + MARRIAGE      2    48798 48824
## + PAY_6         1    48803 48827
## + AGE           1    48868 48892
## <none>              48952 48974
## + LIMIT_BAL     1    48952 48976
## - EDUCATION     4    49169 49183
## - BILL_AMT_SUM  1    49168 49188
## - SEX           1    49218 49238
## - PAY_4         1    49356 49376
## - PAY_2         1    49462 49482
## - PAY_AMT_SUM   1    49672 49692
## - PAY_1         1    50615 50635
```

```
##
## Step:  AIC=48814
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3
##
##                Df Deviance   AIC
## + MARRIAGE      2    48636 48664
## + PAY_5         1    48683 48709
## + PAY_6         1    48683 48709
## + AGE           1    48698 48724
## + LIMIT_BAL     1    48788 48814
## <none>              48790 48814
## - PAY_3         1    48952 48974
## - EDUCATION     4    49003 49019
## - PAY_4         1    48998 49020
## - BILL_AMT_SUM  1    49038 49060
## - SEX           1    49052 49074
## - PAY_2         1    49093 49115
## - PAY_AMT_SUM   1    49477 49499
## - PAY_1         1    50297 50319
##
## Step:  AIC=48664
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE
##
##                Df Deviance   AIC
## + PAY_5         1    48524 48554
## + PAY_6         1    48525 48555
## + AGE           1    48596 48626
## <none>              48636 48664
## + LIMIT_BAL     1    48636 48666
## - MARRIAGE      2    48790 48814
## - PAY_3         1    48798 48824
## - EDUCATION     4    48835 48855
## - PAY_4         1    48851 48877
## - BILL_AMT_SUM  1    48889 48915
## - SEX           1    48903 48929
## - PAY_2         1    48941 48967
## - PAY_AMT_SUM   1    49324 49350
## - PAY_1         1    50145 50171
##
## Step:  AIC=48554
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5
##
##                Df Deviance   AIC
## + PAY_6         1    48467 48499
## + AGE           1    48480 48512
## + LIMIT_BAL     1    48521 48553
## <none>              48524 48554
## - PAY_4         1    48626 48654
## - PAY_3         1    48635 48663
## - PAY_5         1    48636 48664
## - MARRIAGE      2    48683 48709
```

```
## - EDUCATION     4    48720 48742
## - PAY_2         1    48778 48806
## - SEX           1    48789 48817
## - BILL_AMT_SUM  1    48802 48830
## - PAY_AMT_SUM   1    49207 49235
## - PAY_1         1    49940 49968
##
## Step:  AIC=48499
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5 + PAY_6
##
##                Df Deviance   AIC
## + AGE           1    48420 48454
## + LIMIT_BAL     1    48461 48495
## <none>               48467 48499
## - PAY_6         1    48524 48554
## - PAY_5         1    48525 48555
## - PAY_4         1    48541 48571
## - PAY_3         1    48561 48591
## - MARRIAGE      2    48628 48656
## - EDUCATION     4    48663 48687
## - PAY_2         1    48698 48728
## - SEX           1    48732 48762
## - BILL_AMT_SUM  1    48761 48791
## - PAY_AMT_SUM   1    49148 49178
## - PAY_1         1    49833 49863
##
## Step:  AIC=48454
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5 + PAY_6 + AGE
##
##                Df Deviance   AIC
## + LIMIT_BAL     1    48418 48454
## <none>               48420 48454
## - AGE           1    48467 48499
## - PAY_6         1    48480 48512
## - PAY_5         1    48481 48513
## - PAY_4         1    48495 48527
## - PAY_3         1    48518 48550
## - MARRIAGE      2    48525 48555
## - EDUCATION     4    48594 48620
## - PAY_2         1    48658 48690
## - SEX           1    48671 48703
## - BILL_AMT_SUM  1    48732 48764
## - PAY_AMT_SUM   1    49105 49137
## - PAY_1         1    49796 49828
##
## Step:  AIC=48454
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5 + PAY_6 + AGE + LIMIT_BAL
##
##                Df Deviance   AIC
## <none>               48418 48454
## - LIMIT_BAL     1    48420 48454
```

```
## - AGE           1    48461 48495
## - PAY_6         1    48479 48513
## - PAY_5         1    48479 48513
## - PAY_4         1    48495 48529
## - PAY_3         1    48518 48552
## - MARRIAGE      2    48522 48554
## - EDUCATION     4    48592 48620
## - PAY_2         1    48658 48692
## - SEX           1    48669 48703
## - BILL_AMT_SUM  1    48717 48751
## - PAY_AMT_SUM   1    49082 49116
## - PAY_1         1    49793 49827
```

32.2 Fitting Model

```
##
## Call:
## glm(formula = DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_3 +
##     SEX + EDUCATION + BILL_AMT_SUM + PAY_5 + MARRIAGE + PAY_4 +
##     PAY_6 + AGE, family = "binomial", data = SMOTE.train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -3.593  -1.056   0.496   0.936   4.109
##
## Coefficients:
##                           Estimate    Std. Error z value             Pr(>|z|)
## (Intercept)           0.2126702954  0.0543723889    3.91    0.000091775695322
## PAY_1                 0.3829144068  0.0107031886   35.78 < 0.0000000000000002
## PAY_AMT_SUM          -0.0000099881  0.0000004408  -22.66 < 0.0000000000000002
## PAY_2                 0.1583983971  0.0103426096   15.32 < 0.0000000000000002
## PAY_3                 0.1044585980  0.0105902662    9.86 < 0.0000000000000002
## SEXMale               0.3504123863  0.0221924979   15.79 < 0.0000000000000002
## EDUCATIONHigh.School  0.1684313940  0.0323282642    5.21    0.000000188804815
## EDUCATIONOther       -1.0884479715  0.2507455108   -4.34    0.000014193434278
## EDUCATIONUniversity  -0.1392525536  0.0252242414   -5.52    0.000000033787406
## EDUCATIONUnkown      -0.9000829434  0.1364250648   -6.60    0.000000000041776
## BILL_AMT_SUM         -0.0000006776  0.0000000388  -17.45 < 0.0000000000000002
## PAY_5                 0.0849990208  0.0110064650    7.72    0.000000000000011
## MARRIAGEOther         0.2992876942  0.0880367143    3.40              0.00067
## MARRIAGESingle       -0.2133785938  0.0232130350   -9.19 < 0.0000000000000002
## PAY_4                 0.0931274266  0.0107953973    8.63 < 0.0000000000000002
## PAY_6                 0.0800572406  0.0104384990    7.67    0.000000000000017
## AGE                   0.0088857410  0.0013084228    6.79    0.000000000011122
##
## (Intercept)          ***
## PAY_1                ***
## PAY_AMT_SUM          ***
## PAY_2                ***
## PAY_3                ***
## SEXMale              ***
## EDUCATIONHigh.School ***
## EDUCATIONOther       ***
## EDUCATIONUniversity  ***
```

```
## EDUCATIONUnkown      ***
## BILL_AMT_SUM         ***
## PAY_5                ***
## MARRIAGEOther        ***
## MARRIAGESingle       ***
## PAY_4                ***
## PAY_6                ***
## AGE                  ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 57440  on 41806  degrees of freedom
## Residual deviance: 48420  on 41790  degrees of freedom
## AIC: 48454
##
## Number of Fisher Scoring iterations: 5
```

```
vif(FitLogMSMOTE)
```

```
##              GVIF Df GVIF^(1/(2*Df))
## PAY_1         1.3  1             1.1
## PAY_AMT_SUM   1.3  1             1.1
## PAY_2         1.5  1             1.2
## PAY_3         1.6  1             1.3
## SEX           1.0  1             1.0
## EDUCATION     1.1  4             1.0
## BILL_AMT_SUM  1.5  1             1.2
## PAY_5         1.7  1             1.3
## MARRIAGE      1.1  2             1.0
## PAY_4         1.7  1             1.3
## PAY_6         1.6  1             1.3
## AGE           1.1  1             1.1
```

32.2.1 Predict Test Set

32.2.2 Converting Prob to Classes

32.2.3 ConfusionMatrix and Training Model Evaluation

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  4824 3139
##        Yes 2352 7602
##
##                Accuracy : 0.694
##                  95% CI : (0.687, 0.7)
##     No Information Rate : 0.599
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.373
##
##  Mcnemar's Test P-Value : <0.0000000000000002
```

```
##
##             Sensitivity : 0.708
##             Specificity : 0.672
##          Pos Pred Value : 0.764
##          Neg Pred Value : 0.606
##               Precision : 0.764
##                  Recall : 0.708
##                      F1 : 0.735
##              Prevalence : 0.599
##          Detection Rate : 0.424
##    Detection Prevalence : 0.556
##       Balanced Accuracy : 0.690
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is Better compared to Regular Dataset. Let's try other models on this dataset to get it to perform better

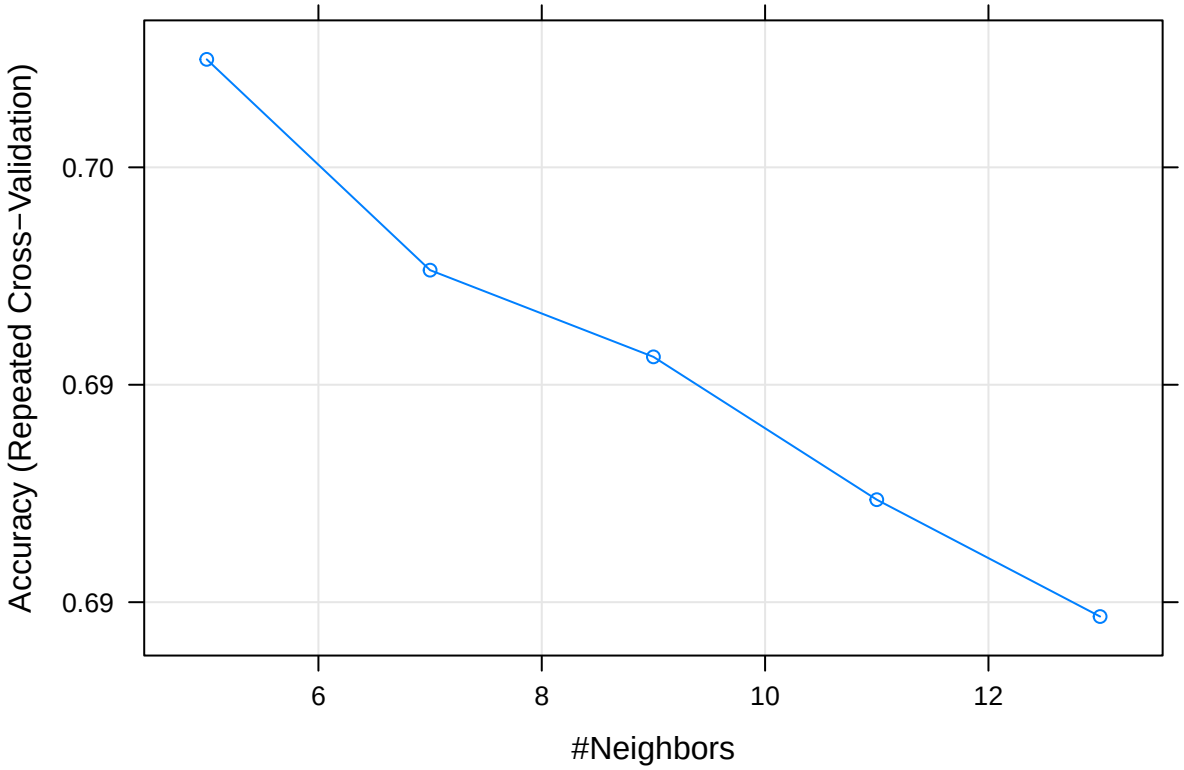
32.2.4 Let's build KNN model with Optimal K value. Also, validate model using 10 fold Cross Validation

```
knnCaretSMOTE <- train(DEFAULT~., data = SMOTE.train, method = "knn",
                 trControl=KNNControlCaretSMOTE,
                 tuneLength = 5)
```

```
knnCaretSMOTE
```

```
## k-Nearest Neighbors
##
## 41807 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 37625, 37626, 37626, 37626, 37627, 37627, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy  Kappa
##    5  0.70      0.39
##    7  0.69      0.38
##    9  0.69      0.38
##   11  0.69      0.37
##   13  0.68      0.37
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 5.
```

32.2.5 #Plotting Number of Neighbours Vs accuracy (based on repeated cross validation)



32.2.6 Let's predict for Test Data

32.2.7 Confusion Matrix

```
confusionMatrix( SMOTE.test$DEFAULT,testCaretKNNSMOTE,positive = "Yes", mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  5825 2138
##        Yes 3210 6744
##
##                Accuracy : 0.702
##                  95% CI : (0.695, 0.708)
##     No Information Rate : 0.504
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.404
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.759
##             Specificity : 0.645
##          Pos Pred Value : 0.678
```

```
##          Neg Pred Value : 0.732
##               Precision : 0.678
##                  Recall : 0.759
##                      F1 : 0.716
##              Prevalence : 0.496
##          Detection Rate : 0.376
##    Detection Prevalence : 0.556
##       Balanced Accuracy : 0.702
##
##        'Positive' Class : Yes
##
```

32.2.8 Let's start building Naive Bayes with Cross Validation using Caret

```
pred_nbSMOTE<-predict(nb_SMOTE,newdata = SMOTE.test[,-12])
```

```
confusionMatrix(pred_nbSMOTE, SMOTE.test$DEFAULT,positive = "Yes", mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6472 2649
##        Yes 1491 7305
##
##                Accuracy : 0.769
##                  95% CI : (0.763, 0.775)
##     No Information Rate : 0.556
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.539
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.734
##             Specificity : 0.813
##          Pos Pred Value : 0.830
##          Neg Pred Value : 0.710
##               Precision : 0.830
##                  Recall : 0.734
##                      F1 : 0.779
##              Prevalence : 0.556
##          Detection Rate : 0.408
##    Detection Prevalence : 0.491
##       Balanced Accuracy : 0.773
##
##        'Positive' Class : Yes
##
```

33 CART with Tuning Parameters

```
fitControlSMOTECART <- trainControl(
    method = 'cv',
```

```
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)
```

```
library(earth)
```

```
## Warning: package 'earth' was built under R version 3.5.3
```

```
## Loading required package: plotmo
```

```
## Warning: package 'plotmo' was built under R version 3.5.3
```

```
## Loading required package: plotrix
```

```
## Warning: package 'plotrix' was built under R version 3.5.3
```

```
##
## Attaching package: 'plotrix'
```

```
## The following object is masked from 'package:gplots':
##
##     plotCI
```

```
## Loading required package: TeachingDemos
```

```
## Warning: package 'TeachingDemos' was built under R version 3.5.3
```

```
##
## Attaching package: 'TeachingDemos'
```

```
## The following object is masked from 'package:klaR':
##
##     triplot
```

```
## The following objects are masked from 'package:Hmisc':
##
##     cnvrt.coords, subplot
```

```
set.seed(seed)
model_marsSMOTE = train(DEFAULT ~ ., data=SMOTE.train, method='rpart',parms = list(split = "information
```

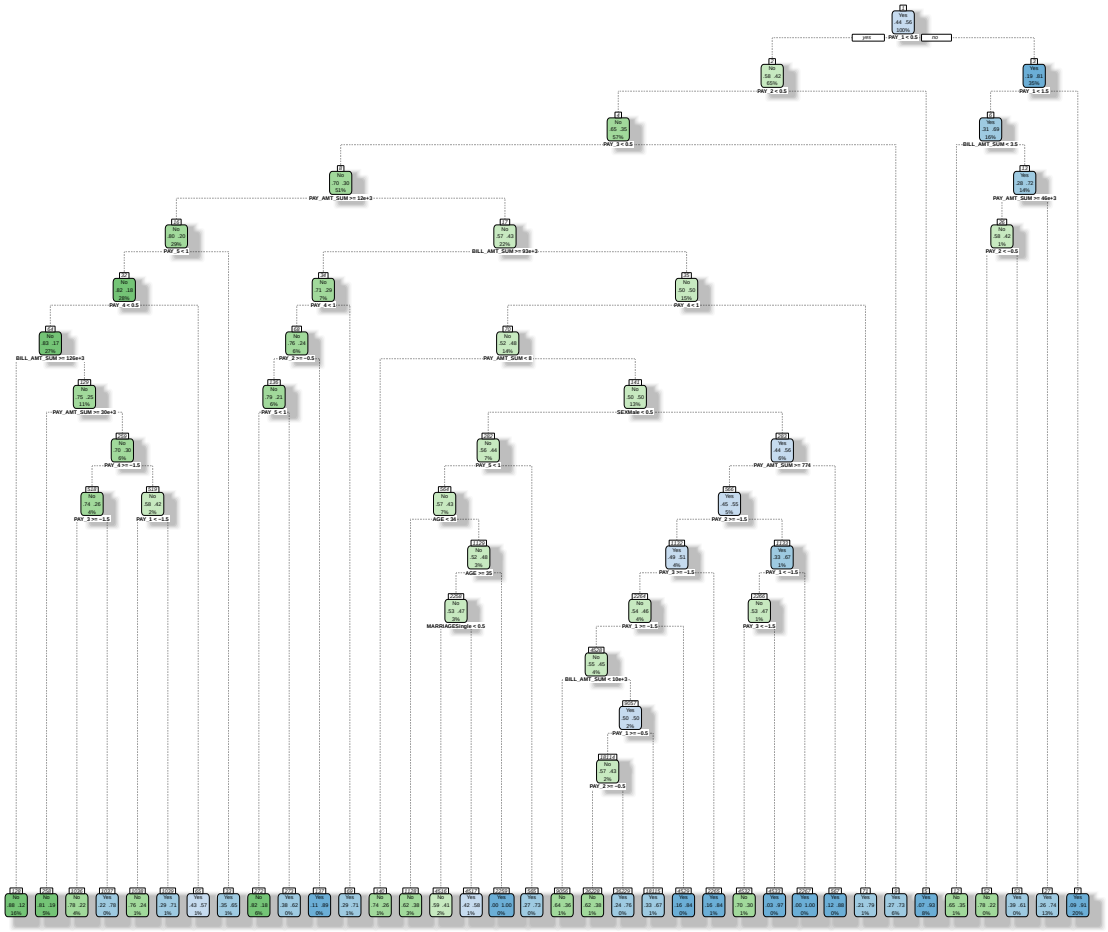
```
model_marsSMOTE
```

```
## CART
##
## 41807 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 33446, 33445, 33445, 33446, 33446
## Resampling results across tuning parameters:
##
##   cp      ROC   Sens  Spec
##   0.0016  0.85  0.76  0.82
##   0.0019  0.84  0.73  0.83
##   0.0019  0.84  0.73  0.83
##   0.0021  0.84  0.73  0.83
```

```
##   0.0024  0.83  0.74  0.82
##   0.0025  0.83  0.73  0.82
##   0.0027  0.83  0.74  0.82
##   0.0034  0.83  0.74  0.82
##   0.0040  0.83  0.74  0.82
##   0.0054  0.82  0.73  0.81
##   0.0090  0.79  0.79  0.74
##   0.0107  0.79  0.80  0.73
##   0.0661  0.76  0.82  0.67
##   0.1482  0.71  0.85  0.56
##   0.2403  0.57  0.34  0.80
##
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.0016.
```

33.1 Build Tree

```
fancyRpartPlot(model_marsSMOTE$finalModel)
```



Rattle 2020−Feb−12 06:40:36 Abhay

Predict on testData and Compute the confusion matrix

```
predictedCARTSMOTE <- predict(model_marsSMOTE, SMOTE.test)
```

```
confusionMatrix(reference = SMOTE.test$DEFAULT, data = predictedCARTSMOTE, mode='everything', positive=
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6164 1586
##        Yes 1799 8368
##
##                Accuracy : 0.811
##                  95% CI : (0.805, 0.817)
##     No Information Rate : 0.556
##     P-Value [Acc > NIR] : < 0.0000000000000002
##
##                   Kappa : 0.616
##
##  Mcnemar's Test P-Value : 0.000269
##
##             Sensitivity : 0.841
##             Specificity : 0.774
##          Pos Pred Value : 0.823
##          Neg Pred Value : 0.795
##               Precision : 0.823
##                  Recall : 0.841
##                      F1 : 0.832
##              Prevalence : 0.556
##          Detection Rate : 0.467
##    Detection Prevalence : 0.567
##       Balanced Accuracy : 0.807
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is Better compared to previous models. Let's try Random Forest on this dataset

