Capstone **Taiwan** Defaults

short line

Bharath K N  
18th January, 2020

# Acknowledgement

I would like to thank Mr. Manju & Ms. Richa( Great lakes institute of management), Mr. Avik Paul & Dr. Hong( Kaggle open source community), Mr. Kunal Jain( Analytical Vidya ) Mr. Vishal and Vish (Stackoverflow) for guiding me through this journey which wouldn’t have been possible otherwise.

# Content

1. Problem introduction ------------------------------------------------------------------------------------------------ 3
2. Univariate and Bivariate analysis -------------------------------------------------------------------------------- 4
3. Initial findings ----------------------------------------------------------------------------------------------------------- 8
4. Collinearity -------------------------------------------------------------------------------------------------------------- 11
5. Feature engineering ------------------------------------------------------------------------------------------------- 11
6. Data processing ------------------------------------------------------------------------------------------------------ 12
7. Outlier detection
8. Class imbalance
9. VIF function
10. Model selection ------------------------------------------------------------------------------------------------------- 13
11. Logistic
12. Random forest
13. Decision tree
14. Ensemble modelling
15. Building logistic models ------------------------------------------------------------------------------------------- 14

* Performance parameters -------------------------------------------------------------------------------15

1. Building Decision tree models ---------------------------------------------------------------------------------- 17

* Performance parameters -------------------------------------------------------------------------------18

1. Building Random forest -------------------------------------------------------------------------------------------- 19

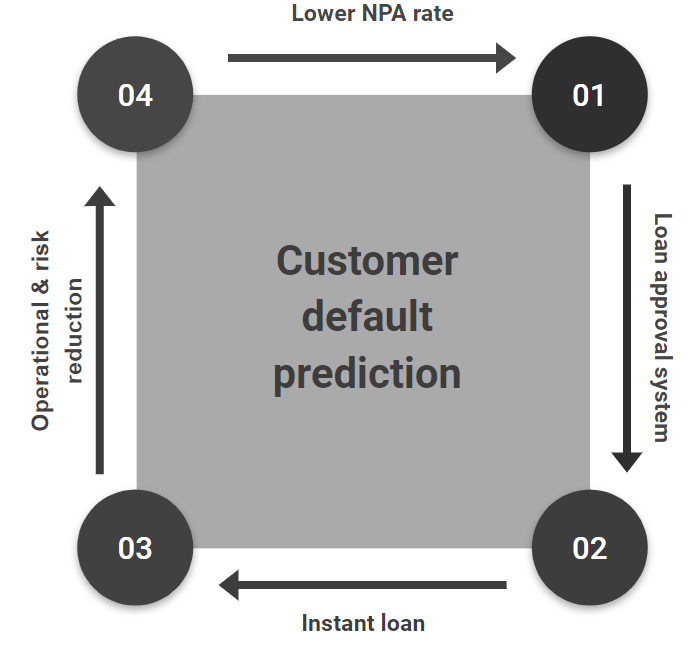
* Performance parameters ------------------------------------------------------------------------------ 20

1. Conclusion ------------------------------------------------------------------------------------------------------------- 21
2. Revolution and recommendation ------------------------------------------------------------------------------ 21
3. Business advantage ------------------------------------------------------------------------------------- 22
4. Revenue prediction -------------------------------------------------------------------------------------- 22
5. Appendix --------------------------------------------------------------------------------------------------------------- 24
6. R code ------------------------------------------------------------------------------------------------------------------- 32

# Problem introduction

The business demands to build a class leading loan approval system which can make data driven decision for NPA reduction and be cost efficient by reducing the operation cost of the company and providing instant loans to the customer.

We aim to be the market leaders by introducing a system which can take autonomous decisions on the customer credit limit and alert recovery teams in case of possible occurrence of default.



The given dataset has payment history, age, sex, education background, limit on balance, billing amount, total amount and many other parameters which will be used in building a scalable model to achieve this business goal. We are to suggest & recommend alternate features which can increase the model performance and predictive power.

# Univariate and Bivariate analysis

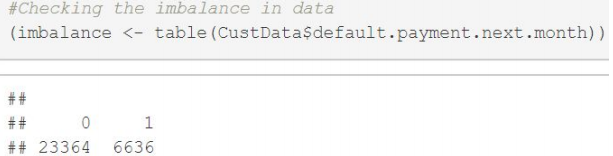
We start our analysis by renaming the columns to our convenience:

1. Repayment columns Pay\_0 to Pay\_6 as *Sept, Aug...April*
2. Bill\_Amt1 to Bill\_Amt6 to *Bill\_Sept, Bill\_Aug...Bill\_April*
3. Pay\_Amt1 to Pay\_Amt6 to *Pay\_Sept, Pay\_Aug...Pay\_April*

Post the name transformation we work our way to convert certain variables from factor to numeric and vice-versa. Some of the variables on which we have carried out this processes are

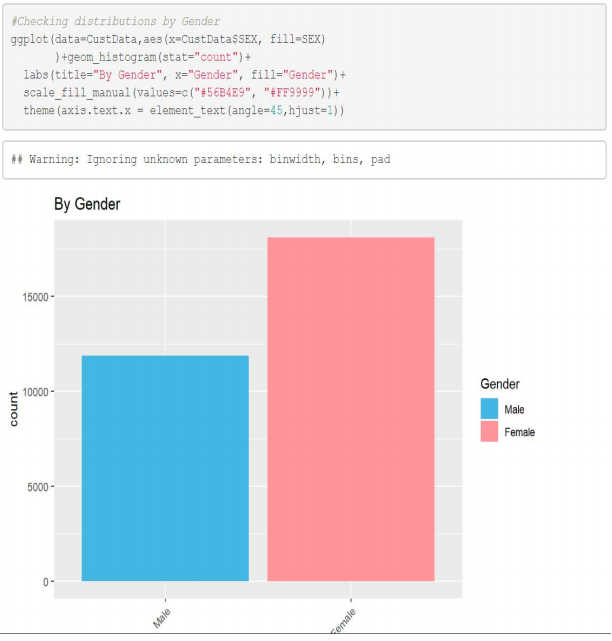
1. Gender
2. Marriage
3. Education
4. Pay\_0 to Pay\_6
5. Default.payment.next.month

We can now check the balance of the dependent variable in the dataset. This will help us in confirming if any oversampling methods are to be used

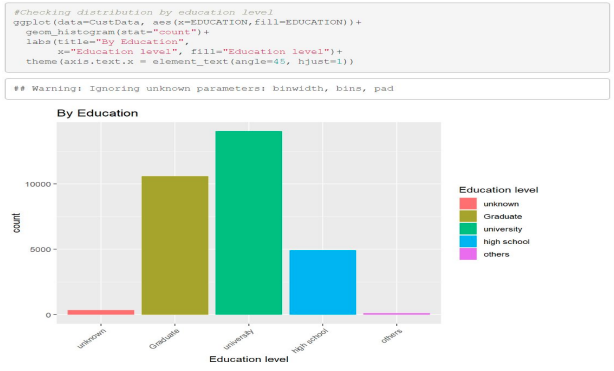


By looking at the data here, we see that there is no class imbalance and thus we carry on with supervised machine learning algorithm and check on the ensemble methods if necessary.

Let us begin with the univariate analysis. The first relationship we observe is in the variable SEX, plotting a basic graph tells us that the data is female dominated.



Checking the variable education reveals that the number of customers who are university passouts are higher than compared to Graduates, high school and later followed by others and unknowns.



The data also showcases that in the dataset the number of singles are higher than compared to married.

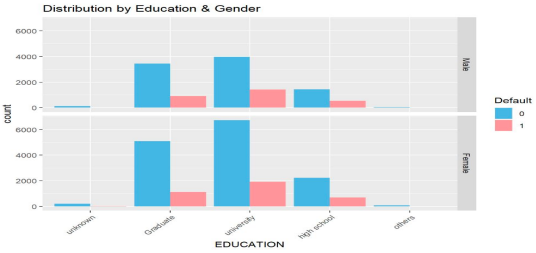
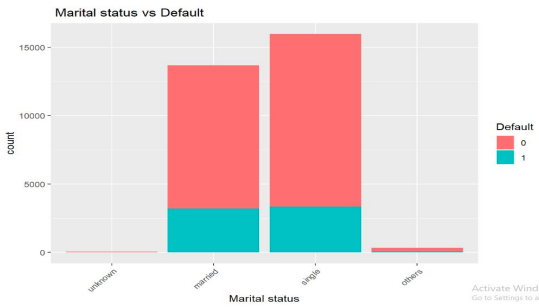
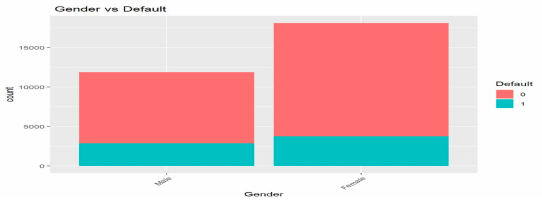
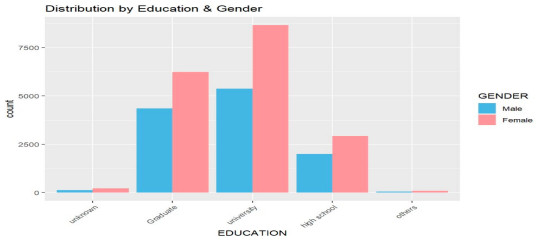


Now that our preliminary checks are done we conclude that

1. There is no class imbalance in the data
2. The dataset is female dominated
3. There are more university graduates than compared to other education type
4. The dataset also reveals that the number of singles are higher than compared to the married
5. There is no missing data

Bivariate analysis can bring in more table and basic associations as we compare it with other variables. Let’s begin!

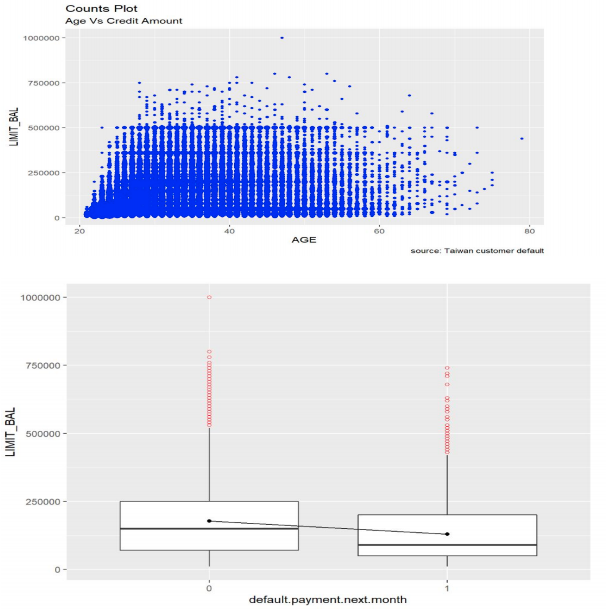
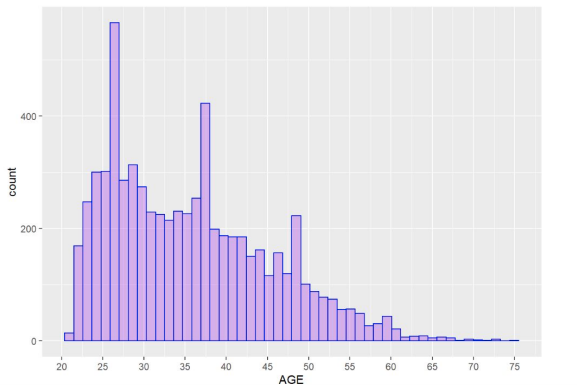
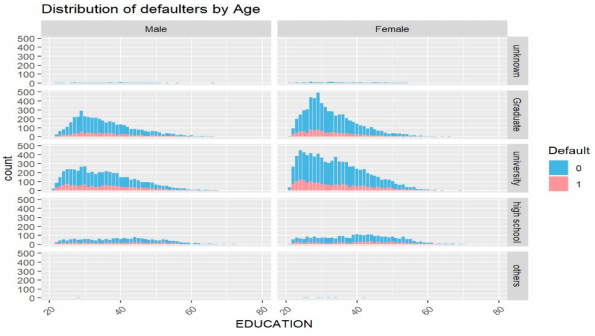
Let’s start by looking at these insights from the data.



We observe here that

1. Across all Education qualification there are more females than compared to the males
2. Though there are higher females in the dataset the number of defaulting males to defaulting females is close to 1
3. We also observe that marital status plays no significance when it comes to defaults
4. When you further compare the default ratios in education by Gender we see that the default rates are similar there as well.

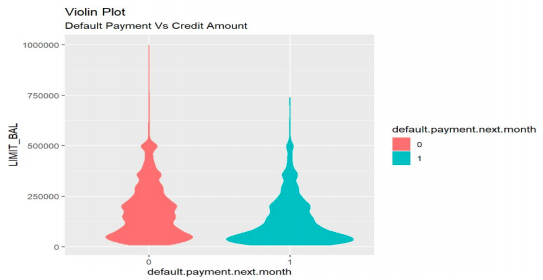
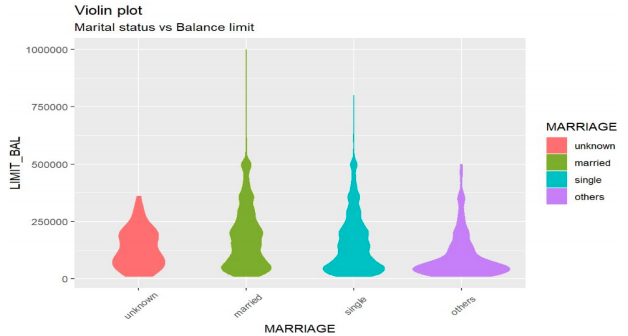
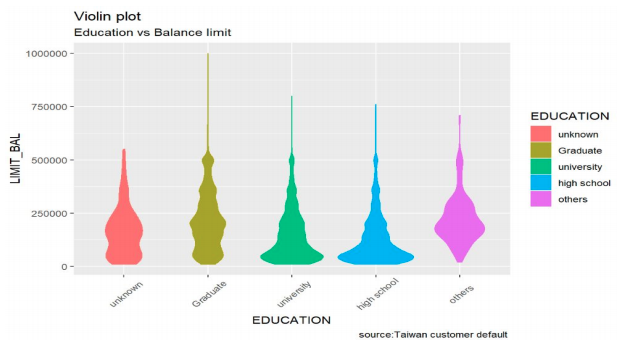
Let’s get in further by doing a small analysis on the category of age group who have been defaulting, this will not only help us in identifying the models about the certain group but also in classifying and informing the operations team to take caution.