33.2 Random FOrest with tuning Parameters

```
model_rfSMOTE
```

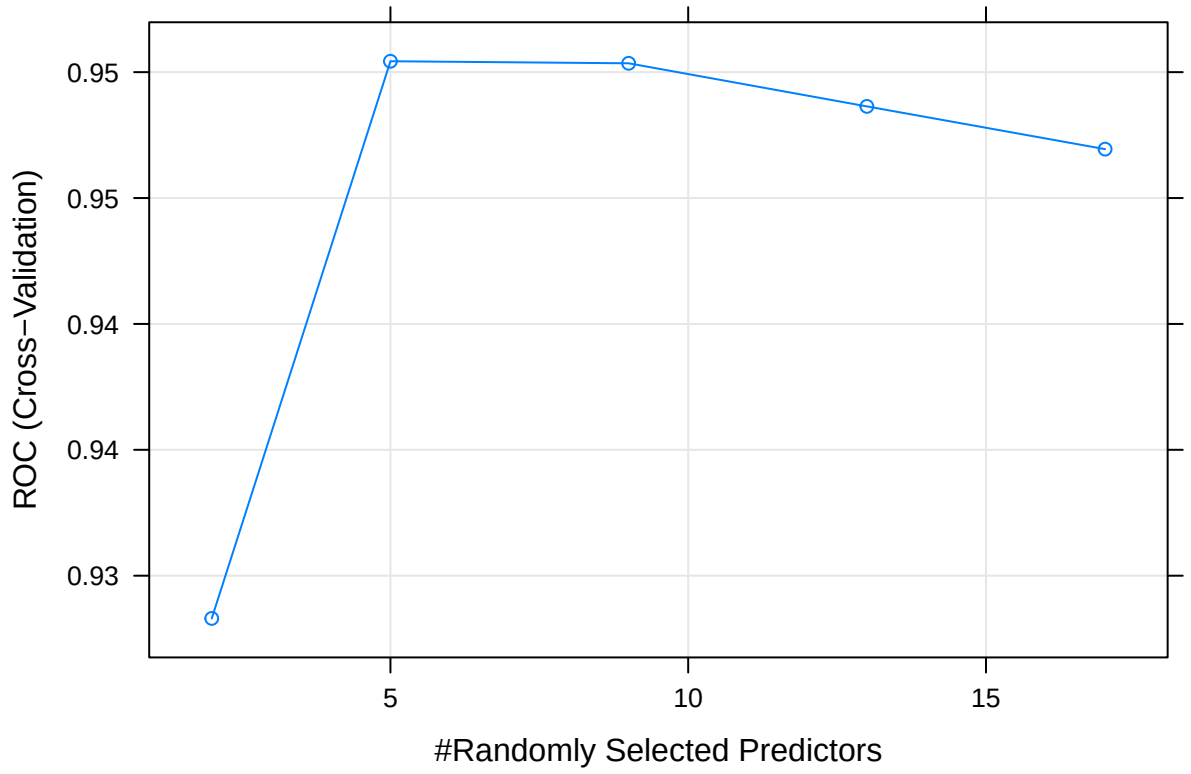
```
## Random Forest
##
## 41807 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 33446, 33445, 33445, 33446, 33446
## Resampling results across tuning parameters:
##
```

```
##   mtry  ROC   Sens  Spec
##    2    0.93  0.85  0.87
##    5    0.95  0.90  0.88
##    9    0.95  0.90  0.88
##   13    0.95  0.90  0.88
##   17    0.95  0.90  0.88
##
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 5.
```

33.2.1 Predict RF

List the importance of the variables. Larger the MeanDecrease values, the more important the variable. Look at the help files to get a better sense of how these are computed.

```
plot(model_rfSMOTE, top = 20)
```



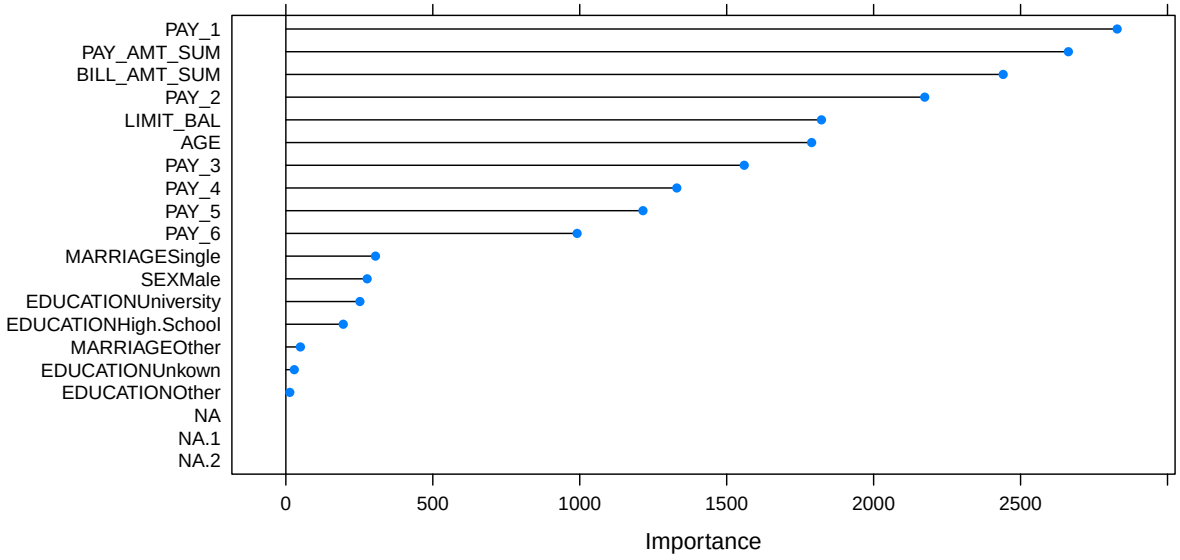

```
RFSMOTEimp <- varImp(model_rfSMOTE, scale = FALSE)

RFSMOTEimp
```

```
## rf variable importance
##
##                     Overall
## PAY_1                2828.7
## PAY_AMT_SUM          2662.7
## BILL_AMT_SUM         2440.9
## PAY_2                2173.8
```

```
## LIMIT_BAL             1822.5
## AGE                   1789.1
## PAY_3                 1559.7
## PAY_4                 1330.3
## PAY_5                 1215.2
## PAY_6                  991.0
## MARRIAGESingle         304.9
## SEXMale                276.6
## EDUCATIONUniversity    251.9
## EDUCATIONHigh.School   195.3
## MARRIAGEOther           49.4
## EDUCATIONUnkown         28.6
## EDUCATIONOther          13.4
```

```
plot(RFSMOTEimp, top = 20)
```




```
confusionMatrix(reference = SMOTE.test$DEFAULT, data = predictedRFSMOTE, mode='everything', positive='Y
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  7308 1095
##        Yes  655 8859
##
##                Accuracy : 0.902
##                  95% CI : (0.898, 0.907)
##     No Information Rate : 0.556
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.803
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.890
```

```
##             Specificity : 0.918
##          Pos Pred Value : 0.931
##          Neg Pred Value : 0.870
##               Precision : 0.931
##                  Recall : 0.890
##                      F1 : 0.910
##              Prevalence : 0.556
##          Detection Rate : 0.494
##    Detection Prevalence : 0.531
##       Balanced Accuracy : 0.904
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is Better compared to all the models so far. Sensitivity is also good.

34 Model using Standardized Dataset

34.1 Logistic Regression on Feature Engineered Dataset

34.1.1 Full Model Stepwise

```
fullmodStandardized <- glm(DEFAULT ~. ,family=binomial,data=standardDS.train)
```

34.1.2 Empty Model

```
emptyModeltandardized<- glm(DEFAULT ~ 1,family=binomial,data = standardDS.train)
```

34.1.3 Backward Selection of significant variables

```
backwardstandardized = step(fullmodStandardized)
```

```
## Start:  AIC=19584
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_4 + PAY_5 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## - PAY_4         1    19548 19582
## - PAY_5         1    19548 19582
## - PAY_6         1    19549 19583
## <none>              19548 19584
## - BILL_AMT_SUM  1    19556 19590
## - PAY_3         1    19556 19590
## - SEX           1    19556 19590
## - MARRIAGE      2    19561 19593
## - AGE           1    19560 19594
## - LIMIT_BAL     1    19561 19595
## - PAY_2         1    19569 19603
## - EDUCATION     4    19589 19617
## - PAY_AMT_SUM   1    19631 19665
## - PAY_1         1    20266 20300
##
```

```
## Step:  AIC=19582
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_5 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## - PAY_5         1    19548 19580
## - PAY_6         1    19549 19581
## <none>               19548 19582
## - BILL_AMT_SUM  1    19556 19588
## - SEX           1    19556 19588
## - PAY_3         1    19558 19590
## - MARRIAGE      2    19561 19591
## - AGE           1    19560 19592
## - LIMIT_BAL     1    19561 19593
## - PAY_2         1    19569 19601
## - EDUCATION     4    19589 19615
## - PAY_AMT_SUM   1    19631 19663
## - PAY_1         1    20269 20301
##
## Step:  AIC=19580
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## <none>               19548 19580
## - PAY_6         1    19552 19582
## - BILL_AMT_SUM  1    19556 19586
## - SEX           1    19557 19587
## - MARRIAGE      2    19561 19589
## - PAY_3         1    19560 19590
## - AGE           1    19560 19590
## - LIMIT_BAL     1    19562 19592
## - PAY_2         1    19570 19600
## - EDUCATION     4    19590 19614
## - PAY_AMT_SUM   1    19631 19661
## - PAY_1         1    20273 20303
```

34.1.4 Variable Selection using direction BOTH

```
forwardsstandardized  = step(emptyModeltandardized,scope=list(lower=formula(emptyModeltandardized),uppe
```

```
## Start:  AIC=22195
## DEFAULT ~ 1
##
##                Df Deviance   AIC
## + PAY_1         1    19994 19998
## + PAY_2         1    20735 20739
## + PAY_3         1    21086 21090
## + PAY_4         1    21300 21304
## + PAY_5         1    21398 21402
## + PAY_6         1    21511 21515
## + LIMIT_BAL     1    21680 21684
## + PAY_AMT_SUM   1    21782 21786
## + EDUCATION     4    22059 22069
```

```
## + SEX           1    22158 22162
## + MARRIAGE      2    22175 22181
## + AGE           1    22185 22189
## <none>               22193 22195
## + BILL_AMT_SUM  1    22192 22196
##
## Step:  AIC=19998
## DEFAULT ~ PAY_1
##
##                Df Deviance   AIC
## + PAY_AMT_SUM   1    19796 19802
## + LIMIT_BAL     1    19866 19872
## + PAY_2         1    19882 19888
## + PAY_3         1    19889 19895
## + PAY_4         1    19928 19934
## + BILL_AMT_SUM  1    19933 19939
## + PAY_5         1    19938 19944
## + EDUCATION     4    19938 19950
## + PAY_6         1    19948 19954
## + MARRIAGE      2    19972 19980
## + AGE           1    19974 19980
## + SEX           1    19980 19986
## <none>               19994 19998
## - PAY_1         1    22193 22195
##
## Step:  AIC=19802
## DEFAULT ~ PAY_1 + PAY_AMT_SUM
##
##                Df Deviance   AIC
## + PAY_2         1    19688 19696
## + PAY_3         1    19693 19701
## + PAY_4         1    19727 19735
## + PAY_5         1    19734 19742
## + PAY_6         1    19741 19749
## + EDUCATION     4    19750 19764
## + LIMIT_BAL     1    19758 19766
## + AGE           1    19773 19781
## + MARRIAGE      2    19774 19784
## + SEX           1    19781 19789
## <none>               19796 19802
## + BILL_AMT_SUM  1    19794 19802
## - PAY_AMT_SUM   1    19994 19998
## - PAY_1         1    21782 21786
##
## Step:  AIC=19696
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2
##
##                Df Deviance   AIC
## + EDUCATION     4    19647 19663
## + AGE           1    19661 19671
## + MARRIAGE      2    19665 19677
## + PAY_3         1    19668 19678
## + LIMIT_BAL     1    19671 19681
## + PAY_4         1    19675 19685
```

```
## + PAY_5          1    19675 19685
## + PAY_6          1    19676 19686
## + SEX            1    19677 19687
## + BILL_AMT_SUM   1    19677 19687
## <none>                19688 19696
## - PAY_2          1    19796 19802
## - PAY_AMT_SUM    1    19882 19888
## - PAY_1          1    20474 20480
##
## Step:  AIC=19663
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION
##
##                Df Deviance   AIC
## + AGE           1    19622 19640
## + MARRIAGE      2    19624 19644
## + PAY_3         1    19627 19645
## + LIMIT_BAL     1    19629 19647
## + PAY_4         1    19634 19652
## + PAY_5         1    19634 19652
## + SEX           1    19635 19653
## + PAY_6         1    19636 19654
## + BILL_AMT_SUM  1    19637 19655
## <none>                19647 19663
## - EDUCATION     4    19688 19696
## - PAY_2         1    19750 19764
## - PAY_AMT_SUM   1    19833 19847
## - PAY_1         1    20432 20446
##
## Step:  AIC=19640
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE
##
##                Df Deviance   AIC
## + LIMIT_BAL     1    19595 19615
## + PAY_3         1    19601 19621
## + PAY_4         1    19608 19628
## + PAY_5         1    19608 19628
## + BILL_AMT_SUM  1    19610 19630
## + PAY_6         1    19610 19630
## + SEX           1    19613 19633
## + MARRIAGE      2    19613 19635
## <none>                19622 19640
## - AGE           1    19647 19663
## - EDUCATION     4    19661 19671
## - PAY_2         1    19729 19745
## - PAY_AMT_SUM   1    19813 19829
## - PAY_1         1    20408 20424
##
## Step:  AIC=19615
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL
##
##                Df Deviance   AIC
## + PAY_3         1    19578 19600
## + PAY_4         1    19585 19607
## + PAY_5         1    19586 19608
```

```
## + MARRIAGE      2    19584 19608
## + PAY_6         1    19587 19609
## + SEX           1    19588 19610
## + BILL_AMT_SUM  1    19591 19613
## <none>               19595 19615
## - LIMIT_BAL     1    19622 19640
## - AGE           1    19629 19647
## - EDUCATION     4    19638 19650
## - PAY_2         1    19682 19700
## - PAY_AMT_SUM   1    19708 19726
## - PAY_1         1    20362 20380
##
## Step:  AIC=19600
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3
##
##                Df Deviance   AIC
## + MARRIAGE      2    19567 19593
## + SEX           1    19572 19596
## + BILL_AMT_SUM  1    19572 19596
## + PAY_6         1    19576 19600
## + PAY_5         1    19576 19600
## <none>               19578 19600
## + PAY_4         1    19577 19601
## - PAY_3         1    19595 19615
## - PAY_2         1    19600 19620
## - LIMIT_BAL     1    19601 19621
## - AGE           1    19613 19633
## - EDUCATION     4    19621 19635
## - PAY_AMT_SUM   1    19694 19714
## - PAY_1         1    20318 20338
##
## Step:  AIC=19593
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3 + MARRIAGE
##
##                Df Deviance   AIC
## + SEX           1    19558 19586
## + BILL_AMT_SUM  1    19561 19589
## + PAY_6         1    19564 19592
## + PAY_5         1    19564 19592
## <none>               19567 19593
## + PAY_4         1    19566 19594
## - MARRIAGE      2    19578 19600
## - AGE           1    19581 19605
## - PAY_3         1    19584 19608
## - PAY_2         1    19588 19612
## - LIMIT_BAL     1    19592 19616
## - EDUCATION     4    19611 19629
## - PAY_AMT_SUM   1    19680 19704
## - PAY_1         1    20305 20329
##
## Step:  AIC=19586
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
```

```
##     PAY_3 + MARRIAGE + SEX
##
##                Df Deviance   AIC
## + BILL_AMT_SUM  1    19552 19582
## + PAY_6         1    19556 19586
## + PAY_5         1    19556 19586
## <none>               19558 19586
## + PAY_4         1    19557 19587
## - SEX           1    19567 19593
## - MARRIAGE      2    19572 19596
## - AGE           1    19570 19596
## - PAY_3         1    19575 19601
## - PAY_2         1    19579 19605
## - LIMIT_BAL     1    19583 19609
## - EDUCATION     4    19602 19622
## - PAY_AMT_SUM   1    19673 19699
## - PAY_1         1    20296 20322
##
## Step:  AIC=19582
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM
##
##                Df Deviance   AIC
## + PAY_6         1    19548 19580
## + PAY_5         1    19549 19581
## <none>               19552 19582
## + PAY_4         1    19551 19583
## - BILL_AMT_SUM  1    19558 19586
## - SEX           1    19561 19589
## - MARRIAGE      2    19565 19591
## - AGE           1    19564 19592
## - LIMIT_BAL     1    19567 19595
## - PAY_3         1    19571 19599
## - PAY_2         1    19575 19603
## - EDUCATION     4    19594 19616
## - PAY_AMT_SUM   1    19634 19662
## - PAY_1         1    20296 20324
##
## Step:  AIC=19580
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
##     PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM + PAY_6
##
##                Df Deviance   AIC
## <none>               19548 19580
## + PAY_5         1    19548 19582
## - PAY_6         1    19552 19582
## + PAY_4         1    19548 19582
## - BILL_AMT_SUM  1    19556 19586
## - SEX           1    19557 19587
## - MARRIAGE      2    19561 19589
## - PAY_3         1    19560 19590
## - AGE           1    19560 19590
## - LIMIT_BAL     1    19562 19592
## - PAY_2         1    19570 19600
```

```
## - EDUCATION     4    19590 19614
## - PAY_AMT_SUM   1    19631 19661
## - PAY_1         1    20273 20303
```

34.1.5 Logistic Regression Model

```
FitLogModelstandardized <- glm(DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
    PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM + PAY_6, data=standardDS.train,family= binomial(link='logit'))
```

34.1.6 Checking for Multicollinearity

```
vif(FitLogModelstandardized)
```

```
##              GVIF Df GVIF^(1/(2*Df))
## PAY_1         1.5  1             1.2
## PAY_AMT_SUM   1.4  1             1.2
## PAY_2         2.6  1             1.6
## EDUCATION     1.2  4             1.0
## AGE           1.4  1             1.2
## LIMIT_BAL     1.5  1             1.2
## PAY_3         2.4  1             1.6
## MARRIAGE      1.3  2             1.1
## SEX           1.0  1             1.0
## BILL_AMT_SUM  1.5  1             1.2
## PAY_6         1.6  1             1.3
```

No Multicollinearity

```
summary(FitLogModelstandardized)
```

```
##
## Call:
## glm(formula = DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION +
##     AGE + LIMIT_BAL + PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM +
##     PAY_6, family = binomial(link = "logit"), data = standardDS.train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -3.141  -0.700  -0.553  -0.291   3.125
##
## Coefficients:
##                      Estimate Std. Error z value             Pr(>|z|)
## (Intercept)           -1.3585     0.0436  -31.17 < 0.0000000000000002 ***
## PAY_1                  0.6402     0.0237   27.01 < 0.0000000000000002 ***
## PAY_AMT_SUM           -0.3303     0.0409   -8.08  0.00000000000000067 ***
## PAY_2                  0.1302     0.0282    4.62  0.00000378992846073 ***
## EDUCATIONHigh.School  -0.0817     0.0565   -1.45              0.14837
## EDUCATIONOther        -1.6768     0.5957   -2.81              0.00488 **
## EDUCATIONUniversity   -0.0847     0.0424   -2.00              0.04560 *
## EDUCATIONUnkown       -1.2117     0.2676   -4.53  0.00000596346444830 ***
## AGE                    0.0712     0.0205    3.48              0.00051 ***
## LIMIT_BAL             -0.0881     0.0243   -3.62              0.00029 ***
## PAY_3                  0.0954     0.0277    3.44              0.00059 ***
## MARRIAGEOther         -0.1603     0.1554   -1.03              0.30228
## MARRIAGESingle        -0.1472     0.0414   -3.55              0.00038 ***
```

```
## SEXMale                0.1073     0.0367    2.93             0.00344 **
## BILL_AMT_SUM          -0.0648     0.0237   -2.73             0.00625 **
## PAY_6                  0.0430     0.0224    1.92             0.05472 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 22193  on 20999  degrees of freedom
## Residual deviance: 19548  on 20984  degrees of freedom
## AIC: 19580
##
## Number of Fisher Scoring iterations: 5
```

34.1.7 Predict Test Set

```
logpredstandarized<- predict(FitLogModelstandardized,standardDS.test[-12],type = "response")
```

34.1.8 Converting Prob to Classes

```
ypredlogstandardized <- as.factor(ifelse(logpredstandarized > 0.5,"Yes","No"))
```

34.1.9 Confusion Matrix

```
confusionMatrix( standardDS.test$DEFAULT,ypredlogstandardized,positive = "Yes", mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6852  157
##        Yes 1515  476
##
##                Accuracy : 0.814
##                  95% CI : (0.806, 0.822)
##     No Information Rate : 0.93
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.287
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.7520
##             Specificity : 0.8189
##          Pos Pred Value : 0.2391
##          Neg Pred Value : 0.9776
##               Precision : 0.2391
##                  Recall : 0.7520
##                      F1 : 0.3628
##              Prevalence : 0.0703
##          Detection Rate : 0.0529
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7855
```