This visualization tells us that

1. A lot of defaulters fall under the age group of 21-35 and as the age increases we see the trend line declines in terms of default
2. We also observe that for this particular age-group the credit limit provided is higher than compared to the other groups.
3. The amount $50,000 is the credit balance which serves as a difference between the defaulter and a non defaulter at this age group

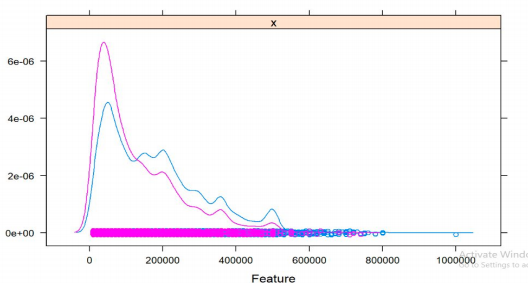
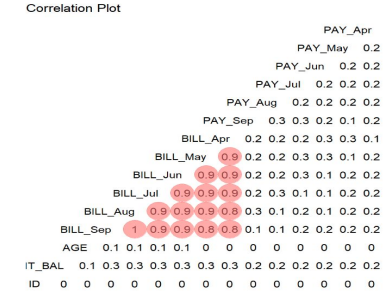
Now for the love of clarity and density of distribution, let’s check by plotting different violin graphs and listen to the story of visualization.



This indicated that

1. Graduates have a higher credit limit wherein the majority of the population is spread between $250,000 to $500,000
2. People holding education other than graduate’s limit is way lesser and its concentrated upto the $250,000 mark and not further
3. It is unclear on the default margin and the marital status
4. Defaulters are concentrated at the lower end of the spectrum, i.e between $20,000 -$100,000

Last but not least, checking our dataset for possible outliers. We need to be remove outliers as we are dealing with sensitive financial data and consequently we need to check for correlation among the variables as well.



We observe that

1. In the boxplot mentioned above, we have a few outliers which will be removed
2. We observe alarming correlation between Bill\_may to Bill\_sep data
3. But we also see a clear demarcation line in the feature graph which means that there is no overlap

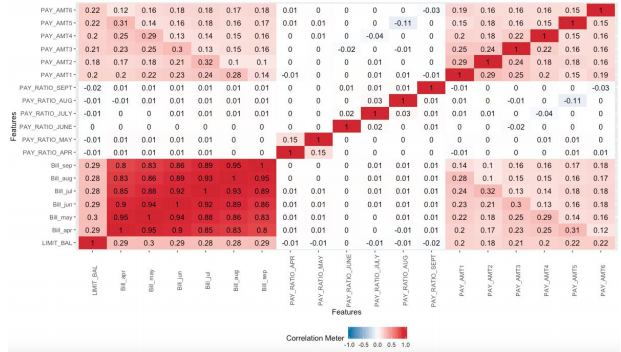
# Collinearity check

The graph clearly shows that there is clear collinearity among 6 variables marked in red squares in the figure given below and it would be advisable to drop these variables by building new features.



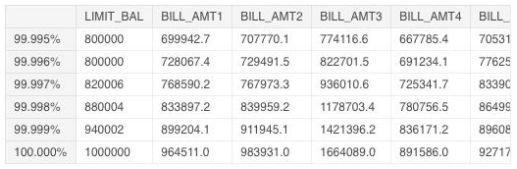
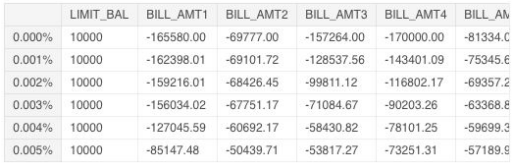
# Feature engineering

The feature that we are looking at is the ratio of bill paid to bill generated which will replace the above mentioned features. Let’s now calculate the collinearity graph again to see if these factors have reduced, but they do look skewed and thus outlier treatment is required.



# Data processing

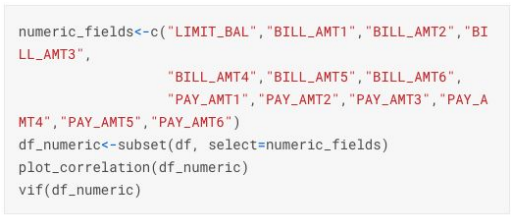
As observed in the collinear plots as well as the boxplots, there are a lot of skewness observed and thus we have decided to remove them at once as this is a financial data and any substitution can create unstable results in the model.



Rechecking on the class imbalance, we see that there is no class imbalance observed



Using VIF we can get rid of multicollinearity, we now have to handle all VIF values greater than 4. To this we shall add our feature engineered variables and build models.



# Model selection

We have four great methods to predict and draw conclusions from and we need to decide the best fit model, considering Accuracy, ROC, Precision and Recall. The models under consideration are

1. Logistic
2. Decision tree
3. Random forest
4. Ensemble modelling

# Logistic model

Logistic model has a better probability of building a model with better accuracy and the control parameters can be adjusted to get higher ROC and precision values. We have tuned the parameters to achieve better performance figures.

Random forest

In random forest, the overfitting can be a real issue however it can be overcome by pruning the tree such that better threshold values can be obtained. We will prune the tree based on the best complexity #parameter that has the least error

Ensemble modelling

Though Ensemble model gives us good results we will not be implementing it because of the following reasons(Source: Analytical Vidya)

1. Ensemble model becomes difficult to interpret and draw business conclusions from
2. The algorithm in general is heavy on the memory and processing time
3. Selecting models in Ensemble modelling is a difficult task to master

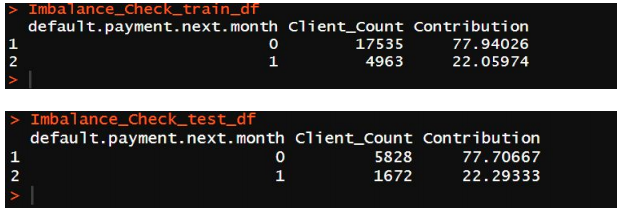
# Logistic model

# Train and test data

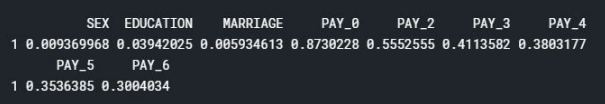
Let’s begin with creating a training and testing dataset from the data we have. For convenience we will use 75-25 split for our train and test respectively.



Let’s recheck on the class imbalance in the train and test data. We observe that there is no class imbalance and thus we go ahead



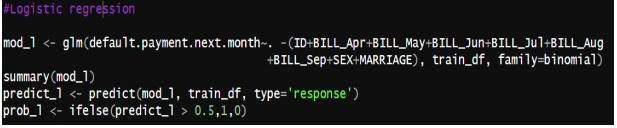
# Information value

Let us check the Information Value of the categorical variables to understand if there are any that could be omitted.

This output here shows that the variables such as SEX and MARRIAGE are very weak predictors and thus we can exclude them from building models.

# Building models

Using the glm function and excluding the variables such as SEX and MARRIAGE we build the logistic model for the given dataset.

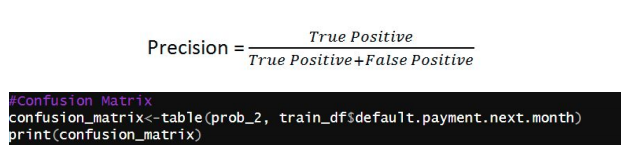


# Step AIC criteria

We use this method to calculate the best subset to build the model. This is useful in getting a better class of predictions from our models.



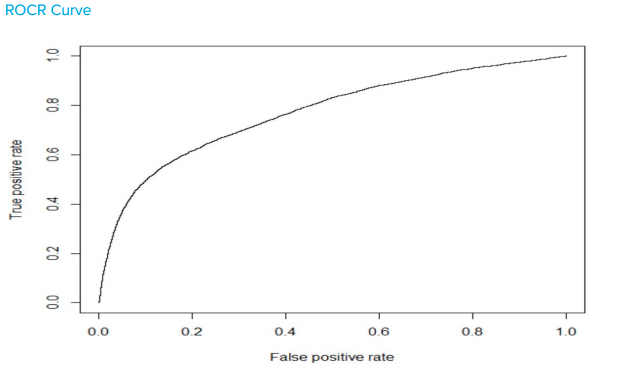
Using this method let’s rerun the model and check for the ROCR, accuracy, and Precision of the model. We know that



We get the confusion matrix and the Accuracy as **82.180**, however considering the Precision also into effect we get the score as **88.34**

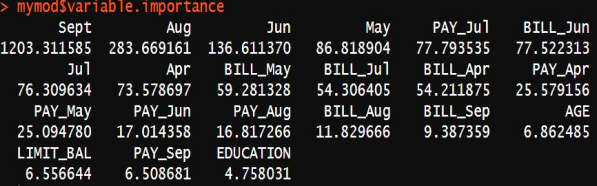
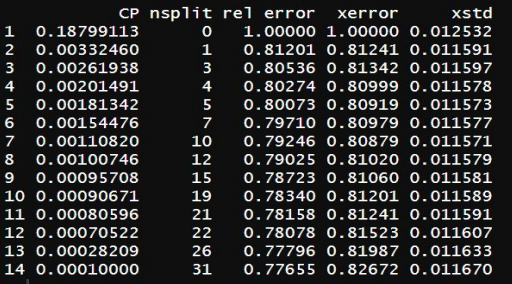


The corresponding ROCR curve is as given below

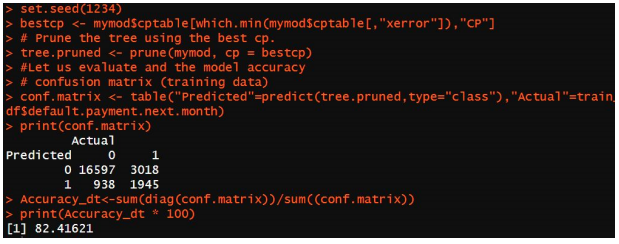


# Decision tree

Though the logistic regression model yields good results, let’s build the Tree and check for variable importance and drop variable which are not important.

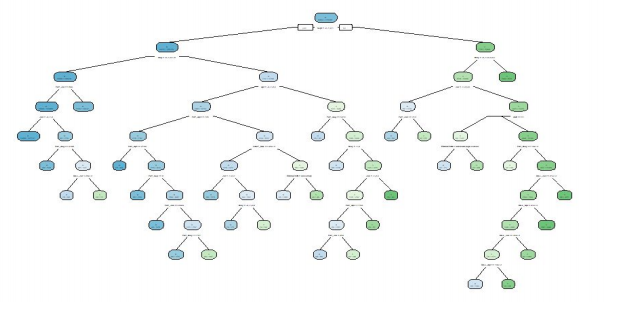


Let's re-run the model after dropping the variables "EDUCATION","SEX","MARRIAGE"



From the above result, we get the Accuracy of the model as **82.41**, which is slightly better than the logistic model and Precision of **82.41.**

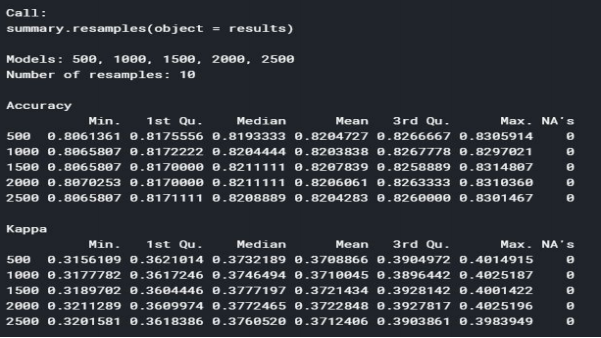
The accuracy achieved using the data set aside for validation is also similar hence our model fits the data well. Overall, we see some improvement in the accuracy of the model using the Decision Tree approach. Let's try random forest to understand if any further improvement can be brought to the model.



# Random forest

Let’s build a Random Forest anc check if the model we built can perform better than the two built above i.e Logistic and Decision tree respectively.

Results and respective Kappa’s are as follows

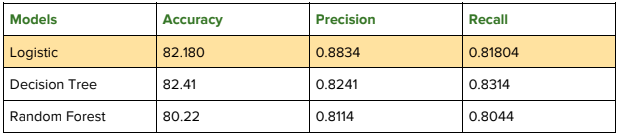


Checking on the accuracy of the following we get about **80.22**



# Conclusion

Let’s compare the respective models along with their performance measures to find out which one works the best for us.



We observe that the Logistic model performs better in this case. Yes, the Decision tree displays better accuracy levels, however it can be observed that the Precision and Recall factors in Logistic model is better than compared to the latter.

Thus we conclude that using the logistic model we can use the given dataset of Taiwan Customer default and the parameters to predict the defaulters with 82% accuracy and with 88% precision at 95% confidence level.

# Revolutionize the market & recommendation

Using machine learning we are trying to bring in the market first instant loan | Credit renewal and pre-approved loans. The algorithm can also be tweaked to inform the customer about the credit limit that can be approved against the asked limit.

In order for this model to perform better we are suggesting the following parameters to be added instead of age,sex,education etc

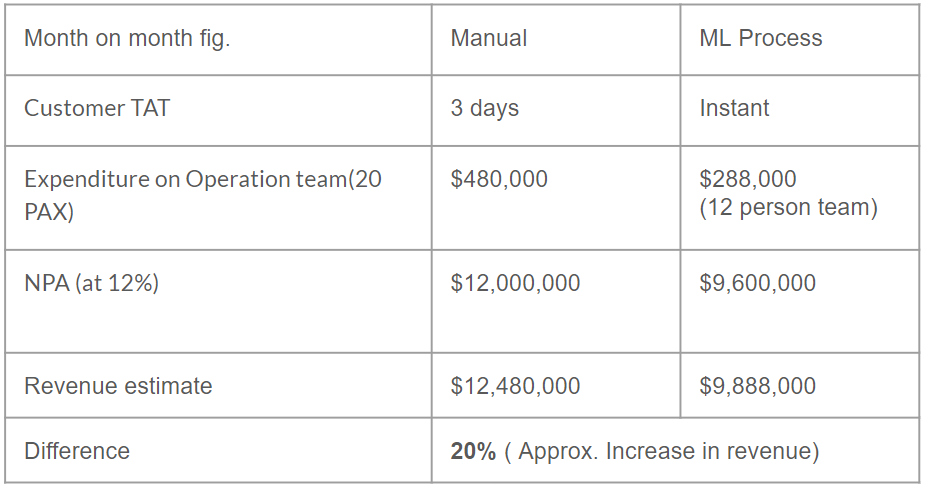
1. Credit score | Social score
2. Ratio of saving vs expenditure
3. Secondary data(cellular device)
4. Connected calls to the customer/activity (lower weightage)

# Business advantage

By implementing these changes we can bring the following changes to the business

1. Automating the decision making process: reduction in NPA by 20% i.e if the current NPA of the nationalized banks are at 12% the model can bring it down to 9.6%
2. Early warning system to alert the recovery team
3. Instant turnaround time and reduced stress on operations team
4. No additional hires in ops team required

Based on these observations we have now laid out a hypothetical revenue and expenditure table. Please note that we have assumed our customer base per year as 100,000 with an average loan per customer as $1,000.