```
##
##        'Positive' Class : Yes
##
```

Findings

- Accuracy is good. However, Sensitivitycan be improved. We'll try other models

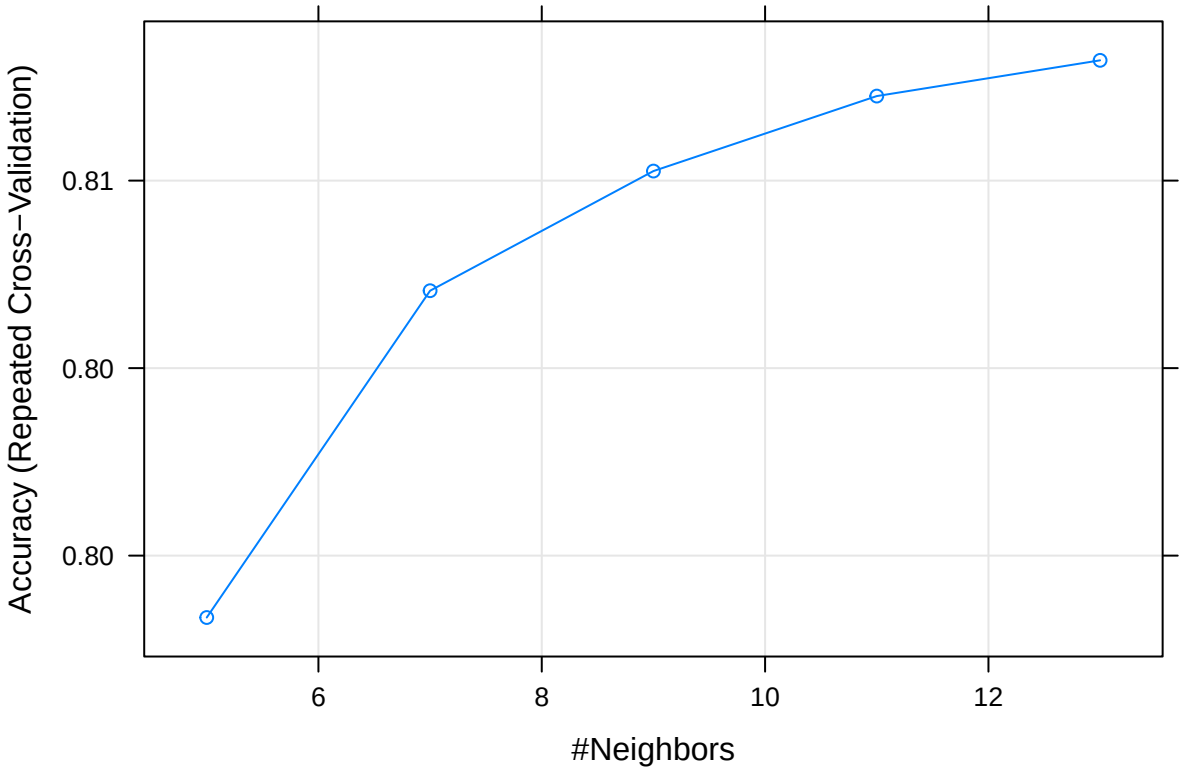
34.2 KNN Model

```
knnCaretstandardized
```

```
## k-Nearest Neighbors
##
## 21000 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 18900, 18900, 18900, 18901, 18900, 18899, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy  Kappa
##    5  0.79      0.32
##    7  0.80      0.33
##    9  0.81      0.34
##   11  0.81      0.34
##   13  0.81      0.34
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 13.
```

34.2.1 #Plotting Number of Neighbours Vs accuracy (based on repeated cross validation)

```
plot(knnCaretstandardized)
```



Findings

- 13 is the best K value

34.2.2 Let's predict for Test Data

```
testCaretstandardized <- predict(knnCaretstandardized, newdata = standardDS.test)
```

34.2.3 Confusion Matrix

```
confusionMatrix( standardDS.test$DEFAULT,testCaretstandardized, positive = "Yes", mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6621  388
##        Yes 1272  719
##
##                Accuracy : 0.816
##                  95% CI : (0.807, 0.824)
##     No Information Rate : 0.877
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.364
##
```

```
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.6495
##             Specificity : 0.8388
##          Pos Pred Value : 0.3611
##          Neg Pred Value : 0.9446
##               Precision : 0.3611
##                  Recall : 0.6495
##                      F1 : 0.4642
##              Prevalence : 0.1230
##          Detection Rate : 0.0799
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7442
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is OK. Let's improve using Naive Bayes, CART, Random Forest and Boosting

34.2.4 Let's start building Naive Bayes with Cross Validation using Caret

```
library(e1071)

nb_standarized<-naiveBayes(x=standardDS.train[,-12], y=standardDS.train[,12])
```

```
pred_nbstandardized<-predict(nb_standarized,newdata = standardDS.test[,-12])
```

```
confusionMatrix( standardDS.test$DEFAULT,pred_nbstandardized,positive = "Yes", mode = "everything")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6339  670
##        Yes 1110  881
##
##                Accuracy : 0.802
##                  95% CI : (0.794, 0.81)
##     No Information Rate : 0.828
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.377
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.5680
##             Specificity : 0.8510
##          Pos Pred Value : 0.4425
##          Neg Pred Value : 0.9044
##               Precision : 0.4425
##                  Recall : 0.5680
##                      F1 : 0.4975
##              Prevalence : 0.1723
```

```
##          Detection Rate : 0.0979
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7095
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity has improved. Let's improve even more using CART, Random Forest and Boosting

35 CART with Tuning Parameters

```
library(earth)
```

```
set.seed(seed)
model_marsstandardized = train(DEFAULT ~ ., data=standardDS.train, method='rpart',parms = list(split =
```

```
model_marsstandardized
```

```
## CART
##
## 21000 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 16800, 16800, 16800, 16800, 16800
## Resampling results across tuning parameters:
##
##   cp       ROC   Sens  Spec
##   0.00048  0.73  0.94  0.37
##   0.00050  0.73  0.94  0.37
##   0.00054  0.73  0.94  0.37
##   0.00065  0.73  0.94  0.37
##   0.00069  0.73  0.94  0.37
##   0.00075  0.73  0.94  0.37
##   0.00086  0.72  0.95  0.35
##   0.00108  0.71  0.95  0.36
##   0.00115  0.71  0.95  0.36
##   0.00129  0.70  0.95  0.36
##   0.00172  0.68  0.95  0.34
##   0.00280  0.68  0.95  0.35
##   0.00334  0.66  0.95  0.34
##   0.00409  0.66  0.95  0.34
##   0.17696  0.58  0.97  0.20
##
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.00075.
```

Findings

- nprune cp = 0.0007534984

35.0.1 Predict on testData and Compute the confusion matrix

```
predictedstandardized <- predict(model_marsstandardized, standardDS.test)
```

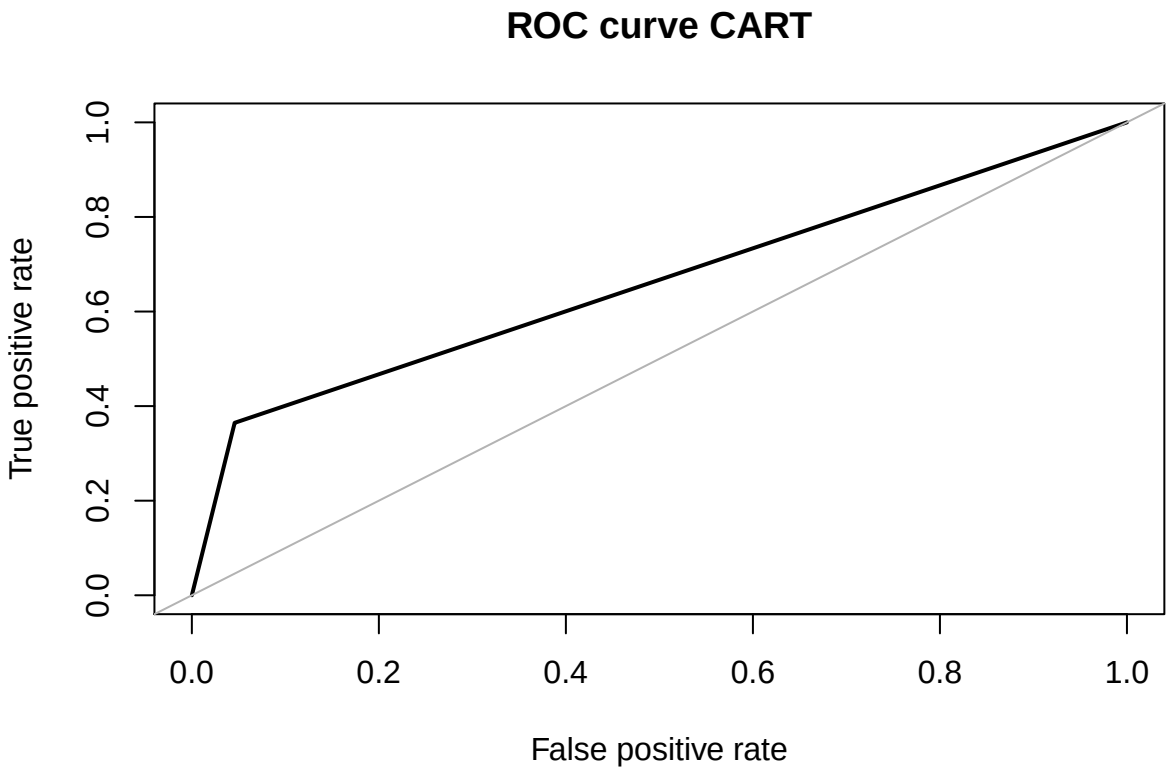
```
confusionMatrix(reference = standardDS.test$DEFAULT, data = predictedstandardized , mode='everything', p
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6688 1265
##        Yes  321  726
##
##                Accuracy : 0.824
##                  95% CI : (0.816, 0.832)
##     No Information Rate : 0.779
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.384
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.3646
##             Specificity : 0.9542
##          Pos Pred Value : 0.6934
##          Neg Pred Value : 0.8409
##               Precision : 0.6934
##                  Recall : 0.3646
##                      F1 : 0.4779
##              Prevalence : 0.2212
##          Detection Rate : 0.0807
##    Detection Prevalence : 0.1163
##       Balanced Accuracy : 0.6594
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity has improved. Let's improve even more using Random Forest and Boosting

35.0.2 Receiver Operating Characteristic Curve (ROC)



35.1 Random FOrest with tuning Parameters

```
model_rfstandardized
```

```
## Random Forest
##
## 21000 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 16800, 16800, 16800, 16800, 16800
## Resampling results across tuning parameters:
##
##   mtry  ROC   Sens  Spec
##    2    0.77  0.95  0.36
##    5    0.76  0.94  0.37
##    9    0.76  0.94  0.38
##   13    0.76  0.93  0.38
##   17    0.75  0.93  0.38
##
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 2.
```

Findings

- mtry = 2 is optimal selected by the model

35.1.1 Predict RF

```
predictedRFstandardize <- predict(model_rfstandardized, standardDS.test)
```

```
confusionMatrix(reference = standardDS.test$DEFAULT, data = predictedRFstandardize, mode='everything', p
```

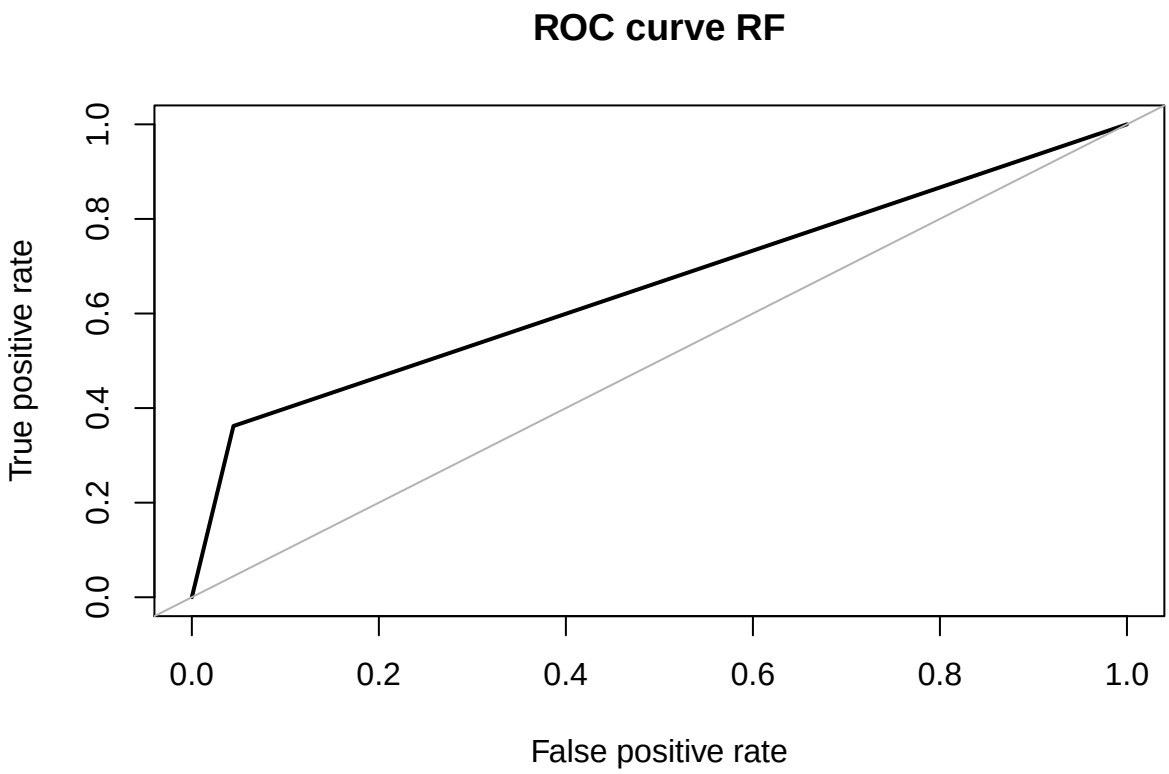
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6697 1270
##        Yes  312  721
##
##                Accuracy : 0.824
##                  95% CI : (0.816, 0.832)
##     No Information Rate : 0.779
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.384
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.3621
##             Specificity : 0.9555
##          Pos Pred Value : 0.6980
##          Neg Pred Value : 0.8406
##               Precision : 0.6980
##                  Recall : 0.3621
##                      F1 : 0.4769
##              Prevalence : 0.2212
##          Detection Rate : 0.0801
##    Detection Prevalence : 0.1148
##       Balanced Accuracy : 0.6588
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is not good.

35.1.2 Receiver Operating Characteristic Curve (ROC) RF

```
ROCtestRF<- roc.curve(standardDS.test$DEFAULT,predictedRFstandardize, main="ROC curve RF")
```



35.2 BAGGING

35.2.1 BAGGING

35.2.2 Predicting Bagging

```
## [1] "factor"
```

35.2.3 Confusion Matrix

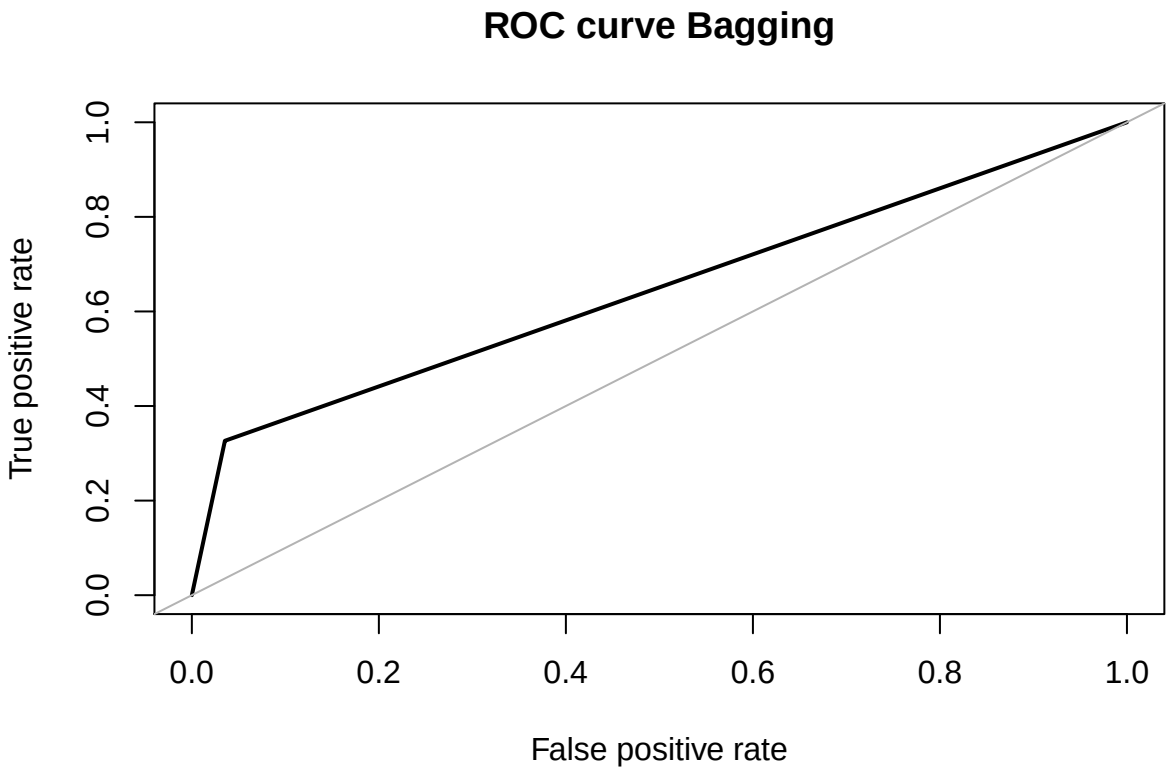
```
confusionMatrix(standardDS.test$DEFAULT, BaggingCarstandardize$class,positive = "Yes", mode = "everythi
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6761  248
##        Yes 1341  650
##
##                Accuracy : 0.823
##                  95% CI : (0.815, 0.831)
##     No Information Rate : 0.9
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.362
##
##  Mcnemar's Test P-Value : <0.0000000000000002
```

```
##
##             Sensitivity : 0.7238
##             Specificity : 0.8345
##          Pos Pred Value : 0.3265
##          Neg Pred Value : 0.9646
##               Precision : 0.3265
##                  Recall : 0.7238
##                      F1 : 0.4500
##              Prevalence : 0.0998
##          Detection Rate : 0.0722
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7792
##
##        'Positive' Class : Yes
##
```

35.2.4 Receiver Operating Characteristic Curve (ROC)

```
ROCtestBAGGING<- roc.curve(standardDS.test$DEFAULT,BaggingCarstandardize$class, main="ROC curve Bagging
```



35.3 Boosting

35.3.1 adaBoost with adabag

```
adaboostmodelstandardize <- boosting(DEFAULT~., data=standardDS.train, boos=TRUE, mfinal=50)
```

35.3.2 Print Tree

```
print(names(adaboostmodelstandardize))
```

```
## [1] "formula"    "trees"      "weights"    "votes"      "prob"
## [6] "class"      "importance" "terms"      "call"
```

```
print(adaboostmodelstandardize$trees[1])
```

```
## [[1]]
## n= 21000
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
## 1) root 21000 4500 No (0.78 0.22)
##   2) PAY_1< 1.3 18773 3000 No (0.84 0.16) *
##   3) PAY_1>=1.3 2227  710 Yes (0.32 0.68) *
```

Predict on test set

```
predAdaBooststandaridize = predict(adaboostmodelstandardize, standardDS.test)
print(predAdaBooststandaridize$confusion)
```

```
##                Observed Class
## Predicted Class   No  Yes
##             No  6713 1303
##             Yes  296  688
```

```
print(predAdaBooststandaridize$error)
```

```
## [1] 0.18
```

Converting to factors

35.3.3 Confusion Matrix and evaluation

```
confusionMatrix(standardDS.test$DEFAULT,predAdaBooststandaridize$class,mode="everything",positive = "Yes
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6713  296
##        Yes 1303  688
##
##                Accuracy : 0.822
##                  95% CI : (0.814, 0.83)
##     No Information Rate : 0.891
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.37
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.6992
##             Specificity : 0.8375
```

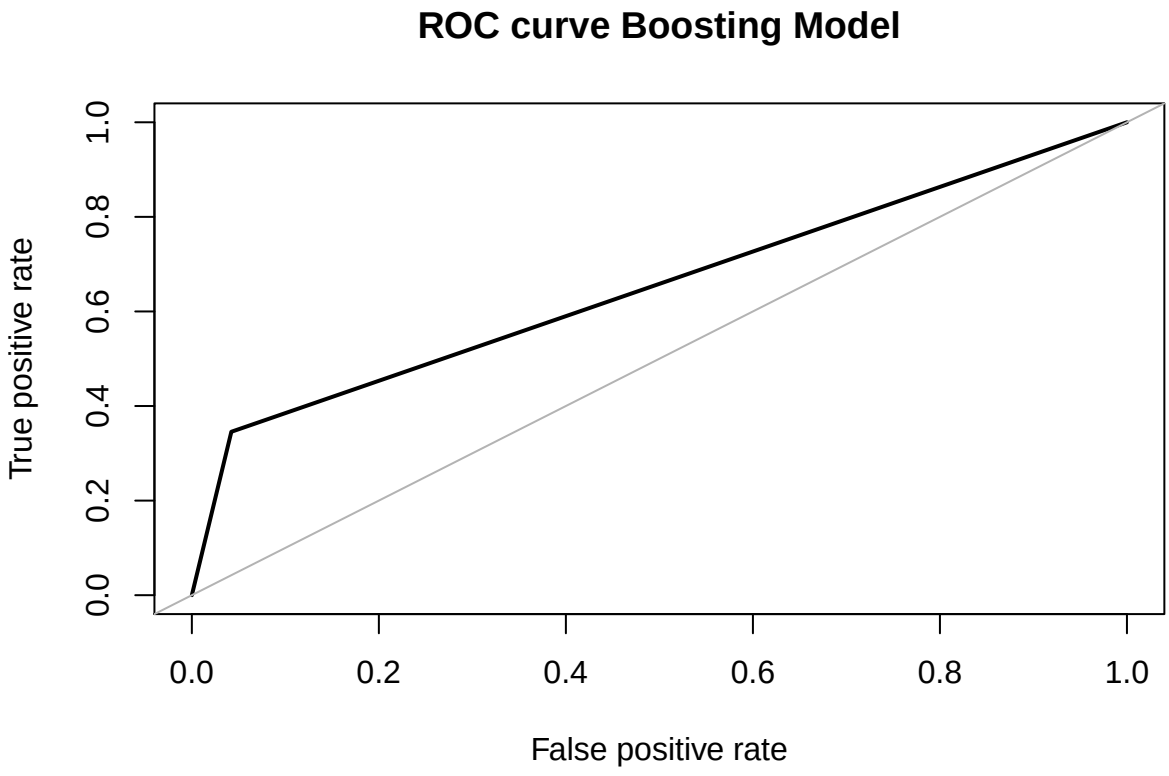
```
##          Pos Pred Value : 0.3456
##          Neg Pred Value : 0.9578
##               Precision : 0.3456
##                  Recall : 0.6992
##                      F1 : 0.4625
##              Prevalence : 0.1093
##          Detection Rate : 0.0764
##    Detection Prevalence : 0.2212
##       Balanced Accuracy : 0.7683
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity is not as good as BAGGING

35.3.4 Receiver Operating Characteristic Curve (ROC)

```
ROCAdaBoost<- roc.curve(standardDS.test$DEFAULT,predAdaBooststandaridize$class, main="ROC curve Boostin
```



36 Create another Dataset called SMOTE Standardize.

36.1 Create SMOTE Standardzed Dataset

```
SMOTESTANDARTDATASET <- SMOTEdataset
```

36.2 Run Standard Deviation to Standardize the Dataset

```
stanvector <-c(1,5,6,7,8,9,10,11,13,14)
```

```
options(digits=2)

for (i in stanvector) {

SMOTESTANDARTDATASET[,i] <- scale(SMOTESTANDARTDATASET[,i],center = TRUE, scale = TRUE)

}
```

36.2.1 Split 70 30 Ratio Train and Test

```
set.seed(seed)

sampleSMOTESTANDARTDATASET <- sample.split(SMOTESTANDARTDATASET$DEFAULT,SplitRatio = 0.7)
SMOTESTANDARTDATASET.train <- subset(SMOTESTANDARTDATASET,sampleSMOTESTANDARTDATASET == TRUE)
SMOTESTANDARTDATASET.test <- subset(SMOTESTANDARTDATASET,sampleSMOTESTANDARTDATASET == FALSE)
```

37 Building Odels using SMOTE Standardized Dataset

37.1 Logistic Regression

37.1.1 Empty Model

```
emptyModelSMOTESTANDART<- glm(DEFAULT ~ 1,family=binomial,data = SMOTESTANDARTDATASET.train)
```

37.1.2 Backward Selection of significant variables

```
backwardSMOTESTANDART = step(fullmodSMOTESTANDART)
```

```
## Start:  AIC=48454
## DEFAULT ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_1 +
##     PAY_2 + PAY_3 + PAY_4 + PAY_5 + PAY_6 + BILL_AMT_SUM + PAY_AMT_SUM
##
##                Df Deviance   AIC
## <none>              48418 48454
## - LIMIT_BAL     1    48420 48454
## - AGE           1    48461 48495
## - PAY_6         1    48479 48513
## - PAY_5         1    48479 48513
## - PAY_4         1    48495 48529
## - PAY_3         1    48518 48552
## - MARRIAGE      2    48522 48554
## - EDUCATION     4    48592 48620
## - PAY_2         1    48658 48692
## - SEX           1    48669 48703
## - BILL_AMT_SUM  1    48717 48751
## - PAY_AMT_SUM   1    49082 49116
```

```
## - PAY_1         1    49793 49827
```

37.1.3 Variable Selection using direction BOTH

```
forwardSMOTESTANDART  = step(emptyModelSMOTESTANDART,scope=list(lower=formula(emptyModelSMOTESTANDART),u
```

```
## Start:  AIC=57442
## DEFAULT ~ 1
##
##                Df Deviance   AIC
## + PAY_1         1    52595 52599
## + PAY_2         1    53935 53939
## + PAY_3         1    54451 54455
## + PAY_4         1    54677 54681
## + PAY_5         1    54902 54906
## + PAY_6         1    55230 55234
## + PAY_AMT_SUM   1    55245 55249
## + LIMIT_BAL     1    55427 55431
## + EDUCATION     4    56822 56832
## + SEX           1    57044 57048
## + MARRIAGE      2    57272 57278
## + BILL_AMT_SUM  1    57343 57347
## + AGE           1    57421 57425
## <none>               57440 57442
##
## Step:  AIC=52599
## DEFAULT ~ PAY_1
##
##                Df Deviance   AIC
## + PAY_AMT_SUM   1    50920 50926
## + PAY_2         1    51658 51664
## + LIMIT_BAL     1    51727 51733
## + PAY_3         1    51743 51749
## + PAY_4         1    51814 51820
## + PAY_5         1    51899 51905
## + BILL_AMT_SUM  1    51966 51972
## + PAY_6         1    52028 52034
## + EDUCATION     4    52189 52201
## + SEX           1    52277 52283
## + MARRIAGE      2    52432 52440
## + AGE           1    52539 52545
## <none>               52595 52599
## - PAY_1         1    57440 57442
##
## Step:  AIC=50926
## DEFAULT ~ PAY_1 + PAY_AMT_SUM
##
##                Df Deviance   AIC
## + PAY_2         1    50035 50043
## + PAY_3         1    50089 50097
## + PAY_4         1    50126 50134
## + PAY_5         1    50169 50177
## + PAY_6         1    50276 50284
## + SEX           1    50610 50618
```

```
## + EDUCATION     4    50629 50643
## + LIMIT_BAL     1    50718 50726
## + MARRIAGE      2    50764 50774
## + AGE           1    50836 50844
## + BILL_AMT_SUM  1    50885 50893
## <none>               50920 50926
## - PAY_AMT_SUM   1    52595 52599
## - PAY_1         1    55245 55249
##
## Step:  AIC=50043
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2
##
##                Df Deviance   AIC
## + PAY_4         1    49680 49690
## + PAY_5         1    49701 49711
## + PAY_3         1    49724 49734
## + SEX           1    49752 49762
## + PAY_6         1    49754 49764
## + EDUCATION     4    49778 49794
## + BILL_AMT_SUM  1    49871 49881
## + MARRIAGE      2    49879 49891
## + AGE           1    49928 49938
## + LIMIT_BAL     1    49968 49978
## <none>               50035 50043
## - PAY_2         1    50920 50926
## - PAY_AMT_SUM   1    51658 51664
## - PAY_1         1    52047 52053
##
## Step:  AIC=49690
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4
##
##                Df Deviance   AIC
## + SEX           1    49407 49419
## + BILL_AMT_SUM  1    49439 49451
## + EDUCATION     4    49436 49454
## + MARRIAGE      2    49519 49533
## + PAY_3         1    49545 49557
## + PAY_5         1    49550 49562
## + AGE           1    49565 49577
## + PAY_6         1    49565 49577
## + LIMIT_BAL     1    49651 49663
## <none>               49680 49690
## - PAY_4         1    50035 50043
## - PAY_2         1    50126 50134
## - PAY_1         1    51240 51248
## - PAY_AMT_SUM   1    51324 51332
##
## Step:  AIC=49419
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX
##
##                Df Deviance   AIC
## + BILL_AMT_SUM  1    49169 49183
## + EDUCATION     4    49168 49188
## + MARRIAGE      2    49241 49257
```

```
## + PAY_3         1    49275 49289
## + PAY_5         1    49280 49294
## + PAY_6         1    49293 49307
## + AGE           1    49310 49324
## + LIMIT_BAL     1    49380 49394
## <none>               49407 49419
## - SEX           1    49680 49690
## - PAY_4         1    49752 49762
## - PAY_2         1    49839 49849
## - PAY_1         1    50954 50964
## - PAY_AMT_SUM   1    51045 51055
##
## Step:  AIC=49183
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM
##
##                Df Deviance   AIC
## + EDUCATION     4    48952 48974
## + MARRIAGE      2    48999 49017
## + PAY_3         1    49003 49019
## + PAY_5         1    49007 49023
## + PAY_6         1    49017 49033
## + AGE           1    49049 49065
## <none>               49169 49183
## + LIMIT_BAL     1    49168 49184
## - BILL_AMT_SUM  1    49407 49419
## - SEX           1    49439 49451
## - PAY_4         1    49590 49602
## - PAY_2         1    49697 49709
## - PAY_AMT_SUM   1    49911 49923
## - PAY_1         1    50850 50862
##
## Step:  AIC=48974
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION
##
##                Df Deviance   AIC
## + PAY_3         1    48790 48814
## + PAY_5         1    48795 48819
## + MARRIAGE      2    48798 48824
## + PAY_6         1    48803 48827
## + AGE           1    48868 48892
## <none>               48952 48974
## + LIMIT_BAL     1    48952 48976
## - EDUCATION     4    49169 49183
## - BILL_AMT_SUM  1    49168 49188
## - SEX           1    49218 49238
## - PAY_4         1    49356 49376
## - PAY_2         1    49462 49482
## - PAY_AMT_SUM   1    49672 49692
## - PAY_1         1    50615 50635
##
## Step:  AIC=48814
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3
```

```
##
##                Df Deviance   AIC
## + MARRIAGE      2    48636 48664
## + PAY_5         1    48683 48709
## + PAY_6         1    48683 48709
## + AGE           1    48698 48724
## + LIMIT_BAL     1    48788 48814
## <none>               48790 48814
## - PAY_3         1    48952 48974
## - EDUCATION     4    49003 49019
## - PAY_4         1    48998 49020
## - BILL_AMT_SUM  1    49038 49060
## - SEX           1    49052 49074
## - PAY_2         1    49093 49115
## - PAY_AMT_SUM   1    49477 49499
## - PAY_1         1    50297 50319
##
## Step:  AIC=48664
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE
##
##                Df Deviance   AIC
## + PAY_5         1    48524 48554
## + PAY_6         1    48525 48555
## + AGE           1    48596 48626
## <none>               48636 48664
## + LIMIT_BAL     1    48636 48666
## - MARRIAGE      2    48790 48814
## - PAY_3         1    48798 48824
## - EDUCATION     4    48835 48855
## - PAY_4         1    48851 48877
## - BILL_AMT_SUM  1    48889 48915
## - SEX           1    48903 48929
## - PAY_2         1    48941 48967
## - PAY_AMT_SUM   1    49324 49350
## - PAY_1         1    50145 50171
##
## Step:  AIC=48554
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5
##
##                Df Deviance   AIC
## + PAY_6         1    48467 48499
## + AGE           1    48480 48512
## + LIMIT_BAL     1    48521 48553
## <none>               48524 48554
## - PAY_4         1    48626 48654
## - PAY_3         1    48635 48663
## - PAY_5         1    48636 48664
## - MARRIAGE      2    48683 48709
## - EDUCATION     4    48720 48742
## - PAY_2         1    48778 48806
## - SEX           1    48789 48817
## - BILL_AMT_SUM  1    48802 48830
```

```
## - PAY_AMT_SUM   1    49207 49235
## - PAY_1         1    49940 49968
##
## Step:  AIC=48499
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5 + PAY_6
##
##                Df Deviance   AIC
## + AGE           1    48420 48454
## + LIMIT_BAL     1    48461 48495
## <none>               48467 48499
## - PAY_6         1    48524 48554
## - PAY_5         1    48525 48555
## - PAY_4         1    48541 48571
## - PAY_3         1    48561 48591
## - MARRIAGE      2    48628 48656
## - EDUCATION     4    48663 48687
## - PAY_2         1    48698 48728
## - SEX           1    48732 48762
## - BILL_AMT_SUM  1    48761 48791
## - PAY_AMT_SUM   1    49148 49178
## - PAY_1         1    49833 49863
##
## Step:  AIC=48454
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5 + PAY_6 + AGE
##
##                Df Deviance   AIC
## + LIMIT_BAL     1    48418 48454
## <none>               48420 48454
## - AGE           1    48467 48499
## - PAY_6         1    48480 48512
## - PAY_5         1    48481 48513
## - PAY_4         1    48495 48527
## - PAY_3         1    48518 48550
## - MARRIAGE      2    48525 48555
## - EDUCATION     4    48594 48620
## - PAY_2         1    48658 48690
## - SEX           1    48671 48703
## - BILL_AMT_SUM  1    48732 48764
## - PAY_AMT_SUM   1    49105 49137
## - PAY_1         1    49796 49828
##
## Step:  AIC=48454
## DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_4 + SEX + BILL_AMT_SUM +
##     EDUCATION + PAY_3 + MARRIAGE + PAY_5 + PAY_6 + AGE + LIMIT_BAL
##
##                Df Deviance   AIC
## <none>               48418 48454
## - LIMIT_BAL     1    48420 48454
## - AGE           1    48461 48495
## - PAY_6         1    48479 48513
## - PAY_5         1    48479 48513
## - PAY_4         1    48495 48529
```

```
## - PAY_3         1    48518 48552
## - MARRIAGE      2    48522 48554
## - EDUCATION     4    48592 48620
## - PAY_2         1    48658 48692
## - SEX           1    48669 48703
## - BILL_AMT_SUM  1    48717 48751
## - PAY_AMT_SUM   1    49082 49116
## - PAY_1         1    49793 49827
```

Logistic Regression Model

```
FitLogModelSMOTESTANDARD <- glm(DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_3 + SEX + EDUCATION +
    BILL_AMT_SUM + PAY_5 + MARRIAGE + PAY_4 + PAY_6 + AGE, data=SMOTESTANDARTDATASET.train,family= binor
```

37.1.4 Checking for Multicollinearity

```
vif(FitLogModelSMOTESTANDARD)
```

```
##              GVIF Df GVIF^(1/(2*Df))
## PAY_1         1.3  1             1.1
## PAY_AMT_SUM   1.3  1             1.1
## PAY_2         1.5  1             1.2
## PAY_3         1.6  1             1.3
## SEX           1.0  1             1.0
## EDUCATION     1.1  4             1.0
## BILL_AMT_SUM  1.5  1             1.2
## PAY_5         1.7  1             1.3
## MARRIAGE      1.1  2             1.0
## PAY_4         1.7  1             1.3
## PAY_6         1.6  1             1.3
## AGE           1.1  1             1.1
```

No Multicollinearity

```
summary(FitLogModelSMOTESTANDARD)
```

```
##
## Call:
## glm(formula = DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_3 +
##     SEX + EDUCATION + BILL_AMT_SUM + PAY_5 + MARRIAGE + PAY_4 +
##     PAY_6 + AGE, family = binomial(link = "logit"), data = SMOTESTANDARTDATASET.train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -3.593  -1.056   0.496   0.936   4.109
##
## Coefficients:
##                      Estimate Std. Error z value             Pr(>|z|)
## (Intercept)            0.2640     0.0255   10.34 < 0.0000000000000002 ***
## PAY_1                  0.5360     0.0150   35.78 < 0.0000000000000002 ***
## PAY_AMT_SUM           -0.5097     0.0225  -22.66 < 0.0000000000000002 ***
## PAY_2                  0.2357     0.0154   15.32 < 0.0000000000000002 ***
## PAY_3                  0.1563     0.0158    9.86 < 0.0000000000000002 ***
## SEXMale                0.3504     0.0222   15.79 < 0.0000000000000002 ***
## EDUCATIONHigh.School   0.1684     0.0323    5.21    0.000000188804815 ***
## EDUCATIONOther        -1.0884     0.2507   -4.34    0.000014193434278 ***
```

```
## EDUCATIONUniversity   -0.1393     0.0252   -5.52    0.000000033787406 ***
## EDUCATIONUnkown       -0.9001     0.1364   -6.60    0.000000000041776 ***
## BILL_AMT_SUM          -0.2484     0.0142  -17.45 < 0.0000000000000002 ***
## PAY_5                  0.1267     0.0164    7.72    0.000000000000011 ***
## MARRIAGEOther          0.2993     0.0880    3.40              0.00067 ***
## MARRIAGESingle        -0.2134     0.0232   -9.19 < 0.0000000000000002 ***
## PAY_4                  0.1406     0.0163    8.63 < 0.0000000000000002 ***
## PAY_6                  0.1186     0.0155    7.67    0.000000000000017 ***
## AGE                    0.0803     0.0118    6.79    0.000000000011122 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 57440  on 41806  degrees of freedom
## Residual deviance: 48420  on 41790  degrees of freedom
## AIC: 48454
##
## Number of Fisher Scoring iterations: 5
```

Predict Test Set

```
logpredSMOTESTANDARD<- predict(FitLogModelSMOTESTANDARD,SMOTESTANDARTDATASET.test[-12],type = "response
```

37.1.5 Converting Prob to Classes

```
ypredlogSMOTESTANDARD <- as.factor(ifelse(logpredSMOTESTANDARD > 0.5,"Yes","No"))
```

37.1.6 Confusion Matrix

```
confusionMatrix( SMOTESTANDARTDATASET.test$DEFAULT,ypredlogSMOTESTANDARD,positive = "Yes", mode = "every
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  4824 3139
##        Yes 2352 7602
##
##                Accuracy : 0.694
##                  95% CI : (0.687, 0.7)
##     No Information Rate : 0.599
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.373
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.708
##             Specificity : 0.672
##          Pos Pred Value : 0.764
##          Neg Pred Value : 0.606
##               Precision : 0.764
##                  Recall : 0.708
```

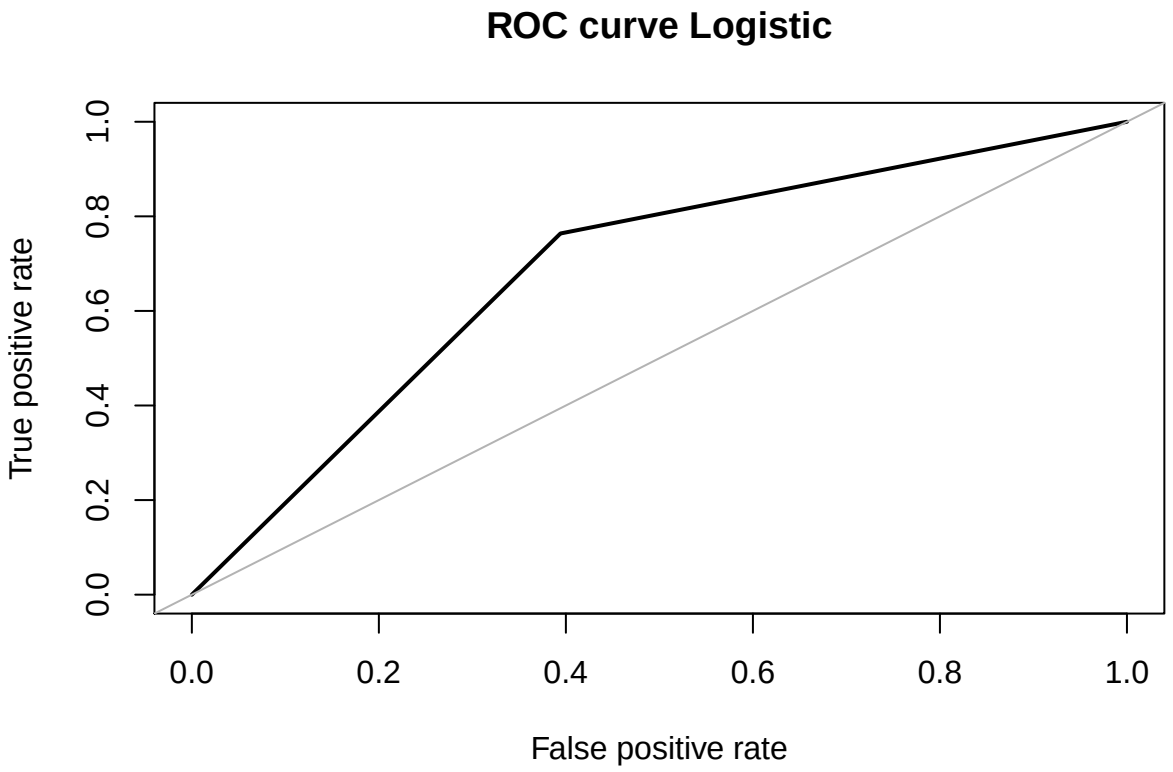
```
##                      F1 : 0.735
##              Prevalence : 0.599
##          Detection Rate : 0.424
##    Detection Prevalence : 0.556
##       Balanced Accuracy : 0.690
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity, Specificity, Precision and Accuracy scores are GOOD. Let's try to iprove using other models

37.1.7 Receiver Operating Characteristic Curve (ROC)

```
ROCtestSMOTEStand<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,ypredlogSMOTESTANDARD, main="ROC curve L
```



37.2 KNN Model

```
knnCaretSMOTESTANDARD
```

```
## k-Nearest Neighbors
##
## 41807 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
```