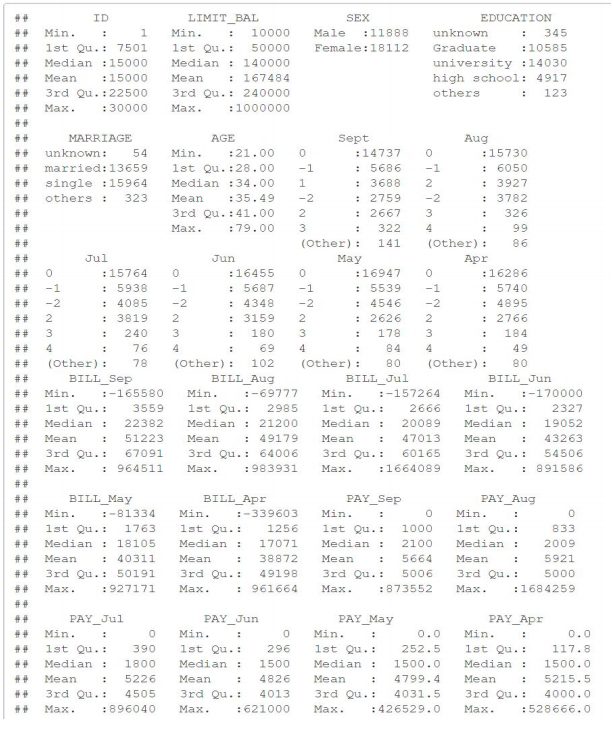


# Appendix

# Univariate observation

This is how our dataset looks like

# 

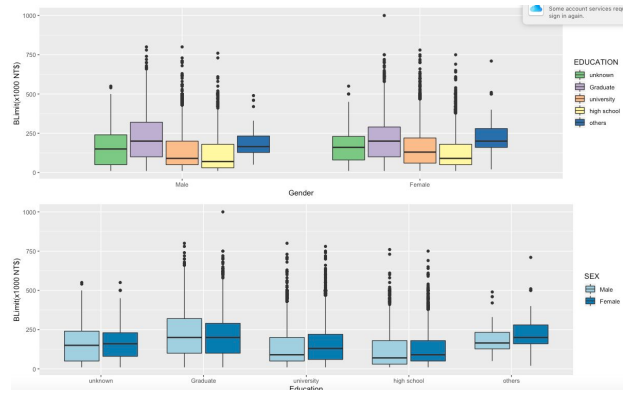


# 

# General observations and graphs

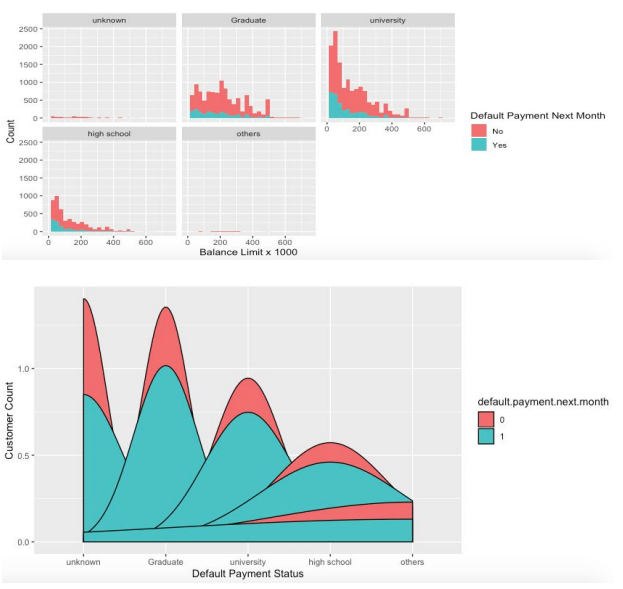
**Education vs balance**

By plotting a boxplot for the variables we can see that education wise there are a lot of outliers or extra line of credit provided to Graduates who are male.



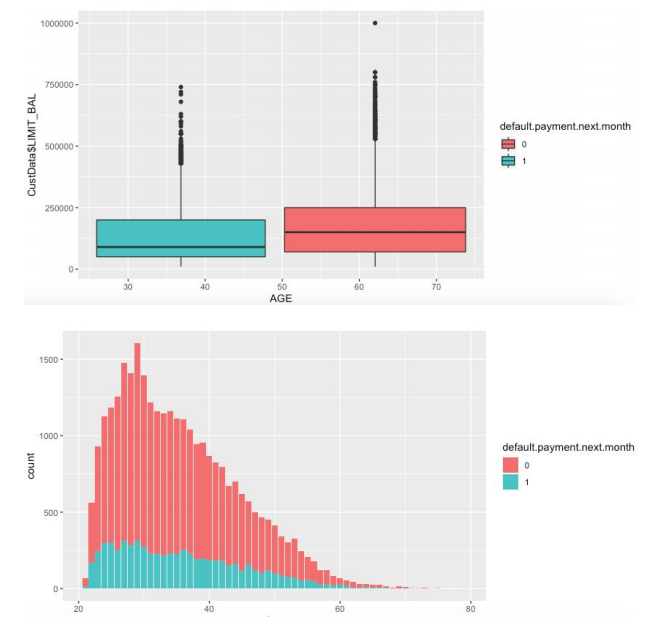
Similarly the male or female gender have almost similarly distributed credit balance wise across all the education qualification.

The defaulters are higher from the university background than compared to any other education qualification.

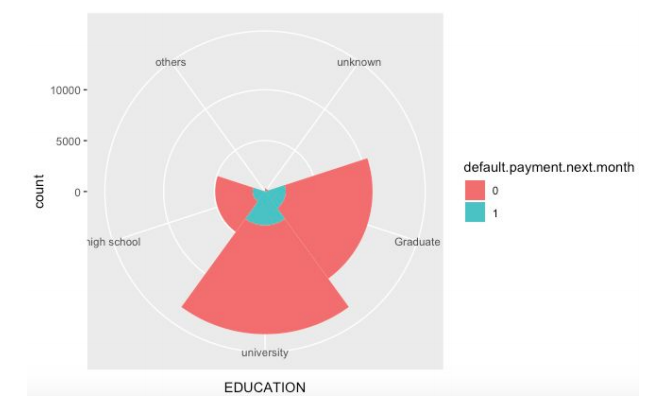


**Range of Age for default**

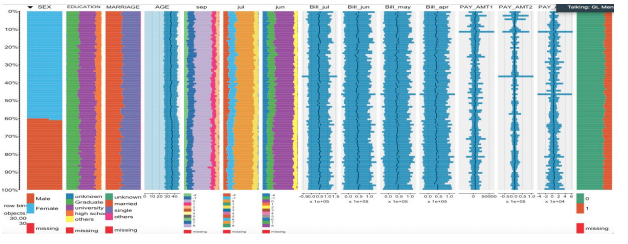
Plotting the graph for this factor we observe that most of the defaulters are from the Age group 21-45 and the default percentage is very less when we look at the higher age, showing better stability economically.



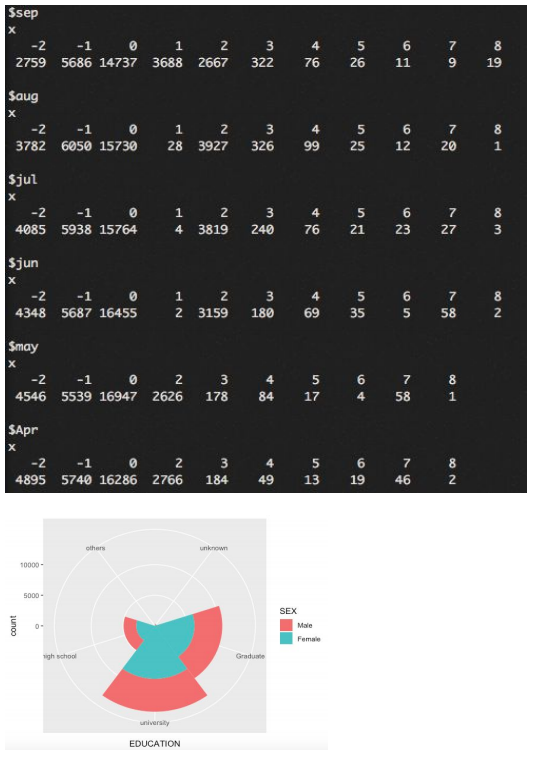
Observe how the defaulters are distributed against the non defaulters. And peculiarly most of them are at a range of 20-45 and then they eventually fade away.



Let’s check on how the payments have been done and lets recheck on how its been distributed. We observe that most of the payments have been done on time however there are very few after 5 months of delay. As we see from the results above, we have very few customers from the 5 or beyond months of delay. Thus we have grouped them under one category.



Let’s see that on display to confirm that post 5 month there are very few cases. Thus we classify all of them under 5 and beyond, to avoid confusion.

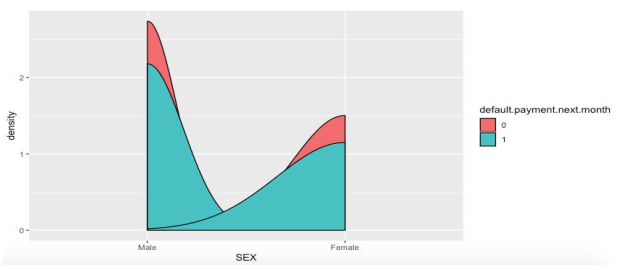


**Marriage status**

Does marital status influence defaults? Well based on the analysis its clear that the marriage status has no impact on the default and thus we can exclude this from the model building stage.



Plotting density plots across gender as well we see that there is no clear demarcation based on gender if one is inclined to default. We can remove this from building our model as well. This shows that there are fewer females compared to males.



# R code

#Set working directory

getwd()

setwd("X:/Workspace")

#Read data to a variable

CustData <- read.csv("Taiwan-Customer defaults.csv", header = TRUE)

str(CustData)

View(CustData)

library(corrplot)

library(ggplot2)

library(psych)

library(dplyr)

library(car)

library(nFactors)

attach(CustData)

###################################################

Custcor=cor(CustData)

corrplot(Custcor, method = "number")

Custcor

corrplot(Custcor)

summary(CustData)

###################################################

(imbalance <- table(CustData$default.payment.next.month))

###########################################

#Converting selected variables to factor and then defining levels for subsequent factors.

CustData$SEX<-as.factor(CustData$SEX)

levels(CustData$SEX) <- c("Male", "Female")

CustData$EDUCATION <- as.factor(CustData$EDUCATION)

levels(CustData$EDUCATION) <- c("unknown","Graduate",

"university",

"high school",

"others",

"unknown",

"unknown")

CustData$MARRIAGE <- as.factor(CustData$MARRIAGE)

levels(CustData$MARRIAGE) <- c("unknown","married"

,"single","others")

View(CustData)

CustData$Sept <- as.factor(CustData$Sept)

CustData$Aug <- as.factor(CustData$Aug)

CustData$Jul <- as.factor(CustData$Jul)

CustData$Jun <- as.factor(CustData$Jun)

CustData$May <- as.factor(CustData$May)

CustData$Apr <- as.factor(CustData$Apr)

CustData$default.payment.next.month <- as.factor(CustData$default.payment.next.month)

View(CustData)

#Checking the imbalance in data

(imbalance <- table(CustData$default.payment.next.month))

View(CustData)

#Using str and summary for deeper understanding of the variables

head(CustData)

str(CustData)

summary(CustData)

dim(CustData)

#For the ease of using variables

attach(CustData)

#Checking if we have any NA's in the data, sneaky guys that these guys are, we need a special check for them.

#Yay! No Na's.

sum(is.na(CustData))

#Loading the library for plots

library(dplyr)

library(ggplot2)

#Checking distributions by Gender

ggplot(data=CustData,aes(x=CustData$SEX, fill=SEX)

)+geom\_histogram(stat="count")+

labs(title="By Gender", x="Gender", fill="Gender")+

scale\_fill\_manual(values=c("RED", "Green"))+

theme(axis.text.x = element\_text(angle=90,hjust=1))

#Checking distribution by education level

ggplot(data=CustData, aes(x=EDUCATION,fill=EDUCATION))+

geom\_histogram(stat="count")+

labs(title="By Education",

x="Education level", fill="Education level")+

scale\_fill\_manual(values=c("orange", "Yellow", "purple","green", "black"))+

theme(axis.text.x = element\_text(angle=90, hjust=1))

#check the general distribution based on Marital status

ggplot(data=CustData, aes(x=MARRIAGE,fill=MARRIAGE))+

geom\_histogram(stat="count")+

labs(title=" Marriage wise distribution",

x="Marital status", fill="Marital status")+

theme(axis.text.x = element\_text(angle=90, hjust=1))

#Lets check the type of audience we are catering to?

ggplot(data=CustData, aes(x=EDUCATION,fill=SEX))+

geom\_bar(position='dodge')+

labs(title = "Distribution by Education & Gender",

x ="EDUCATION",fill = "GENDER")+

scale\_fill\_manual(values=c("pink", "violet"))+

theme(axis.text.x = element\_text(angle = 90,hjust=1))

#How are the categories weigh up to defaults?

ggplot(data=CustData, aes(x=SEX,

fill=default.payment.next.month))+

geom\_histogram(stat="count")+

labs(title=" Gender vs Default ",

x="Gender", fill="Default")+

theme(axis.text.x = element\_text(angle=45, hjust=1))

#Let's check on how marital status has an impact on the defaulters?

ggplot(data=CustData, aes(x=MARRIAGE,

fill=default.payment.next.month))+

geom\_histogram(stat="count")+

labs(title=" Marital status vs Default ",

x="Marital status", fill="Default")+

theme(axis.text.x = element\_text(angle=45, hjust=1))

#DoesEducation & Gender have a pattern when it comes to Defaulting?

ggplot(data=CustData, aes(x=EDUCATION,fill=default.payment.next.month))+

geom\_histogram(position='dodge', stat="count")+

facet\_grid(SEX~.)+

labs(title = "Distribution by Education & Gender", x ="EDUCATION",fill = "Default")+

scale\_fill\_manual(values=c("violet", "Pink"))+

theme(axis.text.x = element\_text(angle = 90,hjust=1))

#Does Age and gender have anything to do with defaults?

ggplot(data=CustData, aes(x=AGE,fill=default.payment.next.month))+

geom\_histogram( stat="count")+

facet\_grid(EDUCATION~SEX)+

labs(title = "Distribution of defaulters by Age", x ="EDUCATION",fill = "Default")+

scale\_fill\_manual(values=c("Orange", "Yellow"))+

theme(axis.text.x = element\_text(angle = 45,hjust=1))

#Dividing into two subsets to do EDA based on the default:: NOTE:: We have used this subsets to understand in depth the charecteristics of the group of defaulters and non defaulters.