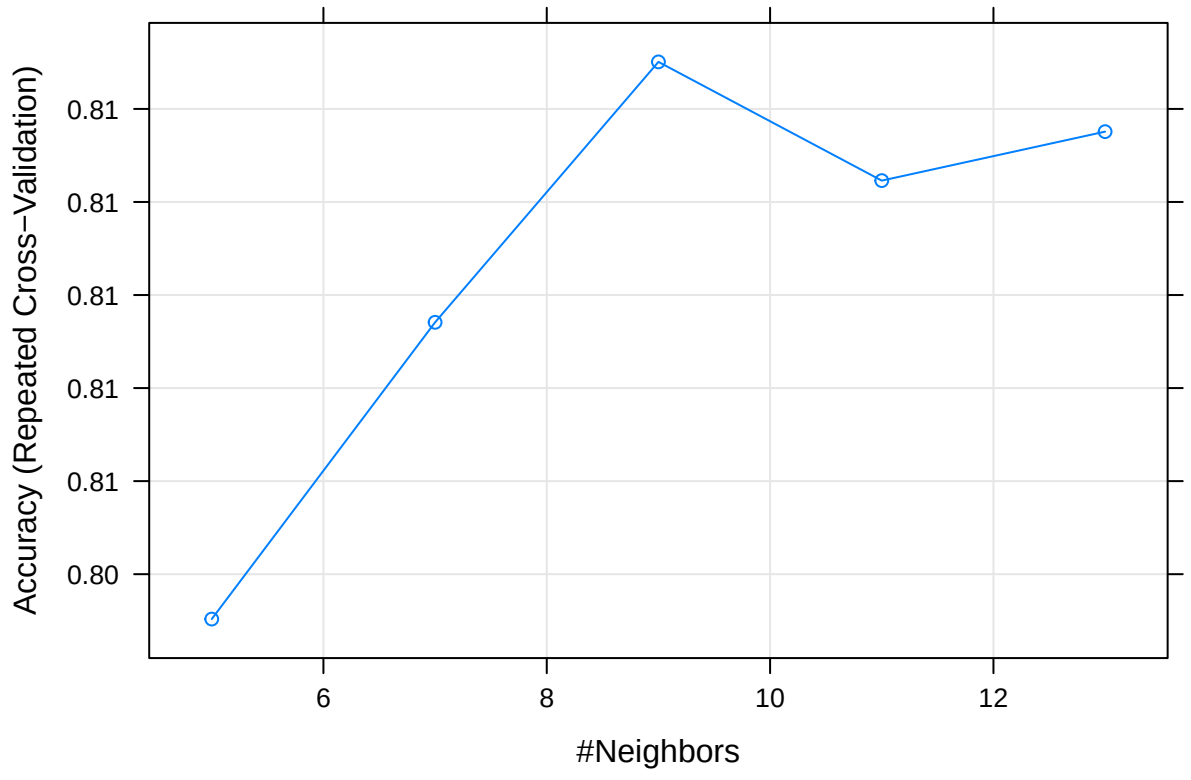
```
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 37627, 37626, 37626, 37626, 37625, 37627, ...
## Resampling results across tuning parameters:
##
##   k   Accuracy  Kappa
##    5  0.80      0.61
##    7  0.81      0.61
##    9  0.81      0.62
##   11  0.81      0.62
##   13  0.81      0.62
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 9.
```

Findings

- K 13 is Optimal K value used in this Model

37.2.1 #Plotting Number of Neighbours Vs accuracy (based on repeated cross validation)

```
plot(knnCaretSMOTESTANDARD)
```



37.2.2 Let's predict for Test Data

```
testCaretSMOTESTANDARD <- predict(knnCaretSMOTESTANDARD, newdata = SMOTESTANDARTDATASET.test)
```

37.2.3 Confusion Matrix

```
confusionMatrix( SMOTESTANDARTDATASET.test$DEFAULT,testCaretSMOTESTANDARD, positive = "Yes", mode = "ev
```

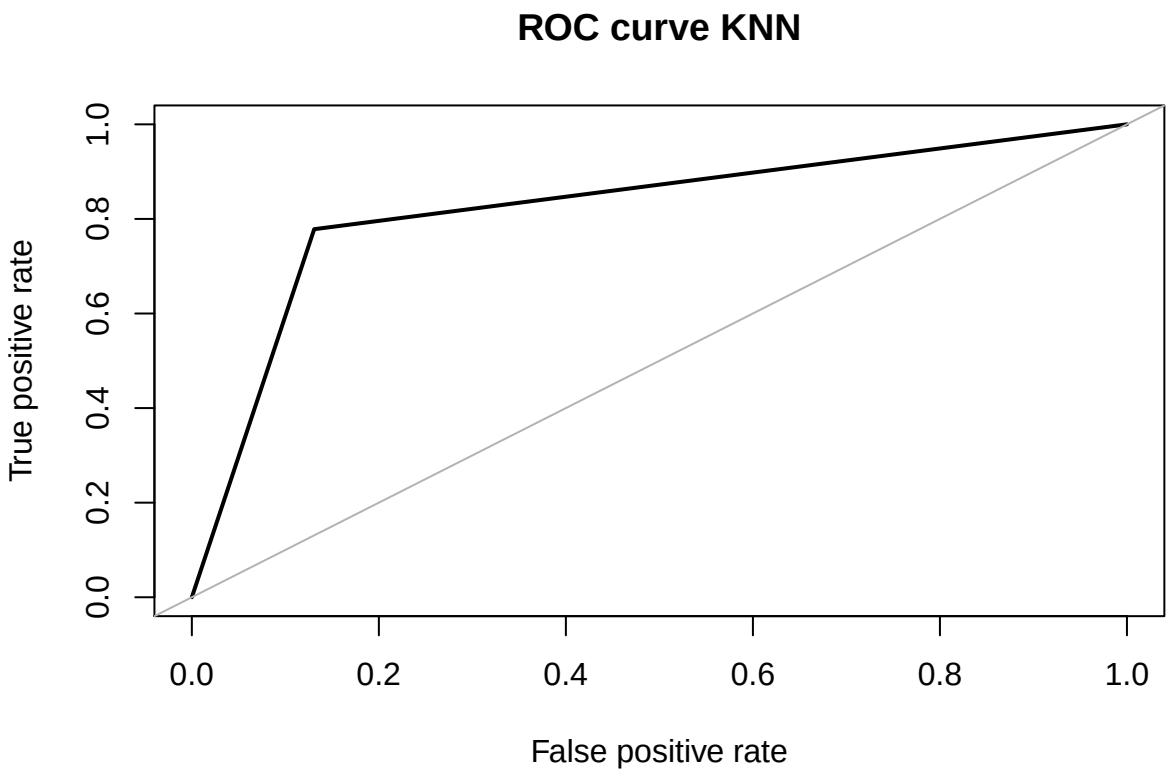
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6921 1042
##        Yes 2207 7747
##
##                Accuracy : 0.819
##                  95% CI : (0.813, 0.824)
##     No Information Rate : 0.509
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.638
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.881
##             Specificity : 0.758
##          Pos Pred Value : 0.778
##          Neg Pred Value : 0.869
##               Precision : 0.778
##                  Recall : 0.881
##                      F1 : 0.827
##              Prevalence : 0.491
##          Detection Rate : 0.432
##    Detection Prevalence : 0.556
##       Balanced Accuracy : 0.820
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity, Specificity, Precision and Accuracy scores are GOOD. Let's try to improve using other models. As we are more keen on Sensitivity for ths project. Sensitivity for this model is GOOD. Let's try and improve using other models.

37.2.4 Receiver Operating Characteristic Curve (ROC)

```
ROCtestSMOTESTANDARD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,testCaretSMOTESTANDARD, main="ROC cu
```



37.2.5 Let's start building Naive Bayes with Cross Validation using Caret

```
nb_standarizedSMOTESTANDARD<-naiveBayes(x=SMOTESTANDARTDATASET.train[,-12], y=SMOTESTANDARTDATASET.trai
pred_nbSMOTESTANDARD<-predict(nb_standarizedSMOTESTANDARD,newdata = SMOTESTANDARTDATASET.test[,-12])
confusionMatrix( SMOTESTANDARTDATASET.test$DEFAULT,pred_nbSMOTESTANDARD,positive = "Yes", mode = "every
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6473 1490
##        Yes 2649 7305
##
##                Accuracy : 0.769
##                  95% CI : (0.763, 0.775)
##     No Information Rate : 0.509
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.539
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.831
##             Specificity : 0.710
```

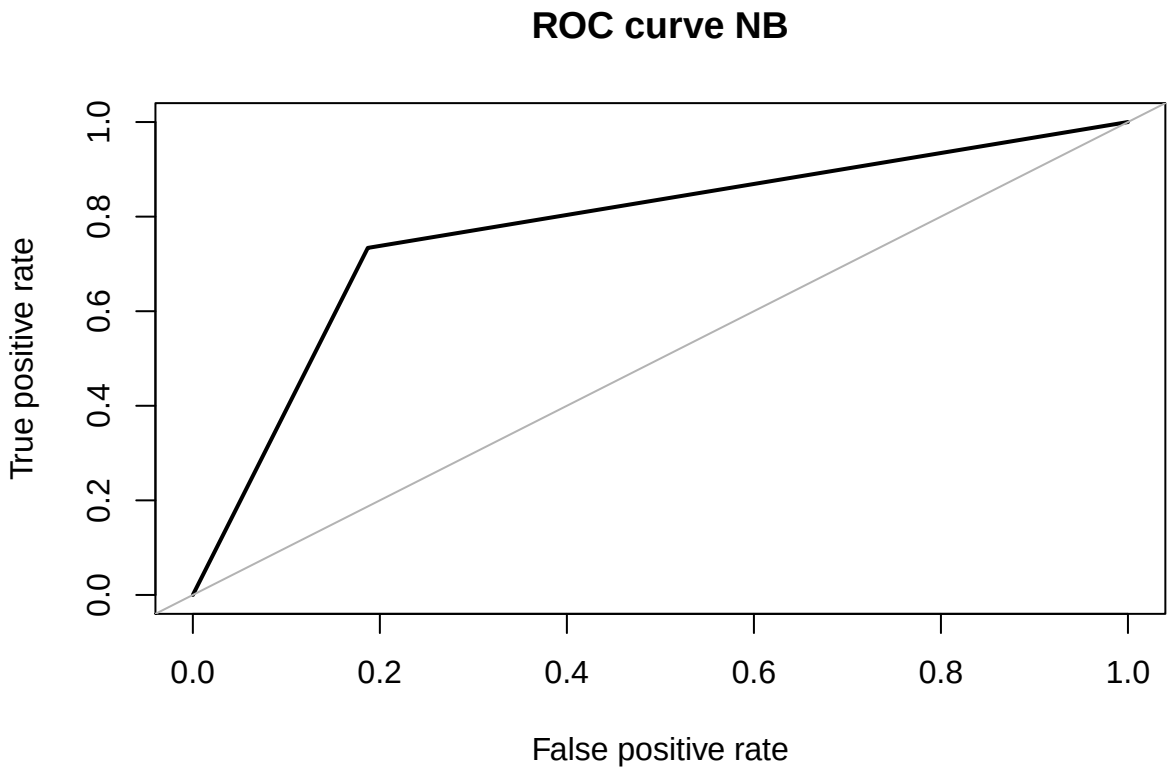
```
##          Pos Pred Value : 0.734
##          Neg Pred Value : 0.813
##               Precision : 0.734
##                  Recall : 0.831
##                      F1 : 0.779
##              Prevalence : 0.491
##          Detection Rate : 0.408
##    Detection Prevalence : 0.556
##       Balanced Accuracy : 0.770
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity, Specificity, Precision and Accuracy scores are GOOD. Let's try to improve using other models. As we are more keen on Sensitivity for ths project. Sensitivity for this model is GOOD. Let's try and improve using other models.

37.2.6 Receiver Operating Characteristic Curve (ROC)

```
ROCtestSMOTESTANDARDD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,pred_nbSMOTESTANDARD, main="ROC cur
```



38 CART with Tuning Parameters

```
model_marsSMOTESTANDARD
```

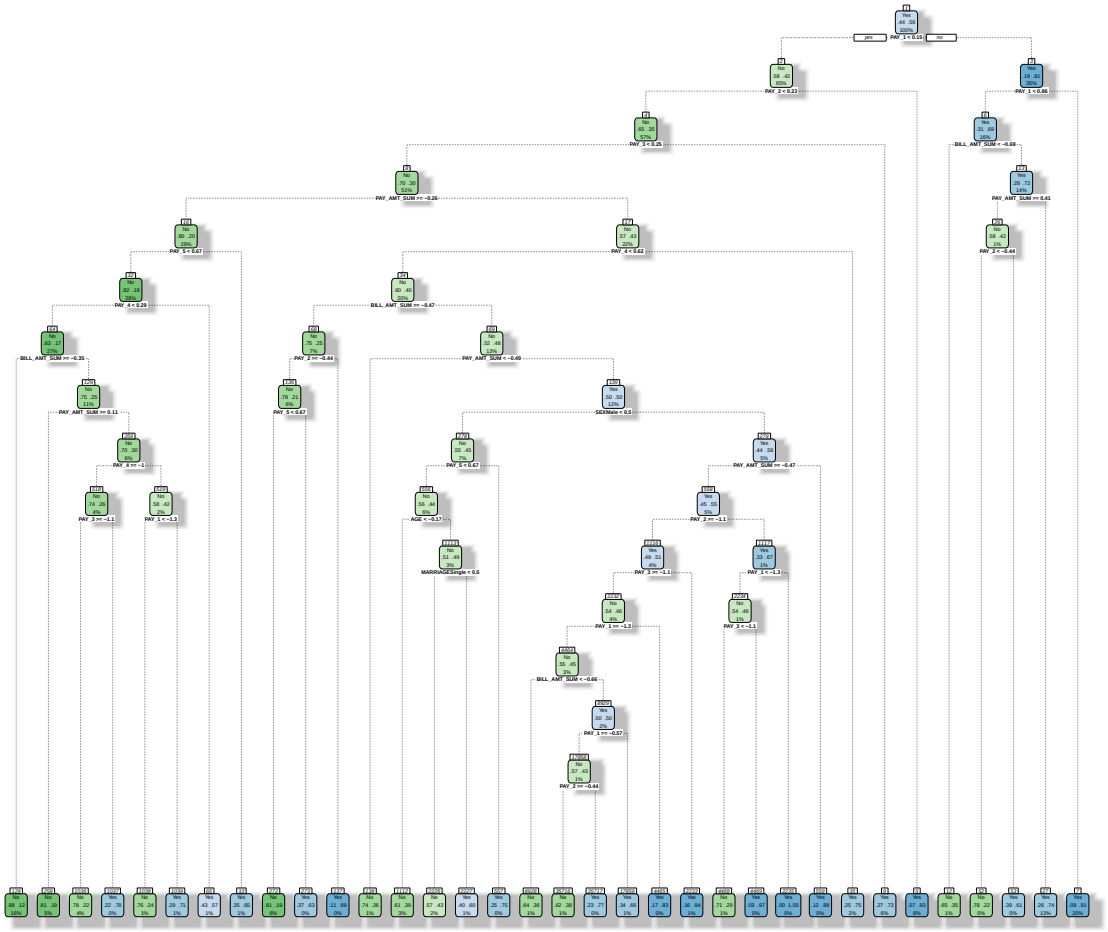
```
## CART
##
## 41807 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 33446, 33445, 33445, 33446, 33446
## Resampling results across tuning parameters:
##
##   cp      ROC   Sens  Spec
##   0.0016  0.85  0.75  0.83
##   0.0019  0.85  0.76  0.82
##   0.0019  0.85  0.76  0.82
##   0.0021  0.85  0.76  0.82
##   0.0024  0.84  0.75  0.82
##   0.0025  0.84  0.75  0.82
##   0.0027  0.84  0.75  0.82
##   0.0034  0.83  0.74  0.82
##   0.0040  0.83  0.74  0.82
##   0.0054  0.82  0.73  0.81
##   0.0090  0.79  0.80  0.73
##   0.0107  0.79  0.80  0.73
##   0.0661  0.76  0.82  0.67
##   0.1482  0.71  0.85  0.56
##   0.2403  0.57  0.34  0.80
##
## ROC was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.0016.
```

Findings

- Nprune of cp = 0.001291642 is applied to the model

38.1 Plot Tree

```
fancyRpartPlot(model_marsSMOTESTANDARD$finalModel)
```



Rattle 2020−Feb−12 07:30:36 Abhay

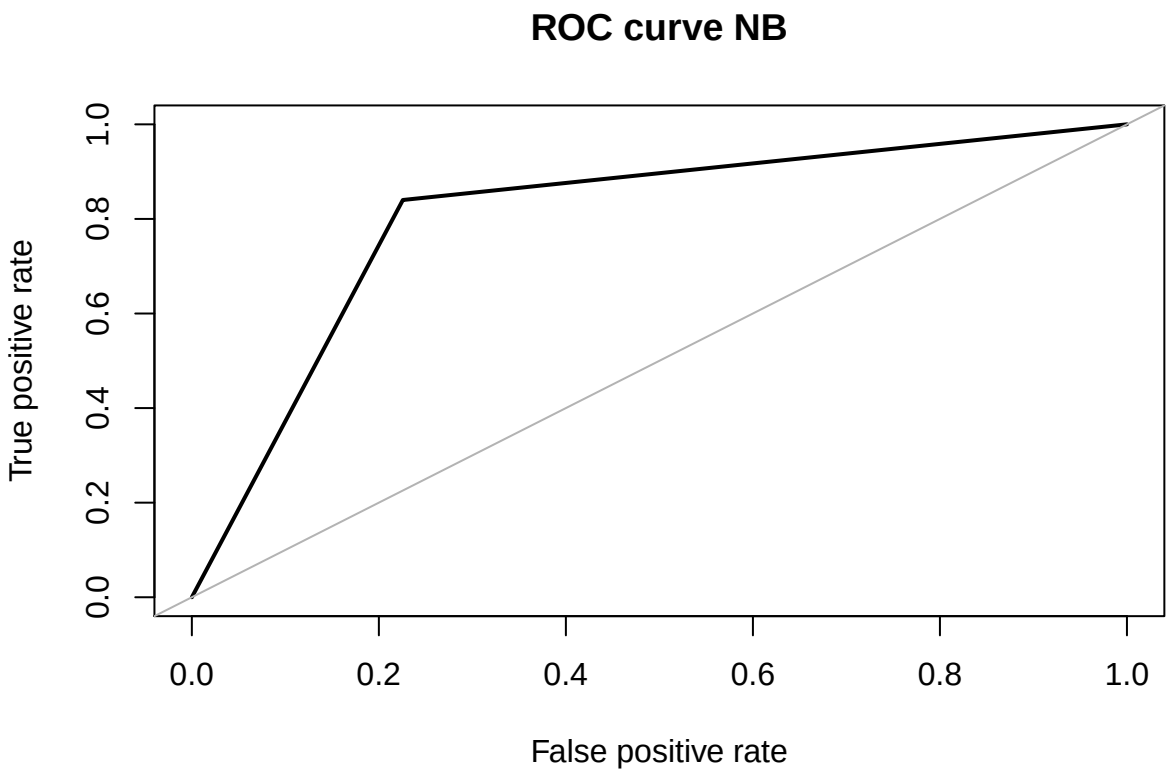
38.1.1 Predict on testData and Compute the confusion matrix

```
confusionMatrix(reference = SMOTESTANDARTDATASET.test$DEFAULT, data = predictedSMOTESTANDARD , mode='ev
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6166 1592
##        Yes 1797 8362
##
##                Accuracy : 0.811
##                  95% CI : (0.805, 0.817)
##     No Information Rate : 0.556
##     P-Value [Acc > NIR] : < 0.0000000000000002
##
##                   Kappa : 0.616
##
##  Mcnemar's Test P-Value : 0.000458
##
##             Sensitivity : 0.840
##             Specificity : 0.774
##          Pos Pred Value : 0.823
##          Neg Pred Value : 0.795
##               Precision : 0.823
##                  Recall : 0.840
##                      F1 : 0.832
##              Prevalence : 0.556
##          Detection Rate : 0.467
##    Detection Prevalence : 0.567
##       Balanced Accuracy : 0.807
##
##        'Positive' Class : Yes
##
```

38.1.2 Receiver Operating Characteristic Curve (ROC)

```
ROCtestSMOTESTANDARDDCART<- roc.curve( SMOTESTANDARTDATASET.test$DEFAULT,predictedSMOTESTANDARD, main="
```



Findings

- Let's improve more using Random Forest and Boosting

38.2 Random FOrest with tuning Parameters

```
model_rfSMOTESTANDARD
```

```
## Random Forest
##
## 41807 samples
##    13 predictor
##     2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 33446, 33445, 33445, 33446, 33446
## Resampling results across tuning parameters:
##
##   mtry  ROC   Sens  Spec
##    2    0.93  0.85  0.87
##    5    0.95  0.90  0.88
##    9    0.95  0.90  0.88
##   13    0.95  0.90  0.88
##   17    0.95  0.90  0.88
##
## ROC was used to select the optimal model using the largest value.
```

```
## The final value used for the model was mtry = 5.
```

Findings

- mtry of 5 is applied as optimal mtry.

38.2.1 Predict RF

```
predictedRFSMOTESTANDARD <- predict(model_rfSMOTESTANDARD, SMOTESTANDARTDATASET.test)
```

```
confusionMatrix(reference = SMOTESTANDARTDATASET.test$DEFAULT, data = predictedRFSMOTESTANDARD, mode='e
```

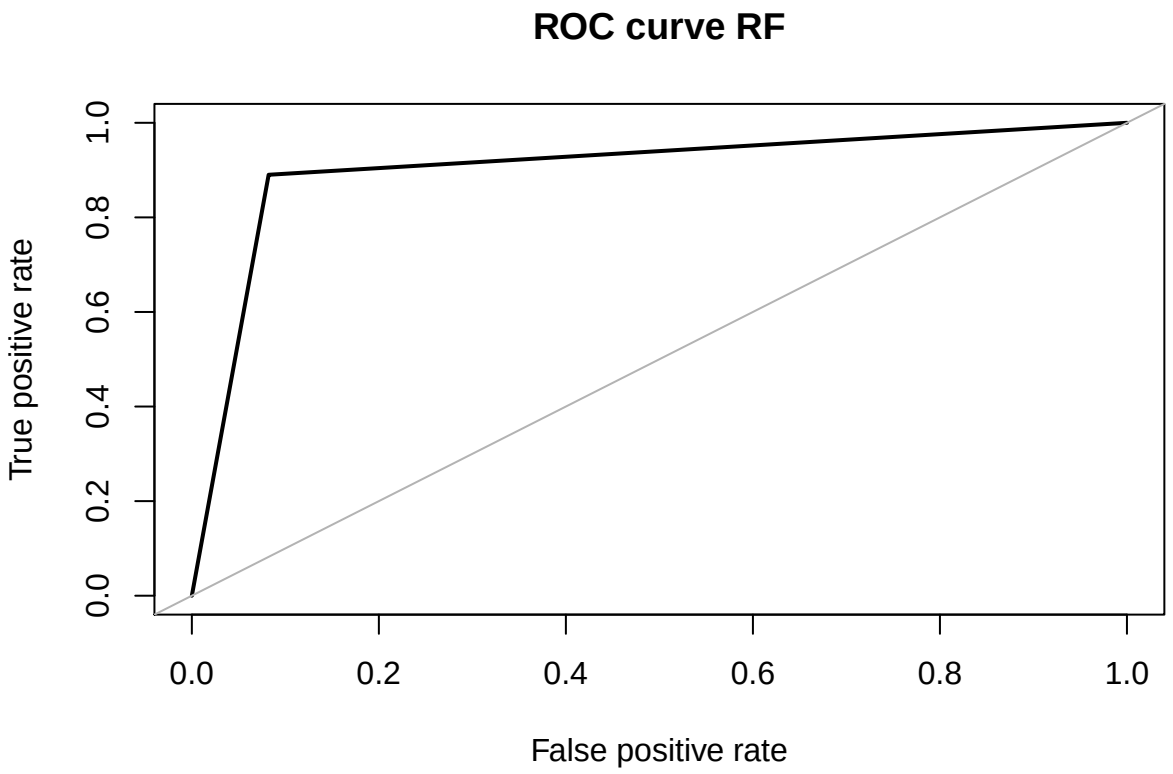
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  7309 1094
##        Yes  654 8860
##
##                Accuracy : 0.902
##                  95% CI : (0.898, 0.907)
##     No Information Rate : 0.556
##     P-Value [Acc > NIR] : <0.0000000000000002
##
##                   Kappa : 0.804
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.890
##             Specificity : 0.918
##          Pos Pred Value : 0.931
##          Neg Pred Value : 0.870
##               Precision : 0.931
##                  Recall : 0.890
##                      F1 : 0.910
##              Prevalence : 0.556
##          Detection Rate : 0.495
##    Detection Prevalence : 0.531
##       Balanced Accuracy : 0.904
##
##        'Positive' Class : Yes
##
```