NoDefaults <- CustData %>%

subset(default.payment.next.month==0)

View(NoDefaults)

YesDefaults <- CustData %>%

subset(default.payment.next.month==1)

#Plot to check the range of the age group defaulting: NOTE:: This data only includes people who have defaulted.

ggplot(data = YesDefaults, aes(x = AGE)) +

geom\_histogram(bins = 50, fill = "orange",

col = "blue", alpha = 0.3) +

scale\_x\_continuous(breaks = seq(min(0), max(90), by = 5), na.value = TRUE)

#Credit limit given across age, what better way than to plot a scatter plot (For Default)

ggplot(data = CustData, aes(x = AGE, y = LIMIT\_BAL))+

geom\_count(col="magenta", show.legend=F) +

labs(title="Counts Plot",

subtitle="Age Vs Credit Amount",

caption="source: Taiwan customer default")

library(GGally)

ggcorr(CustData, geom = "blank", label = TRUE, hjust = 1) +

geom\_point(size = 10, aes(color = coefficient > 0, alpha = abs(coefficient) > 0.5)) +

scale\_alpha\_manual(values = c("TRUE" = 0.6, "FALSE" = 0)) +

guides(color = FALSE, alpha = FALSE)+

ggtitle("Correlation Plot") +

theme(axis.text.x=element\_blank())

#Making feature boxplot across all the variables

boxplot(CustData)

#Let's check the overall default pattern in our data

ggplot(CustData, aes(y= LIMIT\_BAL,

x= default.payment.next.month))+

geom\_boxplot(notch = FALSE,

outlier.shape=1,

outlier.colour="red"

)+

stat\_summary(fun.y = mean, geom="line",aes(group=1))+

stat\_summary(fun.y = mean, geom="point")

#Checking the overall customer repayment history then we compare this with customers who default.

#For the love of programming

pay.cols.names <- c("Apr", "May", "Jun", "Jul",

"Aug", "Sept")

library(Rmisc)

library(ggpubr)

library(gridExtra)

library(grid)

#Defining a function which will write these 6 graphs. Post that we will use a graph aggregation function to list all them all together.

pay.histograms <-

lapply(pay.cols.names,

function (pay.col.name){

ggplot(data = CustData[, pay.cols.names],

aes(x = CustData[, pay.col.name],fill=CustData$default.payment.next.month)) +

geom\_bar(stat = "count") +

theme\_minimal() +

theme(legend.position = "none")+

xlab(paste0("Repayment status ", pay.col.name))+

ylab("Observations count with Defaults")

})

multiplot(plotlist=pay.histograms, cols = 3)

#checking Education profiles with respect to Default next month

ggplot(data=CustData, aes(x=EDUCATION,

fill=default.payment.next.month))+

geom\_histogram(stat="count")+

labs(title=" Education vs Default ",

x="Education", fill="Default")+

theme(axis.text.x = element\_text(angle=45, hjust=1))

#Fun fact: Violin graph is a way to represent numeric data. It is somehow similar to that of a boxplot but with a rotated density plot on each side.

#Violin plot provide us the best of both the world and thus, we shall hop onto to that for the analysis.

#plotting a violin graph between Education and Credit Amount to understand the density of data

ggplot(data=CustData, aes(x=EDUCATION, y=LIMIT\_BAL,

color= EDUCATION,

fill= EDUCATION))+

geom\_violin()+

labs(title= "Violin plot",

subtitle = "Education vs Balance limit",

caption= "source:Taiwan customer default ")+

theme(axis.text.x = element\_text(angle=45, vjust=0.6))

ggplot(data=CustData, aes(x=MARRIAGE, y=LIMIT\_BAL,

color= MARRIAGE,

fill= MARRIAGE))+

geom\_violin()+

labs(title= "Violin plot",

subtitle = "Marital status vs Balance limit",

caption= "source:Taiwan customer default")+

theme(axis.text.x = element\_text(angle=45, vjust=0.6))

ggplot(data = CustData,aes(x = default.payment.next.month, y = LIMIT\_BAL,

fill = default.payment.next.month,

color = default.payment.next.month )) +

geom\_violin() +

labs(title="Violin Plot",

subtitle="Default Payment Vs Credit Amount",

caption="source: Taiwan customer defaults")

#Lets see if there are any overlapping of the defaulters wrt to the non defaulters. We use this graph in the future as a checkopint to ensure that the model built is able to distinguish, thus a graph with no overlapping will mean that there is a clear distinction between the defaulters and non defaulters.

library(caret)

featurePlot(x = LIMIT\_BAL,

y = default.payment.next.month,

plot = "density",

auto.key = T)

# Balance limits by gender and education

d1 <- ggplot(CustData, aes(SEX, (LIMIT\_BAL/1000), fill=EDUCATION)) +

geom\_boxplot() +

xlab("Gender") +

ylab("BLimit(x1000 NT$)") +

scale\_fill\_brewer(palette = "Accent")

# Balance limits by education and gender

d2 <- ggplot(CustData,aes(EDUCATION, (LIMIT\_BAL/1000), fill=SEX)) +

geom\_boxplot() +

xlab("Education") +

ylab("BLimit(x1000 NT$)") +

scale\_fill\_brewer(palette = "Paired")

grid.arrange(d1, d2)

ggplot(aes(x = CustData$LIMIT\_BAL/1000), data = CustData) +

geom\_histogram(aes(fill = CustData$default.payment.next.month)) +

xlab("Balance Limit x 1000") +

ylab("Count") +

scale\_fill\_discrete(name="Default Payment Next Month",

breaks=c(0, 1),

labels=c("No", "Yes")) +

xlim(c(0,750)) +

facet\_wrap(~EDUCATION)

PAY\_VAR<-lapply(CustData[,c("Sept","Aug","Jul","Jun","May","Apr")], function(x) table(x))

print(PAY\_VAR)

Outlier<-data.frame(apply(CustData[,c("LIMIT\_BAL","BILL\_Sep","BILL\_Aug","BILL\_Jul","BILL\_Jun","BILL\_May","BILL\_Apr",

"PAY\_Sep","PAY\_Aug","PAY\_Jul","PAY\_Jun","PAY\_May","PAY\_Apr")],

2, function(x) quantile(x, probs = seq(0, 1, by= 0.00001))))

head(Outlier)

tail(Outlier)

CustData<-subset(CustData,!(CustData$LIMIT\_BAL> quantile(CustData$LIMIT\_BAL, 0.99999) |

CustData$BILL\_Apr< quantile(CustData$BILL\_Apr, 0.00001)))

#Collinearity check

numeric\_fields<-c("LIMIT\_BAL","BILL\_Sep","BILL\_Aug","BILL\_Jul",

"BILL\_Jun","BILL\_May","BILL\_Apr",

"PAY\_Sep","PAY\_Aug","PAY\_Jul","PAY\_Jun","PAY\_May","PAY\_Apr")

CustData\_numeric<-subset(CustData, select=numeric\_fields)

str(CustData\_numeric)

library(DataExplorer)

plot\_correlation(CustData\_numeric)

CustData\_numeric <- lapply(CustData\_numeric, as.numeric)

str(CustData\_numeric)

library(car)

library(psych)

library(usdm)

vif(CustData\_numeric)

CustData$PAY\_RATIO\_APR<-ifelse(is.nan(CustData$PAY\_Apr/CustData$BILL\_Apr),0,

ifelse(is.infinite(CustData$PAY\_Apr/CustData$BILL\_Apr),

0,round(CustData$PAY\_Apr/CustData$BILL\_Apr,2)))

CustData$PAY\_RATIO\_MAY<-ifelse(is.nan(CustData$PAY\_May/CustData$BILL\_May),0,

ifelse(is.infinite(CustData$PAY\_May/CustData$BILL\_May),0,round(CustData$PAY\_May/CustData$BILL\_May,2)))

CustData$PAY\_RATIO\_JUNE<-ifelse(is.nan(CustData$PAY\_Jun/CustData$BILL\_Jun),0,

ifelse(is.infinite(CustData$PAY\_Jun/CustData$BILL\_Jun),0,round(CustData$PAY\_Jun/CustData$BILL\_Jun,2)))