Findings

- Excellent Model

38.2.2 Receiver Operating Characteristic Curve (ROC)

```
ROCtestSMOTESTANDARDDDDDD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,predictedRFSMOTESTANDARD, main=
```



38.3 BAGGING

38.3.1 BAGGING

38.3.2 Predicting Bagging

```
## [1] "factor"
```

38.3.3 Confusion Matrix

```
confusionMatrix(SMOTESTANDARTDATASET.test$DEFAULT, BaggingSmoteStandardize$class,positive = "Yes", mode
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6476 1487
##        Yes 2637 7317
##
##                Accuracy : 0.77
##                  95% CI : (0.764, 0.776)
##     No Information Rate : 0.509
##     P-Value [Acc > NIR] : <0.0000000000000002
##
```

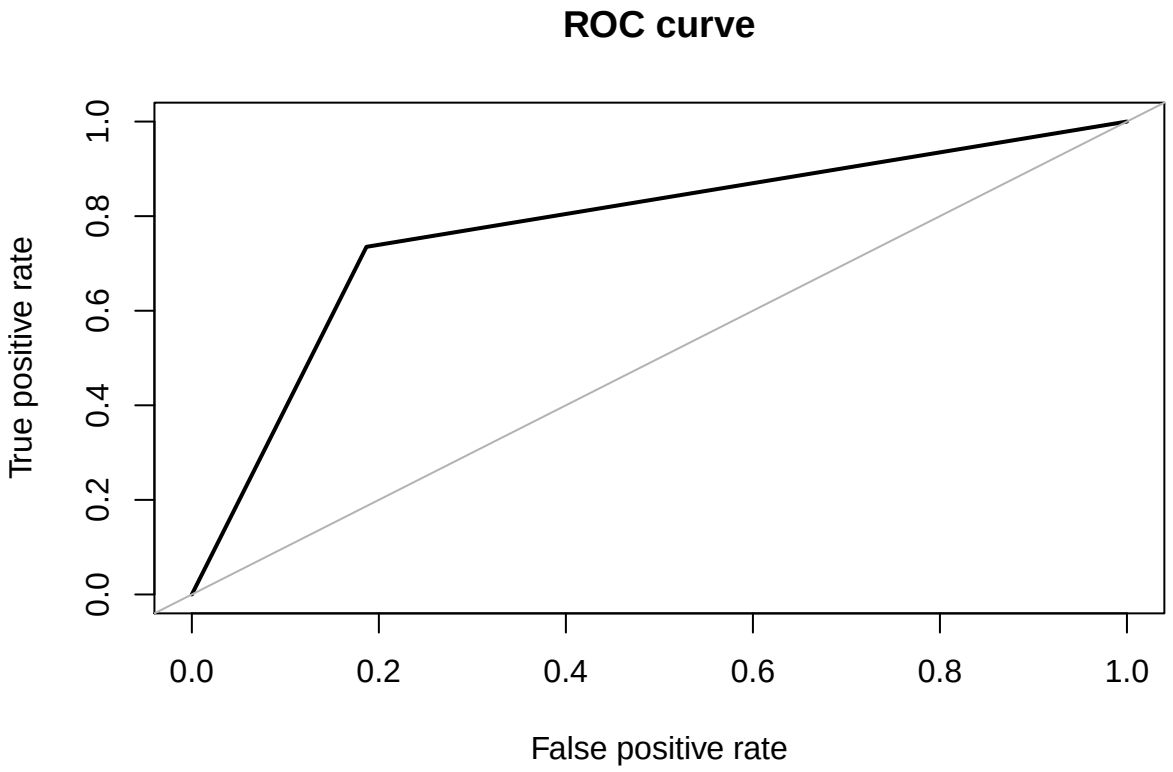
```
##                   Kappa : 0.541
##
##  Mcnemar's Test P-Value : <0.0000000000000002
##
##             Sensitivity : 0.831
##             Specificity : 0.711
##          Pos Pred Value : 0.735
##          Neg Pred Value : 0.813
##               Precision : 0.735
##                  Recall : 0.831
##                      F1 : 0.780
##              Prevalence : 0.491
##          Detection Rate : 0.408
##    Detection Prevalence : 0.556
##       Balanced Accuracy : 0.771
##
##        'Positive' Class : Yes
##
```

Findings

- Sensitivity and other scores of RF is better. Let's try Boosting

38.3.4 Receiver Operating Characteristic Curve (ROC)

```
ROCtestSMOTESTANDARddded<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,BaggingSmoteStandardize$class, ma
```



38.4 Boosting

38.4.1 adaBoost with adabag

```
adaboostmodelSmoteStandard <- boosting(DEFAULT~., data=SMOTESTANDARTDATASET.train, boos=TRUE, mfinal=50)
```

38.4.2 Print Tree

```
print(names(adaboostmodelSmoteStandard))
```

```
## [1] "formula"    "trees"      "weights"    "votes"      "prob"
## [6] "class"      "importance" "terms"      "call"
```

```
print(adaboostmodelSmoteStandard$trees[1])
```

```
## [[1]]
## n= 41807
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
## 1) root 41807 19000 Yes (0.443 0.557)
##   2) PAY_1< 0.15 27085 11000 No (0.580 0.420)
##     4) PAY_2< 0.23 23880  8400 No (0.649 0.351)
##       8) PAY_3< 0.59 21229  6400 No (0.698 0.302) *
##       9) PAY_3>=0.59 2651   660 Yes (0.251 0.749) *
##     5) PAY_2>=0.23 3205   230 Yes (0.071 0.929) *
##   3) PAY_1>=0.15 14722  2800 Yes (0.190 0.810) *
```

Predict on test set

```
##                Observed Class
## Predicted Class   No  Yes
##             No  6546 1374
##             Yes 1417 8580
```

```
## [1] 0.16
```

Converting to factors

```
predAdaBoostSMOTEStandard$class<- as.factor(predAdaBoostSMOTEStandard$class)
```

38.4.3 Confusion Matrix and evaluation

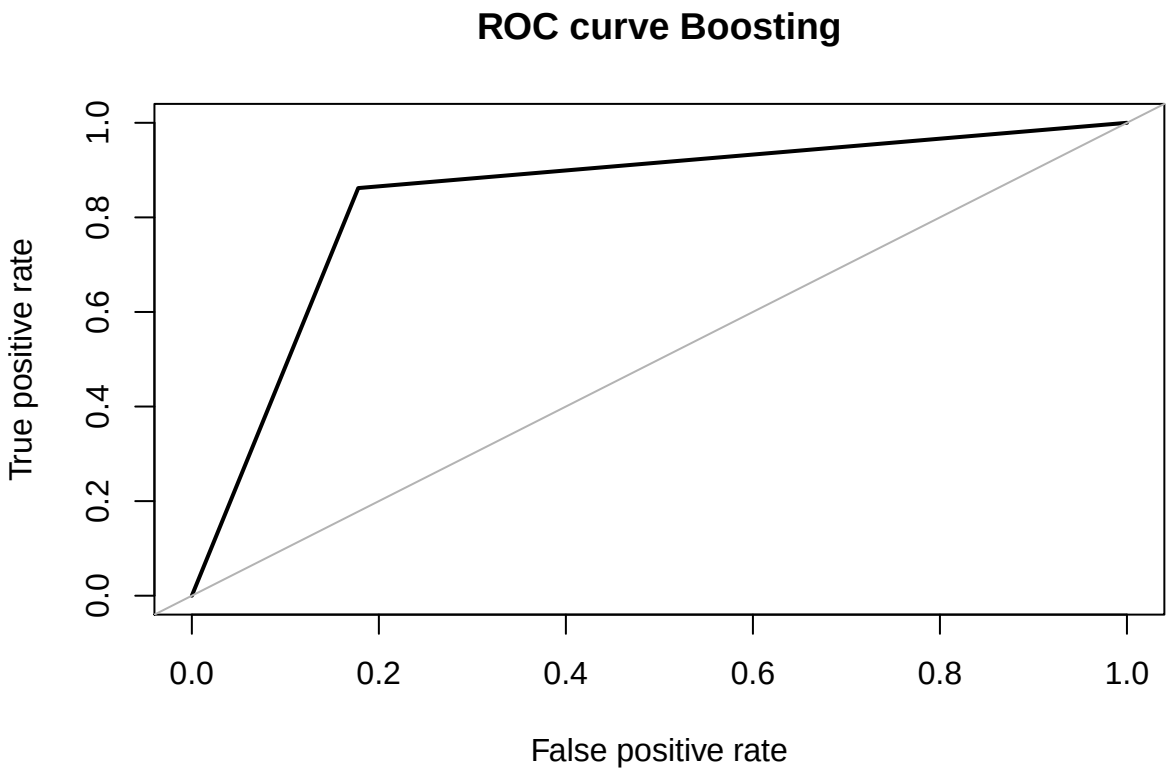
```
confusionMatrix(SMOTESTANDARTDATASET.test$DEFAULT,predAdaBoostSMOTEStandard$class,mode="everything",posi
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  6546 1417
##        Yes 1374 8580
##
##                Accuracy : 0.844
##                  95% CI : (0.839, 0.85)
##     No Information Rate : 0.558
##     P-Value [Acc > NIR] : <0.0000000000000002
```

```
##
##                   Kappa : 0.684
##
##  Mcnemar's Test P-Value : 0.427
##
##             Sensitivity : 0.858
##             Specificity : 0.827
##          Pos Pred Value : 0.862
##          Neg Pred Value : 0.822
##               Precision : 0.862
##                  Recall : 0.858
##                      F1 : 0.860
##              Prevalence : 0.558
##          Detection Rate : 0.479
##    Detection Prevalence : 0.556
##       Balanced Accuracy : 0.842
##
##        'Positive' Class : Yes
##
```

38.4.4 Receiver Operating Characteristic Curve (ROC)

```
ROCtestlogSMOTESTANDRDDD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,predAdaBoostSMOTEStandard$class,
```



39 Model Comparison

39.1 Have built multiple models on 4 datasets

1) Logistic Regression
2) Naive Bayes
3) KNN
4) CART
5) Random Forest
6) Bagging
7) Boosting

39.2 Datasets

1) Feature Engineered is Normal Dataset
2) SMOTE
3) Standardized Dataset
4) SMOTE Standardized Dataset

39.3 Regular Dataset

Regular Dataset	Logistic Regression	KNN	Naive Bayes	CART	Random Forest	Bagging	Boosting
Accuracy	81.40%	77.10%	80.20%	82.30%	82.40%	82.30%	82.20%
Sensitivity	75.20%	42.19%	56.80%	36.11%	35.96%	72.38%	68.40%
Specificity	81.89%	78.93%	85.10%	95.46%	95.56%	83.45%	84.10%
Precision	23.91%	9.49%	44.25%	69.33%	69.72%	32.65%	36.60%
F1	36.28	15.50	49.75%	47.749%	47.45	45.00%	47.70%

39.4 SMOTE

SMOTE Dataset	Logistic Regression	KNN	Naive Bayes	CART	Random Forest	Bagging	Boosting
Accuracy	69.20%	70.70%	76.40%	78.00%	89.90%	82.30%	60.40%
Sensitivity	70.80%	76.40%	73.20%	79.40%	88.60%	72.38%	68.40%
Specificity	66.69%	65.10%	80.30%	76.20%	91.50%	83.45%	54.20%
Precision	75.59%	68.40%	82.30%	80.70%	92.90%	32.65%	36.60%
F1	73.30%	72.20%	77.50%	80.00%	90.70%	45.00%	47.70%

39.5 Standardize Dataset

Standardize Dataset	Logistic Regression	KNN	Naive Bayes	CART	Random Forest	Bagging	Boosting
Accuracy	81.40%	81.60%	80.20%	82.30%	82.40%	82.30%	82.20%

Standardize Dataset	Logistic Regression	KNN	Naive Bayes	CART	Random Forest	Bagging	Boosting
Sensitivity	75.20%	64.95%	56.80%	36.11%	36.21%	72.38%	69.96%
Specificity	81.89%	83.88%	85.10%	95.46%	95.56%	83.45%	83.76%
Precision	23.91%	36.11%	44.25%	69.33%	69.86%	32.65%	34.51%
F1	36.28%	46.42%	49.75%	47.49%	47.70%	45.00%	46.22%

39.6 SMOTE and Standardized

SMOTE and Standardize	Logistic Regression	KNN	Naive Bayes	CART	Random Forest	Bagging	Boosting
Accuracy	69.20%	81.70%	76.40%	78.20%	90.00%	76.80%	85.20%
Sensitivity	70.80%	87.90%	82.30%	79.40%	88.60%	82.80%	86.10%
Specificity	66.90%	75.70%	70.60%	76.70%	91.60%	70.90%	84.10%
Precision	75.90%	77.70%	73.20%	81.10%	93.30%	73.50%	87.60%
F1	73.30%	82.50%	77.50%	80.20%	90.70%	77.90%	86.80%

40 Conclusion and Recommendation

1) Accuracy, Sensitivity, Specificity and Precision of RANDOM FOREST on SMOTE + STANDARDIZED DATASET is the highest. Would recomend using Random Forest Model with SMOTE and Standardized dataset
2) A Taiwan-based bank wants to improve their prediction of defaults of their customers, as well as identify the patterns that determine this likelihood. This would help the bank decide whether to issue the credit card or not. Also, fix credit limt and risk type to the customer and avoid future defaults.
3) To lower the risk of default, must be very cautious on clients payment behaviour.
4) More cautious of High School level clients.
5) Marketing campaign should be aimed at clients' age from 25 to 35.
6) As the model can predict defaulters. Bank rep can keep an eye on customers who are likely to default and contact them immediately as they default 1st payment. Ensure it doesn't become NPA.
7) Spread awareness regarding good credit history/score. Advantages of good credit score and disadvantages of bad credit score should be communicated with borrower on regular bases.
8) Provide Cashback or other benefits for people making regular payments.
9) Those with higher Pay Amount Sum are less likely to default. Encourage these customers to make referrals.
10) Provide facilities like collecting cash from home for customers.

Taiwan Bank can now predict customer(applicants) who are going to default using Random Forest model built. Sensitivity (also called the true positive rate, the recall, or probability of detection in some fields) measures the proportion of actual positives that are correctly identified as such (e.g., the percentage of customers default who are correctly identified as having to default)

Taiwan Bank can now mitigate the Default Risk more efficiently using the Random forest built.

41 Appendix

```
knitr::opts_chunk$set(echo = TRUE)

library(knitr)
library(readxl)
library(DataExplorer)
library(memisc)
library(funModeling)
library(cowplot)
library(MASS)
library(DMwR)
library(caTools)
library(DataExplorer)
library(ggplot2)
library(caTools)
library(skimr)
library(caret)
library(cowplot)
library(caTools)
library(ROSE)
library(ROCR)
library(MLmetrics)
library(MASS)
library(class)
library(e1071)
library(car)
library(ROSE)
library(MASS)
library(pROC)
library(e1071)
library(class)
library(lattice)
library(klaR)
library(ipred)
library(rpart)
library(xgboost)
library(adabag)
library(pROC)
library(rattle)
library(parallel)
library(doParallel)
clusterforspeed <- makeCluster(detectCores() - 1) ## convention to leave 1 core for OS
registerDoParallel(clusterforspeed)
setwd("Z:\\Projects\\Capstone")
getwd()
myrawdata <- read_excel("Taiwan-Customer defaults.xls", skip = 1)
myrawdata<- as.data.frame(myrawdata)
head(myrawdata)
dim(myrawdata)
introduce(myrawdata)

a <- c(3,4,5,25)
```

```
converteddata <- myrawdata
for (i in a) {
  converteddata[,i] <- as.factor(converteddata[,i])
}
introduce(converteddata)
datadetails <-df_status(converteddata)
names(converteddata)[names(converteddata) == "PAY_0"] <- "PAY_1"

names(converteddata)[names(converteddata) == "default payment next month"] <- "DEFAULT "

converteddata$DEFAULT <- as.factor(ifelse(converteddata$DEFAULT == 1, "Yes", "No"))
converteddata$SEX <- as.factor(ifelse(converteddata$SEX == 1, "Male", "Female"))
converteddata$MARRIAGE <- as.factor(ifelse(converteddata$MARRIAGE == 1, "Married",
                                ifelse(converteddata$MARRIAGE == 2, "Single", "Other")))
converteddata$EDUCATION <- as.factor(ifelse(converteddata$EDUCATION == 1, "Graduate.School",
                                 ifelse(converteddata$EDUCATION == 2, "University",
                                        ifelse(converteddata$EDUCATION == 3, "High.School",
                                               ifelse(converteddata$EDUCATION == 4, "Other", "Unkown")))
table(converteddata$MARRIAGE)

table(converteddata$SEX)

table(converteddata$EDUCATION)
converteddata<- converteddata[,c(-25,-1)]
backupdata <- converteddata
plot_missing(converteddata)
plot_correlation(converteddata)
a <- ggplot(converteddata) +
 aes(x = DEFAULT, y = prop.table(stat(count)),fill = DEFAULT,label = scales::percent(prop.table(stat(co
 geom_bar() +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Default Split Percentage") +
 theme(plot.title = element_text(hjust = 0.5))+
geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)

a
b <- ggplot(converteddata) +
 aes(x = SEX, y = prop.table(stat(count)),fill = SEX,label = scales::percent(prop.table(stat(count)))) +
 geom_bar() +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "SEX Split Percentage") +
 theme(plot.title = element_text(hjust = 0.5))+
geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)
b
c <- ggplot(converteddata) +
 aes(x = EDUCATION, y = prop.table(stat(count)),fill = EDUCATION,label = scales::percent(prop.table(sta
 geom_bar() +
```

```
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "EDUCATION Split Percentage") +
 theme(plot.title = element_text(hjust = 0.5))+
geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)

c
d <- ggplot(converteddata) +
 aes(x = MARRIAGE, y = prop.table(stat(count)),fill = MARRIAGE,label = scales::percent(prop.table(stat(
 geom_bar() +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "MARRIAGE Split Percentage") +
 theme(plot.title = element_text(hjust = 0.5))+
geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)
d
e <- ggplot(converteddata) +
 aes(x = LIMIT_BAL) +
 geom_histogram(bins = 20L, fill = "#a6cee3") +
 labs(title = "Histogram of Limit Balance") +
 theme(plot.title = element_text(hjust = 0.5))




f <- ggplot(converteddata) +
 aes(x = "", y = LIMIT_BAL) +
 geom_boxplot(fill = "#a6cee3") +
 labs(title = "Box Plot of Limit Balance") +
 theme(plot.title = element_text(hjust = 0.5))


plot_grid(e,f)

summary(converteddata$LIMIT_BAL)
sd(converteddata$LIMIT_BAL)

g <- ggplot(converteddata) +
 aes(x = AGE) +
 geom_histogram(bins = 20L, fill = "#a6cee3") +
 labs(title = "Histogram of AGE") +
 theme(plot.title = element_text(hjust = 0.5))




h <- ggplot(converteddata) +
```

```
 aes(x = "", y = AGE) +
 geom_boxplot(fill = "#a6cee3") +
 labs(title = "Box Plot of AGE") +
 theme(plot.title = element_text(hjust = 0.5))


plot_grid(g,h)

summary(converteddata$AGE)
sd(converteddata$AGE)
k <- ggplot(converteddata) +
 aes(x = SEX,  y = prop.table(stat(count)),fill = DEFAULT,label = scales::percent(prop.table(stat(count)
 geom_bar(position = "dodge") +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "SEX VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))+geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)
k
SEXtable <- table(converteddata$SEX,converteddata$DEFAULT)

SEXtable
SEXchi<- chisq.test(SEXtable)

SEXchi
l<-  ggplot(converteddata) +
 aes(x = EDUCATION,  y = prop.table(stat(count)),fill = DEFAULT,label = scales::percent(prop.table(stat
 geom_bar(position = "dodge") +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "EDUCATION VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))+geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)

l
EducationTable <- table(converteddata$EDUCATION,converteddata$DEFAULT)

EducationTable
Educhisq<- chisq.test(EducationTable)

Educhisq
m<-  ggplot(converteddata) +
 aes(x = MARRIAGE,  y = prop.table(stat(count)),fill = DEFAULT,label = scales::percent(prop.table(stat(
 geom_bar(position = "dodge") +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "MARRIAGE VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))+geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)
```

```
m
MarriageTable <- table(converteddata$DEFAULT,converteddata$MARRIAGE)

MarriageTable
MarriageChi<- chisq.test(MarriageTable)

MarriageChi

LD <- ggplot(converteddata) +
 aes(x = LIMIT_BAL, fill = DEFAULT) +
 geom_density(adjust = 1L) +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Density Plot LIMIT BAL VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))

dodge <- position_dodge(width = 0.9)

 Boxld <- ggplot(converteddata) +
 aes(x = "", y = LIMIT_BAL, fill = DEFAULT) +
 geom_violin(adjust = 1L, scale = "area") +geom_boxplot(width=0.1,position = dodge)+
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Violin Plot LIMIT BAL VS DEFAULT") +



 theme(plot.title = element_text(hjust = 0.5))



plot_grid(LD,Boxld)


AGEDen <- ggplot(converteddata) +
 aes(x = AGE, fill = DEFAULT) +
 geom_density(adjust = 1L) +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Density Plot AGE VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))

dodge <- position_dodge(width = 0.9)

 BOXAGE<- ggplot(converteddata) +
 aes(x = "", y = AGE, fill = DEFAULT) +
 geom_violin(adjust = 1L, scale = "area") +geom_boxplot(width=0.1,position = dodge)+
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Violin Plot AGE VS DEFAULT") +



 theme(plot.title = element_text(hjust = 0.5))



plot_grid(AGEDen ,BOXAGE)
```

```
BILL_AMT1Den <- ggplot(converteddata) +
 aes(x = BILL_AMT1, fill = DEFAULT) +
 geom_density(adjust = 1L) +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Density Plot BILL_AMT1 VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))

dodge <- position_dodge(width = 0.9)

 BOXBILL_AMT1<- ggplot(converteddata) +
 aes(x = "", y = BILL_AMT1, fill = DEFAULT) +
 geom_violin(adjust = 1L, scale = "area") +geom_boxplot(width=0.1,position = dodge)+
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Violin Plot BILL_AMT1 VS DEFAULT") +



 theme(plot.title = element_text(hjust = 0.5))



plot_grid(BILL_AMT1Den ,BOXBILL_AMT1)

BILL_AMT6Den <- ggplot(converteddata) +
 aes(x = BILL_AMT6, fill = DEFAULT) +
 geom_density(adjust = 1L) +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Density Plot BILL_AMT6 VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))

dodge <- position_dodge(width = 0.9)

 BOXBILL_AMT6<- ggplot(converteddata) +
 aes(x = "", y = BILL_AMT6, fill = DEFAULT) +
 geom_violin(adjust = 1L, scale = "area") +geom_boxplot(width=0.1,position = dodge)+
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Violin Plot BILL_AMT6 VS DEFAULT") +



 theme(plot.title = element_text(hjust = 0.5))



plot_grid(BILL_AMT6Den ,BOXBILL_AMT6)
PAY_AMT1Den <- ggplot(converteddata) +
 aes(x = PAY_AMT1, fill = DEFAULT) +
 geom_density(adjust = 1L) +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Density Plot PAY_AMT1 VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))

dodge <- position_dodge(width = 0.9)
```

```
 BOXPAY_AMT1<- ggplot(converteddata) +
 aes(x = "", y = PAY_AMT1, fill = DEFAULT) +
 geom_violin(adjust = 1L, scale = "area") +geom_boxplot(width=0.1,position = dodge)+
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Violin Plot PAY_AMT1 VS DEFAULT") +



 theme(plot.title = element_text(hjust = 0.5))



plot_grid(PAY_AMT1Den ,BOXPAY_AMT1)
plot_correlation((converteddata), type = "c")
ggplot(converteddata) +
 aes(x = SEX,  y = prop.table(stat(count)),fill = DEFAULT,label =scales::percent(prop.table(stat(count))
 geom_bar(position = "dodge") +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "GENDER & MARITAL STATUS VS DEFAULT") +
 theme(plot.title = element_text(hjust = 0.5))+geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3) +
 facet_wrap(vars(MARRIAGE))
a <- ggplot(converteddata) +
 aes(x = DEFAULT, y = prop.table(stat(count)),fill = DEFAULT,label = scales::percent(prop.table(stat(co
 geom_bar() +
 scale_fill_brewer(palette = "Pastel2") +
 labs(title = "Default Split Percentage") +
 theme(plot.title = element_text(hjust = 0.5))+
geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3)

a
Paytofactors <- c(6,7,8,9,10,11)
for (i in Paytofactors) {
  converteddata[,i] <- as.factor(converteddata[,i])
}
introduce(converteddata)
table(converteddata$MARRIAGE)

table(converteddata$SEX)

table(converteddata$EDUCATION)
ggplot(converteddata) +
 aes(x = AGE, y = LIMIT_BAL, colour = DEFAULT) +
 geom_point(size = 1L) +
 scale_color_brewer(palette = "Pastel2") +
 theme_minimal()
backupdata <- converteddata
outliersdata <- converteddata
```

```r
boxplot(outliersdata$AGE, main="BoxPlot of Age")
boxplot(outliersdata$LIMIT_BAL, main="Boxplot of Limit Balance")
summary(outliersdata$LIMIT_BAL)
benchlimitbalancehigh <- 240000 + (1.5 * IQR(outliersdata$LIMIT_BAL))

benchlimitbalanceLOW <- 50000 - (1.5 * IQR(outliersdata$LIMIT_BAL))

IQR(outliersdata$LIMIT_BAL)

benchlimitbalancehigh

benchlimitbalanceLOW
outliersdata$LIMIT_BAL[outliersdata$LIMIT_BAL > benchlimitbalancehigh] <- benchlimitbalancehigh
options(scipen=999)
boxplot(outliersdata$LIMIT_BAL, main = "Boxplot of Limit Balance")
summary(outliersdata$LIMIT_BAL)
boxplot(outliersdata$BILL_AMT1,main = "Boxplot of BillAMT 1")

summary(outliersdata$BILL_AMT1)
benchbalAMT1high <- 67091 + (1.5 * IQR(outliersdata$BILL_AMT1))

benchbalAMT1LOW <- 3559 - (1.5 * IQR(outliersdata$BILL_AMT1))

IQR(outliersdata$BILL_AMT1)

benchbalAMT1high

benchbalAMT1LOW
outliersdata$BILL_AMT1[outliersdata$BILL_AMT1 > benchbalAMT1high] <- benchbalAMT1high

outliersdata$BILL_AMT1[outliersdata$BILL_AMT1 < benchbalAMT1LOW] <- benchbalAMT1LOW


boxplot(outliersdata$BILL_AMT1, main = "Boxplot of BILL AMT1")
summary(outliersdata$BILL_AMT1)
boxplot(outliersdata$BILL_AMT2,main = "Boxplot of BillAMT 2")

summary(outliersdata$BILL_AMT2)
benchbalAMT2high <- 64006 + (1.5 * IQR(outliersdata$BILL_AMT2))

benchbalAMT2LOW <- 2985 - (1.5 * IQR(outliersdata$BILL_AMT2))

IQR(outliersdata$BILL_AMT2)

benchbalAMT2high

benchbalAMT2LOW
outliersdata$BILL_AMT2[outliersdata$BILL_AMT2 > benchbalAMT2high] <- benchbalAMT2high

outliersdata$BILL_AMT2[outliersdata$BILL_AMT2 < benchbalAMT2LOW] <- benchbalAMT2LOW


boxplot(outliersdata$BILL_AMT2, main = "Boxplot of BILL AMT2")
```

```r
summary(outliersdata$BILL_AMT2)
boxplot(outliersdata$BILL_AMT3,main = "Boxplot of BillAMT 3")

summary(outliersdata$BILL_AMT3)
benchbalAMT3high <- 60165 + (1.5 * IQR(outliersdata$BILL_AMT3))

benchbalAMT3LOW <- 2666 - (1.5 * IQR(outliersdata$BILL_AMT3))

IQR(outliersdata$BILL_AMT3)

benchbalAMT3high

benchbalAMT3LOW
outliersdata$BILL_AMT3[outliersdata$BILL_AMT3 > benchbalAMT3high] <- benchbalAMT3high

outliersdata$BILL_AMT3[outliersdata$BILL_AMT3 < benchbalAMT3LOW] <- benchbalAMT3LOW


boxplot(outliersdata$BILL_AMT3, main = "Boxplot of BILL AMT3")
summary(outliersdata$BILL_AMT3)
boxplot(outliersdata$BILL_AMT4,main = "Boxplot of BillAMT 4")

summary(outliersdata$BILL_AMT4)
benchbalAMT4high <- 54506 + (1.5 * IQR(outliersdata$BILL_AMT4))

benchbalAMT4LOW <- 2327- (1.5 * IQR(outliersdata$BILL_AMT4))

IQR(outliersdata$BILL_AMT4)

benchbalAMT4high

benchbalAMT4LOW
outliersdata$BILL_AMT4[outliersdata$BILL_AMT4 > benchbalAMT4high] <- benchbalAMT4high

outliersdata$BILL_AMT4[outliersdata$BILL_AMT4 < benchbalAMT4LOW] <- benchbalAMT4LOW


boxplot(outliersdata$BILL_AMT4, main = "Boxplot of BILL AMT4")
summary(outliersdata$BILL_AMT4)
boxplot(outliersdata$BILL_AMT5,main = "Boxplot of BillAMT 5")

summary(outliersdata$BILL_AMT5)
benchbalAMT5high <- 50191+ (1.5 * IQR(outliersdata$BILL_AMT5))

benchbalAMT5LOW <- 1763 - (1.5 * IQR(outliersdata$BILL_AMT5))

IQR(outliersdata$BILL_AMT5)

benchbalAMT5high

benchbalAMT5LOW
outliersdata$BILL_AMT5[outliersdata$BILL_AMT5 > benchbalAMT5high] <- benchbalAMT5high
```

```
outliersdata$BILL_AMT5[outliersdata$BILL_AMT5 < benchbalAMT5LOW] <- benchbalAMT5LOW


boxplot(outliersdata$BILL_AMT5, main = "Boxplot of BILL AMT5")
summary(outliersdata$BILL_AMT5)
boxplot(outliersdata$BILL_AMT6,main = "Boxplot of BillAMT 6")

summary(outliersdata$BILL_AMT6)
benchbalAMT6high <- 49198+ (1.5 * IQR(outliersdata$BILL_AMT6))

benchbalAMT6LOW <- 1256 - (1.5 * IQR(outliersdata$BILL_AMT6))

IQR(outliersdata$BILL_AMT6)

benchbalAMT6high

benchbalAMT6LOW
outliersdata$BILL_AMT6[outliersdata$BILL_AMT6 > benchbalAMT6high] <- benchbalAMT6high

outliersdata$BILL_AMT6[outliersdata$BILL_AMT6 < benchbalAMT6LOW] <- benchbalAMT6LOW


boxplot(outliersdata$BILL_AMT6, main = "Boxplot of BILL AMT6")
summary(outliersdata$BILL_AMT6)
boxplot(outliersdata$PAY_AMT1,main = "Boxplot of PAY_AMT1")

summary(outliersdata$PAY_AMT1)
benchbalPAY_AMT1high <- 5006 + (1.5 * IQR(outliersdata$PAY_AMT1))

benchbalPAY_AMT1LOW <- 1000 - (1.5 * IQR(outliersdata$PAY_AMT1))

IQR(outliersdata$PAY_AMT1)

benchbalPAY_AMT1high

benchbalPAY_AMT1LOW
outliersdata$PAY_AMT1[outliersdata$PAY_AMT1 > benchbalPAY_AMT1high] <- benchbalPAY_AMT1high

outliersdata$PAY_AMT1[outliersdata$PAY_AMT1 <benchbalPAY_AMT1LOW] <-benchbalPAY_AMT1LOW


boxplot(outliersdata$PAY_AMT1, main = "Boxplot ofPAY_AMT1")
summary(outliersdata$PAY_AMT1)
boxplot(outliersdata$PAY_AMT2,main = "Boxplot of PAY_AMT2")

summary(outliersdata$PAY_AMT2)
benchbalPAY_AMT2high <- 5000 + (1.5 * IQR(outliersdata$PAY_AMT2))

benchbalPAY_AMT2LOW <- 833 - (1.5 * IQR(outliersdata$PAY_AMT2))

IQR(outliersdata$PAY_AMT2)

benchbalPAY_AMT2high
```

```
benchbalPAY_AMT2LOW
outliersdata$PAY_AMT2[outliersdata$PAY_AMT2 > benchbalPAY_AMT2high] <- benchbalPAY_AMT2high

outliersdata$PAY_AMT2[outliersdata$PAY_AMT2 <benchbalPAY_AMT2LOW] <-benchbalPAY_AMT2LOW


boxplot(outliersdata$PAY_AMT2, main = "Boxplot ofPAY_AMT2")
summary(outliersdata$PAY_AMT2)
boxplot(outliersdata$PAY_AMT3,main = "Boxplot of PAY_AMT3")

summary(outliersdata$PAY_AMT3)
benchbalPAY_AMT3high <-4505 + (1.5 * IQR(outliersdata$PAY_AMT3))

benchbalPAY_AMT3LOW <- 390 - (1.5 * IQR(outliersdata$PAY_AMT3))

IQR(outliersdata$PAY_AMT3)

benchbalPAY_AMT3high

benchbalPAY_AMT3LOW
outliersdata$PAY_AMT3[outliersdata$PAY_AMT3 > benchbalPAY_AMT3high] <- benchbalPAY_AMT3high

outliersdata$PAY_AMT3[outliersdata$PAY_AMT3 <benchbalPAY_AMT3LOW] <-benchbalPAY_AMT3LOW


boxplot(outliersdata$PAY_AMT3, main = "Boxplot ofPAY_AMT3")
summary(outliersdata$PAY_AMT3)
boxplot(outliersdata$PAY_AMT4,main = "Boxplot of PAY_AMT4")

summary(outliersdata$PAY_AMT4)
benchbalPAY_AMT4high <- 4013+ (1.5 * IQR(outliersdata$PAY_AMT4))

benchbalPAY_AMT4LOW <- 296- (1.5 * IQR(outliersdata$PAY_AMT4))

IQR(outliersdata$PAY_AMT4)

benchbalPAY_AMT4high

benchbalPAY_AMT4LOW
outliersdata$PAY_AMT4[outliersdata$PAY_AMT4 > benchbalPAY_AMT4high] <- benchbalPAY_AMT4high

outliersdata$PAY_AMT4[outliersdata$PAY_AMT4 <benchbalPAY_AMT4LOW] <-benchbalPAY_AMT4LOW


boxplot(outliersdata$PAY_AMT4, main = "Boxplot ofPAY_AMT4")
summary(outliersdata$PAY_AMT4)
boxplot(outliersdata$PAY_AMT5,main = "Boxplot of PAY_AMT5")

summary(outliersdata$PAY_AMT5)
benchbalPAY_AMT5high <- 4031.5+ (1.5 * IQR(outliersdata$PAY_AMT5))

benchbalPAY_AMT5LOW <- 252.5 - (1.5 * IQR(outliersdata$PAY_AMT5))
```

```
IQR(outliersdata$PAY_AMT5)

benchbalPAY_AMT5high

benchbalPAY_AMT5LOW
outliersdata$PAY_AMT5[outliersdata$PAY_AMT5 > benchbalPAY_AMT5high] <- benchbalPAY_AMT5high

outliersdata$PAY_AMT5[outliersdata$PAY_AMT5 <benchbalPAY_AMT5LOW] <-benchbalPAY_AMT5LOW


boxplot(outliersdata$PAY_AMT5, main = "Boxplot ofPAY_AMT5")
summary(outliersdata$PAY_AMT5)
boxplot(outliersdata$PAY_AMT6,main = "Boxplot of PAY_AMT6")

summary(outliersdata$PAY_AMT6)
benchbalPAY_AMT6high <- 4000.0 + (1.5 * IQR(outliersdata$PAY_AMT6))

benchbalPAY_AMT6LOW <- 117.8 - (1.5 * IQR(outliersdata$PAY_AMT6))

IQR(outliersdata$PAY_AMT6)

benchbalPAY_AMT6high

benchbalPAY_AMT6LOW
outliersdata$PAY_AMT6[outliersdata$PAY_AMT6 > benchbalPAY_AMT6high] <- benchbalPAY_AMT6high

outliersdata$PAY_AMT6[outliersdata$PAY_AMT6 <benchbalPAY_AMT6LOW] <-benchbalPAY_AMT6LOW


boxplot(outliersdata$PAY_AMT6, main = "Boxplot ofPAY_AMT6")
summary(outliersdata$PAY_AMT6)
converteddata$BILL_AMT_SUM <-
  rowSums(converteddata[c("BILL_AMT1",
                                  "BILL_AMT2",
                                  "BILL_AMT3",
                                  "BILL_AMT4",
                                  "BILL_AMT5",
                                  "BILL_AMT6")])
converteddata$PAY_AMT_SUM <-
  rowSums(converteddata[c("PAY_AMT1",
                                  "PAY_AMT2",
                                  "PAY_AMT3",
                                  "PAY_AMT4",
                                  "PAY_AMT5",
                                  "PAY_AMT6")])
ggplot(converteddata) +
 aes(x = BILL_AMT_SUM, fill = DEFAULT) +
 geom_histogram(bins = 30L) +
 scale_fill_brewer(palette = "Pastel2") +
 theme_minimal()
ggplot(converteddata) +
 aes(x = PAY_AMT_SUM, fill = DEFAULT) +
 geom_histogram(bins = 30L) +
```

```
 scale_fill_brewer(palette = "Pastel2") +
 theme_minimal()
ggplot(converteddata) +
 aes(x = BILL_AMT_SUM, y = PAY_AMT_SUM, colour = DEFAULT) +
 geom_point(size = 1L) +
 scale_color_brewer(palette = "Pastel2") +
 theme_minimal()
plot_correlation((converteddata), type = "c")
prop.table(table(converteddata$DEFAULT))* 100
table(converteddata$DEFAULT)
smotebalancedset <- converteddata
smotebalancedset <- SMOTE(DEFAULT ~ ., smotebalancedset, perc.over = 400,perc.under=100)
table(smotebalancedset$DEFAULT)
prop.table(table(smotebalancedset$DEFAULT)) * 100

outliersdata <- cbind(outliersdata,converteddata[,c(25,26)])


SMOTEOUTLIERTREATED <- outliersdata
SMOTEOUTLIERTREATED <- SMOTE(DEFAULT ~ ., SMOTEOUTLIERTREATED, perc.over = 400,perc.under=100)
table(SMOTEOUTLIERTREATED$DEFAULT)
prop.table(table(SMOTEOUTLIERTREATED$DEFAULT)) * 100


standardizeddaaset <- converteddata
standardizeddaaset$PAY_1<- as.numeric(as.character(standardizeddaaset$PAY_1))
standardizeddaaset$PAY_2<- as.numeric(as.character(standardizeddaaset$PAY_2))
standardizeddaaset$PAY_3<- as.numeric(as.character(standardizeddaaset$PAY_3))
standardizeddaaset$PAY_4<- as.numeric(as.character(standardizeddaaset$PAY_4))
standardizeddaaset$PAY_5<- as.numeric(as.character(standardizeddaaset$PAY_5))
standardizeddaaset$PAY_6<- as.numeric(as.character(standardizeddaaset$PAY_6))
vecforstan<- c(1,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,25,26)
options(digits=2)

for (i in vecforstan) {

standardizeddaaset[,i] <- scale(standardizeddaaset[,i],center = TRUE, scale = TRUE)

}
summary(standardizeddaaset)
standardizeddaaset<- as.data.frame(standardizeddaaset)
converteddata$PAY_1<- as.numeric(as.character(converteddata$PAY_1))
converteddata$PAY_2<- as.numeric(as.character(converteddata$PAY_2))
converteddata$PAY_3<- as.numeric(as.character(converteddata$PAY_3))
converteddata$PAY_4<- as.numeric(as.character(converteddata$PAY_4))
converteddata$PAY_5<- as.numeric(as.character(converteddata$PAY_5))
converteddata$PAY_6<- as.numeric(as.character(converteddata$PAY_6))
FeatEngineered <- converteddata[,-c(12:23)]
outliersdata$PAY_1<- as.numeric(as.character(outliersdata$PAY_1))
outliersdata$PAY_2<- as.numeric(as.character(outliersdata$PAY_2))
outliersdata$PAY_3<- as.numeric(as.character(outliersdata$PAY_3))
outliersdata$PAY_4<- as.numeric(as.character(outliersdata$PAY_4))
outliersdata$PAY_5<- as.numeric(as.character(outliersdata$PAY_5))
```

```
outliersdata$PAY_6<- as.numeric(as.character(outliersdata$PAY_6))
OTFeatEnginerd <- outliersdata[,-c(12:23)]
smotebalancedset$PAY_1<- as.numeric(as.character(smotebalancedset$PAY_1))
smotebalancedset$PAY_2<- as.numeric(as.character(smotebalancedset$PAY_2))
smotebalancedset$PAY_3<- as.numeric(as.character(smotebalancedset$PAY_3))
smotebalancedset$PAY_4<- as.numeric(as.character(smotebalancedset$PAY_4))
smotebalancedset$PAY_5<- as.numeric(as.character(smotebalancedset$PAY_5))
smotebalancedset$PAY_6<- as.numeric(as.character(smotebalancedset$PAY_6))
SMOTEdataset <- smotebalancedset[,-c(12:23)]
SMOTEOUTLIERTREATED$PAY_1<- as.numeric(as.character(SMOTEOUTLIERTREATED$PAY_1))
SMOTEOUTLIERTREATED$PAY_2<- as.numeric(as.character(SMOTEOUTLIERTREATED$PAY_2))
SMOTEOUTLIERTREATED$PAY_3<- as.numeric(as.character(SMOTEOUTLIERTREATED$PAY_3))
SMOTEOUTLIERTREATED$PAY_4<- as.numeric(as.character(SMOTEOUTLIERTREATED$PAY_4))
SMOTEOUTLIERTREATED$PAY_5<- as.numeric(as.character(SMOTEOUTLIERTREATED$PAY_5))
SMOTEOUTLIERTREATED$PAY_6<- as.numeric(as.character(SMOTEOUTLIERTREATED$PAY_6))
OTSMOTE <- SMOTEOUTLIERTREATED[,-c(12:23)]
standardDS<- standardizeddaaset[,-c(12:23)]
seed <- 101
set.seed(seed)

sampleFeatEngineered <- sample.split(FeatEngineered$DEFAULT,SplitRatio = 0.7)
FeatEngineered.train <- subset(FeatEngineered,sampleFeatEngineered == TRUE)
FeatEngineered.test <- subset(FeatEngineered,sampleFeatEngineered == FALSE)

set.seed(seed)
sampleOTFeatEnginerd <- sample.split(OTFeatEnginerd$DEFAULT,SplitRatio = 0.7)
OTFeatEnginerd.train <- subset(OTFeatEnginerd,sampleOTFeatEnginerd == TRUE)
OTFeatEnginerd.test <- subset(OTFeatEnginerd,sampleOTFeatEnginerd == FALSE)

set.seed(seed)

sampleSMOTE <- sample.split(SMOTEdataset$DEFAULT,SplitRatio = 0.7)
SMOTE.train <- subset(SMOTEdataset,sampleSMOTE == TRUE)
SMOTE.test <- subset(SMOTEdataset,sampleSMOTE == FALSE)


set.seed(seed)

sampleOTSMOTE <- sample.split(OTSMOTE$DEFAULT,SplitRatio = 0.7)
SMOTE.OTSMOTE.train <- subset(OTSMOTE,sampleOTSMOTE == TRUE)
SMOTE.OTSMOTE.test <- subset(OTSMOTE,sampleOTSMOTE == FALSE)
set.seed(seed)

samplestandardDS <- sample.split(standardDS$DEFAULT,SplitRatio = 0.7)
standardDS.train <- subset(standardDS,samplestandardDS == TRUE)
standardDS.test <- subset(standardDS,samplestandardDS == FALSE)
fullmodFeatEngineered <- glm(DEFAULT ~. ,family=binomial,data=FeatEngineered.train)

emptyModelFeatEngineered<- glm(DEFAULT ~ 1,family=binomial,data = FeatEngineered.train)
backwardsFeatEngineered = step(fullmodFeatEngineered)
forwardsFeatEngineered = step(emptyModelFeatEngineered,scope=list(lower=formula(emptyModelFeatEngineere
FitLogModelFeatEng <- glm(DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
    PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM + PAY_6 , data=FeatEngineered.train,family="binomial")
```

```
summary(FitLogModelFeatEng)
logpredFeatEng<- predict(FitLogModelFeatEng,FeatEngineered.test[-12],type = "response")
ypredlogfeatEng <- as.factor(ifelse(logpredFeatEng > 0.5,"Yes","No"))

ypredlogfeatEng
confusionMatrix(FeatEngineered.test$DEFAULT,ypredlogfeatEng, positive = "Yes", mode = "everything")
KNNControlCaretFeatEng <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
knnCaretFeatEng <- train(DEFAULT~., data = FeatEngineered.train, method = "knn",
                 trControl=KNNControlCaretFeatEng,
                 tuneLength = 5)


knnCaretFeatEng
plot(knnCaretFeatEng)
testCaretKNNFeatEngg <- predict(knnCaretFeatEng, newdata = FeatEngineered.test)
confusionMatrix( FeatEngineered.test$DEFAULT,testCaretKNNFeatEngg,positive = "Yes", mode = "everything"
ROCtestFeatEnginerd<- roc.curve(FeatEngineered.test$DEFAULT,testCaretKNNFeatEngg, main="ROC curve KNN")


nb_FeatEngreed<-naiveBayes(x=FeatEngineered.train[,-12], y=FeatEngineered.train[,12])



pred_nbFeatEngineered<-predict(nb_FeatEngreed,newdata = FeatEngineered.test[,-12])


confusionMatrix( FeatEngineered.test$DEFAULT,pred_nbFeatEngineered,positive = "Yes", mode = "everything
ROCtestfeatenginerrred<- roc.curve(FeatEngineered.test$DEFAULT,pred_nbFeatEngineered, main="ROC curve N

fitControlFeatengnred <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)

set.seed(seed)
model_marsfeatenginerred = train(DEFAULT ~ ., data=FeatEngineered.train, method='rpart', tuneLength = 15

model_marsfeatenginerred
fancyRpartPlot(model_marsfeatenginerred$finalModel)

predictedfeatenginrd <- predict(model_marsfeatenginerred, FeatEngineered.test)

confusionMatrix(reference = FeatEngineered.test$DEFAULT, data = predictedfeatenginrd , mode='everything
ROCtestfeatengrdcart<- roc.curve(FeatEngineered.test$DEFAULT,predictedfeatenginrd, main="ROC curve CART
fitControlStandadizedRF <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
```

```
)
set.seed(seed)


model_rfFeatEnginerred= train(DEFAULT ~ ., data=FeatEngineered.train, method='rf', tuneLength=5, metric=

model_rfFeatEnginerred
predictedRFFeatEnginerred <- predict(model_rfFeatEnginerred, FeatEngineered.test)
confusionMatrix(reference = FeatEngineered.test$DEFAULT, data = predictedRFFeatEnginerred, mode='everyth
ROCtestFeatEngrded<- roc.curve(FeatEngineered.test$DEFAULT,predictedRFFeatEnginerred, main="ROC curve R
BaggingFeatEngineered <- bagging(DEFAULT ~.,
data=FeatEngineered.train,
control=rpart.control(maxdepth=30, minsplit=1))


BaggingFeatEnginrrrrdd <- predict(BaggingFeatEngineered, FeatEngineered.test)
BaggingFeatEnginrrrrdd$class<- as.factor(BaggingFeatEnginrrrrdd$class)
class(BaggingFeatEnginrrrrdd$class)

confusionMatrix(FeatEngineered.test$DEFAULT, BaggingFeatEnginrrrrdd$class,positive = "Yes", mode = "eve
ROCtestfeatEnginederedered<- roc.curve(FeatEngineered.test$DEFAULT,BaggingFeatEnginrrrrdd$class, main="
adaboostmodelFeatEngineered <- boosting(DEFAULT~., data=FeatEngineered.train, boos=TRUE, mfinal=50)
print(names(adaboostmodelFeatEngineered))
print(adaboostmodelFeatEngineered$trees[1])
predAdaBoostsFeatEngineered = predict(adaboostmodelFeatEngineered, FeatEngineered.test)
print(predAdaBoostsFeatEngineered$confusion)
print(predAdaBoostsFeatEngineered$error)
predAdaBoostsFeatEngineered$class<- as.factor(predAdaBoostsFeatEngineered$class)
confusionMatrix(FeatEngineered.test$DEFAULT,predAdaBoostsFeatEngineered$class,mode="everything",positive
ROCtestFeatEnginered<- roc.curve(FeatEngineered.test$DEFAULT,predAdaBoostsFeatEngineered$class, main="R
fullmodSMOTE <- glm(DEFAULT ~. ,family=binomial,data=SMOTE.train)

emptyModelSMOTE<- glm(DEFAULT ~ 1,family=binomial,data = SMOTE.train)
backwardsSMOTE = step(fullmodSMOTE)
forwardsSMOTE = step(emptyModelSMOTE,scope=list(lower=formula(emptyModelSMOTE),upper=formula(fullmodSMO
FitLogMSMOTE <- glm(DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_3 + SEX + EDUCATION +
    BILL_AMT_SUM + PAY_5 + MARRIAGE + PAY_4 + PAY_6 + AGE , data= SMOTE.train,family="binomial")

summary(FitLogMSMOTE)
vif(FitLogMSMOTE)
logpredSMOTE<- predict(FitLogMSMOTE,SMOTE.test,type = "response")
ypredlogSMOTE <- as.factor(ifelse(logpredSMOTE > 0.5,"Yes","No"))

ypredlogSMOTE
confusionMatrix( SMOTE.test$DEFAULT,ypredlogSMOTE,positive = "Yes", mode = "everything")
KNNControlCaretSMOTE <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
knnCaretSMOTE <- train(DEFAULT~., data = SMOTE.train, method = "knn",
                 trControl=KNNControlCaretSMOTE,
                 tuneLength = 5)


knnCaretSMOTE
plot(knnCaretSMOTE)
```

```
testCaretKNNSMOTE <- predict(knnCaretSMOTE, newdata = SMOTE.test)
confusionMatrix( SMOTE.test$DEFAULT,testCaretKNNSMOTE,positive = "Yes", mode = "everything")
library(e1071)

nb_SMOTE<-naiveBayes(x=SMOTE.train[,-12], y=SMOTE.train[,12])



pred_nbSMOTE<-predict(nb_SMOTE,newdata = SMOTE.test[,-12])


confusionMatrix(pred_nbSMOTE, SMOTE.test$DEFAULT,positive = "Yes", mode = "everything")

fitControlSMOTECART <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)
library(earth)

set.seed(seed)
model_marsSMOTE = train(DEFAULT ~ ., data=SMOTE.train, method='rpart',parms = list(split = "information

model_marsSMOTE
fancyRpartPlot(model_marsSMOTE$finalModel)

predictedCARTSMOTE <- predict(model_marsSMOTE, SMOTE.test)

confusionMatrix(reference = SMOTE.test$DEFAULT, data = predictedCARTSMOTE, mode='everything', positive=
fitControlSMOTERF <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)
set.seed(seed)


model_rfSMOTE = train(DEFAULT ~ ., data=SMOTE.train, method='rf', tuneLength=5, metric='ROC', trControl

model_rfSMOTE
predictedRFSMOTE <- predict(model_rfSMOTE, SMOTE.test)
plot(model_rfSMOTE, top = 20)
RFSMOTEimp <- varImp(model_rfSMOTE, scale = FALSE)

RFSMOTEimp

plot(RFSMOTEimp, top = 20)
confusionMatrix(reference = SMOTE.test$DEFAULT, data = predictedRFSMOTE, mode='everything', positive='Y
fullmodStandardized <- glm(DEFAULT ~. ,family=binomial,data=standardDS.train)
```

```
emptyModeltandardized<- glm(DEFAULT ~ 1,family=binomial,data = standardDS.train)
backwardstandardized = step(fullmodStandardized)

forwardsstandardized  = step(emptyModeltandardized,scope=list(lower=formula(emptyModeltandardized),uppe


FitLogModelstandardized <- glm(DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + EDUCATION + AGE + LIMIT_BAL +
    PAY_3 + MARRIAGE + SEX + BILL_AMT_SUM + PAY_6, data=standardDS.train,family= binomial(link='logit'))


vif(FitLogModelstandardized)
summary(FitLogModelstandardized)
logpredstandarized<- predict(FitLogModelstandardized,standardDS.test[-12],type = "response")
ypredlogstandardized <- as.factor(ifelse(logpredstandarized > 0.5,"Yes","No"))

ypredlogstandardized
confusionMatrix( standardDS.test$DEFAULT,ypredlogstandardized,positive = "Yes", mode = "everything")
KNNControlstandardized <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
knnCaretstandardized <- train(DEFAULT~., data = standardDS.train, method = "knn",
                 trControl=KNNControlstandardized,
                 tuneLength = 5)


knnCaretstandardized
plot(knnCaretstandardized)
testCaretstandardized <- predict(knnCaretstandardized, newdata = standardDS.test)
confusionMatrix( standardDS.test$DEFAULT,testCaretstandardized, positive = "Yes", mode = "everything")
library(e1071)

nb_standarized<-naiveBayes(x=standardDS.train[,-12], y=standardDS.train[,12])



pred_nbstandardized<-predict(nb_standarized,newdata = standardDS.test[,-12])


confusionMatrix( standardDS.test$DEFAULT,pred_nbstandardized,positive = "Yes", mode = "everything")

fitControlstandardized <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)
library(earth)

set.seed(seed)
model_marsstandardized = train(DEFAULT ~ ., data=standardDS.train, method='rpart',parms = list(split =

model_marsstandardized

predictedstandardized <- predict(model_marsstandardized, standardDS.test)
```

```
confusionMatrix(reference = standardDS.test$DEFAULT, data = predictedstandardized , mode='everything', p
ROCtestCART<- roc.curve(standardDS.test$DEFAULT,predictedstandardized, main="ROC curve CART")
fitControlStandadizedRF <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)
set.seed(seed)


model_rfstandardized = train(DEFAULT ~ ., data=standardDS.train, method='rf', tuneLength=5, metric='ROC

model_rfstandardized
predictedRFstandardize <- predict(model_rfstandardized, standardDS.test)
confusionMatrix(reference = standardDS.test$DEFAULT, data = predictedRFstandardize, mode='everything', p
ROCtestRF<- roc.curve(standardDS.test$DEFAULT,predictedRFstandardize, main="ROC curve RF")
Car.Baggingstandardize <- bagging(DEFAULT ~.,
data=standardDS.train,
control=rpart.control(maxdepth=30, minsplit=1))


BaggingCarstandardize <- predict(Car.Baggingstandardize, standardDS.test)
BaggingCarstandardize$class<- as.factor(BaggingCarstandardize$class)
class(BaggingCarstandardize$class)

confusionMatrix(standardDS.test$DEFAULT, BaggingCarstandardize$class,positive = "Yes", mode = "everythi
ROCtestBAGGING<- roc.curve(standardDS.test$DEFAULT,BaggingCarstandardize$class, main="ROC curve Bagging
adaboostmodelstandardize <- boosting(DEFAULT~., data=standardDS.train, boos=TRUE, mfinal=50)
print(names(adaboostmodelstandardize))
print(adaboostmodelstandardize$trees[1])
predAdaBooststandaridize = predict(adaboostmodelstandardize, standardDS.test)
print(predAdaBooststandaridize$confusion)
print(predAdaBooststandaridize$error)
predAdaBooststandaridize$class<- as.factor(predAdaBooststandaridize$class)
confusionMatrix(standardDS.test$DEFAULT,predAdaBooststandaridize$class,mode="everything",positive = "Yes
ROCAdaBoost<- roc.curve(standardDS.test$DEFAULT,predAdaBooststandaridize$class, main="ROC curve Boostin
SMOTESTANDARTDATASET <- SMOTEdataset
stanvector <-c(1,5,6,7,8,9,10,11,13,14)
options(digits=2)

for (i in stanvector) {

SMOTESTANDARTDATASET[,i] <- scale(SMOTESTANDARTDATASET[,i],center = TRUE, scale = TRUE)

}

set.seed(seed)

sampleSMOTESTANDARTDATASET <- sample.split(SMOTESTANDARTDATASET$DEFAULT,SplitRatio = 0.7)
SMOTESTANDARTDATASET.train <- subset(SMOTESTANDARTDATASET,sampleSMOTESTANDARTDATASET == TRUE)
SMOTESTANDARTDATASET.test <- subset(SMOTESTANDARTDATASET,sampleSMOTESTANDARTDATASET == FALSE)
```

```r
fullmodSMOTESTANDART <- glm(DEFAULT ~. ,family=binomial,data=SMOTESTANDARTDATASET.train)

emptyModelSMOTESTANDART<- glm(DEFAULT ~ 1,family=binomial,data = SMOTESTANDARTDATASET.train)
backwardSMOTESTANDART = step(fullmodSMOTESTANDART)

forwardSMOTESTANDART  = step(emptyModelSMOTESTANDART,scope=list(lower=formula(emptyModelSMOTESTANDART),


FitLogModelSMOTESTANDARD <- glm(DEFAULT ~ PAY_1 + PAY_AMT_SUM + PAY_2 + PAY_3 + SEX + EDUCATION +
    BILL_AMT_SUM + PAY_5 + MARRIAGE + PAY_4 + PAY_6 + AGE, data=SMOTESTANDARTDATASET.train,family= binor


vif(FitLogModelSMOTESTANDARD)
summary(FitLogModelSMOTESTANDARD)
logpredSMOTESTANDARD<- predict(FitLogModelSMOTESTANDARD,SMOTESTANDARTDATASET.test[-12],type = "response
ypredlogSMOTESTANDARD <- as.factor(ifelse(logpredSMOTESTANDARD > 0.5,"Yes","No"))

confusionMatrix( SMOTESTANDARTDATASET.test$DEFAULT,ypredlogSMOTESTANDARD,positive = "Yes", mode = "every
ROCtestSMOTEStand<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,ypredlogSMOTESTANDARD, main="ROC curve
KNNControlSMOTESTANDARD <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
knnCaretSMOTESTANDARD <- train(DEFAULT~., data = SMOTESTANDARTDATASET.train, method = "knn",
                 trControl=KNNControlSMOTESTANDARD,
                 tuneLength = 5)


knnCaretSMOTESTANDARD
plot(knnCaretSMOTESTANDARD)
testCaretSMOTESTANDARD <- predict(knnCaretSMOTESTANDARD, newdata = SMOTESTANDARTDATASET.test)
confusionMatrix( SMOTESTANDARTDATASET.test$DEFAULT,testCaretSMOTESTANDARD, positive = "Yes", mode = "ev
ROCtestSMOTESTANDARD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,testCaretSMOTESTANDARD, main="ROC cu


nb_standarizedSMOTESTANDARD<-naiveBayes(x=SMOTESTANDARTDATASET.train[,-12], y=SMOTESTANDARTDATASET.trai



pred_nbSMOTESTANDARD<-predict(nb_standarizedSMOTESTANDARD,newdata = SMOTESTANDARTDATASET.test[,-12])


confusionMatrix( SMOTESTANDARTDATASET.test$DEFAULT,pred_nbSMOTESTANDARD,positive = "Yes", mode = "every
ROCtestSMOTESTANDARDD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,pred_nbSMOTESTANDARD, main="ROC cur

fitControlSMOTESTANDARD <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)

set.seed(seed)
model_marsSMOTESTANDARD = train(DEFAULT ~ ., data=SMOTESTANDARTDATASET.train, method='rpart',parms = lis
```

```
model_marsSMOTESTANDARD
fancyRpartPlot(model_marsSMOTESTANDARD$finalModel)

predictedSMOTESTANDARD <- predict(model_marsSMOTESTANDARD, SMOTESTANDARTDATASET.test)

confusionMatrix(reference = SMOTESTANDARTDATASET.test$DEFAULT, data = predictedSMOTESTANDARD , mode='ev
ROCtestSMOTESTANDARDDCART<- roc.curve( SMOTESTANDARTDATASET.test$DEFAULT,predictedSMOTESTANDARD, main="
fitControlSMOTESTANDARDRF <- trainControl(
    method = 'cv',
    number = 5,
    savePredictions = 'final',
    classProbs = T,
    summaryFunction=twoClassSummary
)
set.seed(seed)


model_rfSMOTESTANDARD = train(DEFAULT ~ ., data=SMOTESTANDARTDATASET.train, method='rf', tuneLength=5, n

model_rfSMOTESTANDARD
predictedRFSMOTESTANDARD <- predict(model_rfSMOTESTANDARD, SMOTESTANDARTDATASET.test)
confusionMatrix(reference = SMOTESTANDARTDATASET.test$DEFAULT, data = predictedRFSMOTESTANDARD, mode='ev
ROCtestSMOTESTANDARDDDDDD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,predictedRFSMOTESTANDARD, main=
Car.BaggingSMOTESTANDARDe <- bagging(DEFAULT ~.,
data=SMOTESTANDARTDATASET.train,
control=rpart.control(maxdepth=30, minsplit=1))


BaggingSmoteStandardize <- predict(Car.BaggingSMOTESTANDARDe, SMOTESTANDARTDATASET.test)
BaggingSmoteStandardize$class<- as.factor(BaggingSmoteStandardize$class)
class(BaggingSmoteStandardize$class)

confusionMatrix(SMOTESTANDARTDATASET.test$DEFAULT, BaggingSmoteStandardize$class,positive = "Yes", mode
ROCtestSMOTESTANDARddded<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,BaggingSmoteStandardize$class, ma
adaboostmodelSmoteStandard <- boosting(DEFAULT~., data=SMOTESTANDARTDATASET.train, boos=TRUE, mfinal=50)
print(names(adaboostmodelSmoteStandard))
print(adaboostmodelSmoteStandard$trees[1])
predAdaBoostSMOTEStandard = predict(adaboostmodelSmoteStandard, SMOTESTANDARTDATASET.test)
print(predAdaBoostSMOTEStandard$confusion)
print(predAdaBoostSMOTEStandard$error)
predAdaBoostSMOTEStandard$class<- as.factor(predAdaBoostSMOTEStandard$class)
confusionMatrix(SMOTESTANDARTDATASET.test$DEFAULT,predAdaBoostSMOTEStandard$class,mode="everything",posi
ROCtestlogSMOTESTANDRDDD<- roc.curve(SMOTESTANDARTDATASET.test$DEFAULT,predAdaBoostSMOTEStandard$class,
```