CustData$PAY\_RATIO\_JULY<-ifelse(is.nan(CustData$PAY\_Jul/CustData$BILL\_Jul),0,

ifelse(is.infinite(CustData$PAY\_Jul/CustData$BILL\_Jul),0,round(CustData$PAY\_Jul/CustData$BILL\_Jul,2)))

CustData$PAY\_RATIO\_AUG<-ifelse(is.nan(CustData$PAY\_Aug/CustData$BILL\_Aug),0,

ifelse(is.infinite(CustData$PAY\_Aug/CustData$BILL\_Aug),0,round(CustData$PAY\_Aug/CustData$BILL\_Aug,2)))

CustData$PAY\_RATIO\_SEPT<-ifelse(is.nan(CustData$PAY\_Sep/CustData$BILL\_Sep),0,

ifelse(is.infinite(CustData$PAY\_Sep/CustData$BILL\_Sep),0,round(CustData$PAY\_Sep/CustData$BILL\_Sep,2)))

numeric\_fields<-c("LIMIT\_BAL","BILL\_Apr","BILL\_May","BILL\_Jun","BILL\_Jul","BILL\_Aug",

"BILL\_Sep","PAY\_RATIO\_APR","PAY\_RATIO\_MAY","PAY\_RATIO\_JUNE",

"PAY\_RATIO\_JULY","PAY\_RATIO\_AUG","PAY\_RATIO\_SEPT",

"PAY\_Apr","PAY\_May","PAY\_Jun","PAY\_May","PAY\_Jun","PAY\_Jul", "PAY\_Sep")

View(CustData)

CustData\_numeric<-subset(CustData, select = numeric\_fields)

plot\_correlation(CustData\_numeric)

ggplot(data=CustData,mapping = aes(x=AGE,y=CustData$LIMIT\_BAL,

fill=default.payment.next.month)) + geom\_boxplot()

library(ggthemes)

ggplot(data=CustData, mapping = aes(x=MARRIAGE,

fill=default.payment.next.month)) + geom\_bar()+theme\_few()

ggplot(data=CustData, mapping = aes(x=SEX, fill=default.payment.next.month)) + geom\_bar()+theme\_few()

ggplot(data=CustData,mapping = aes(x=EDUCATION,

y=CustData$LIMIT\_BAL,fill=default.payment.next.month)) + geom\_boxplot()

ggplot(CustData, aes(x = SEX, fill = default.payment.next.month)) + geom\_density()

ggplot(CustData, aes(x = EDUCATION, fill = default.payment.next.month)) + geom\_density() +

xlab("Default Payment Status") + ylab("Customer Count")

CustData %>% group\_by(EDUCATION,AGE) %>% summarise(mn\_creditlmt=mean(LIMIT\_BAL)) -> df

ggplot(df, aes(x=EDUCATION, y=AGE, fill=mn\_creditlmt)) + geom\_tile() + scale\_fill\_gradient(low="white", high="steelblue")

ggplot(CustData, aes(x = AGE, fill = default.payment.next.month)) +

geom\_bar() +

labs(x = 'Age')

ggplot(CustData, aes(x=EDUCATION, fill = default.payment.next.month))+

geom\_bar(width = 1)+

coord\_polar()

ggplot(CustData, aes(x=EDUCATION, fill = SEX))+

geom\_bar(width = 1)+

coord\_polar()

qplot(AGE, data = CustData, geom = "density", fill = default.payment.next.month)

qplot(SEX, data = CustData, geom = "density", fill = EDUCATION)

qplot(AGE, data = CustData, geom = "density", fill = SEX)

# Trying to see even in such sparse data

featurePlot(x = CustData[, c(3,4,5,2)], # MathsMarks & scienceteacher

y = CustData$AGE, # Develop relationship with y

plot = "pairs", # Plot in pairs

auto.key = T # Show legend

)

library(ff)

library(ggplot2)

library(tabplot)

tableplot(CustData,

#sortCol = AGE, # Sorted on target variable

select = c(3:7,9,10,15:21,25)) # How these behave vis-a-vis target

tableplot(CustData,

#sortCol = AGE, # Target variable

select = c(1:6,11:14,21,25))

#Building models and diving the dataset into training and test data

set.seed(101)

#Train and Test Data

train\_df <- sample\_frac(CustData, 0.75)

test\_df <- subset(CustData, !(CustData$ID %in% train\_df$ID))

#Imbalance check in the dependent variable for the Train and Test data

Imbalance\_Check\_train\_df<-aggregate(ID ~ default.payment.next.month,train\_df,length)

colnames(Imbalance\_Check\_train\_df)[2]<-"Client\_Count"

Imbalance\_Check\_train\_df$Contribution<-(Imbalance\_Check\_train\_df$Client\_Count/sum(Imbalance\_Check\_train\_df$Client\_Count))\*100

Imbalance\_Check\_train\_df

Imbalance\_Check\_test\_df<-aggregate(ID~default.payment.next.month,test\_df,length)

colnames(Imbalance\_Check\_test\_df)[2]<-"Client\_Count"

Imbalance\_Check\_test\_df$Contribution<-(Imbalance\_Check\_test\_df$Client\_Count/sum(Imbalance\_Check\_test\_df$Client\_Count))\*100

Imbalance\_Check\_test\_df

#Logistic regression

mod\_l <- glm(default.payment.next.month~. -(ID+BILL\_Apr+BILL\_May+BILL\_Jun+BILL\_Jul+BILL\_Aug

+BILL\_Sep+SEX+MARRIAGE), train\_df, family=binomial)

summary(mod\_l)

predict\_l <- predict(mod\_l, train\_df, type='response')

prob\_l <- ifelse(predict\_l > 0.5,1,0)

#Confusion matrix

confusion\_matrix<-table(prob\_l, train\_df$default.payment.next.month)

print(confusion\_matrix)

#model Accuracy

Accuracy <- sum(diag(confusion\_matrix))/sum(confusion\_matrix)

print(Accuracy\*100)

library(prediction)

library(ROCR)

pred0 <- prediction(predict\_l, train\_df$default.payment.next.month)

pred0 <- performance(pred0, "tpr", "fpr")

plot(pred0)

#Step AIC criterion

step\_AIC <- stepAIC(mod\_l, direction='backward')

#Re-running the model with the best subset obtained

mod\_2 <- glm(default.payment.next.month ~ LIMIT\_BAL+ EDUCATION + PAY\_Apr+ PAY\_May

+PAY\_Jun+PAY\_Jul+ PAY\_Aug+ PAY\_Sep +Apr + May + Jun+ Jul + Aug + Sept,

train\_df, family='binomial')

summary(mod\_2)

predict\_2<-predict(mod\_2,train\_df,type='response')

prob\_2<-ifelse(predict\_2>0.5,1,0)

#Confusion Matrix

confusion\_matrix<-table(prob\_2, train\_df$default.payment.next.month)

print(confusion\_matrix)

#Model Accuracy

Accuracy<-sum(diag(confusion\_matrix))/sum(confusion\_matrix)

print(Accuracy\*100)

#The ROCR Curve

library(prediction)

library(ROCR)

pred1 <- prediction(predict(mod\_2), train\_df$default.payment.next.month)

pred1 <- performance(pred1, "tpr", "fpr")

plot(pred1)

auc.tmp<- performance(predict(mod\_2),"auc")

#There isn't any significant change in accuracy of the

#model so we can assume our model performs well. But can we

#raise the accuracy of the model using some other technique?

#So let us try the Decision Tree approach

#Decision tree

library(rpart)

set.seed(1234)

mymod<-rpart(default.payment.next.month ~ PAY\_Apr+PAY\_May+PAY\_Jun+PAY\_Jul+PAY\_Aug+PAY\_Sep+LIMIT\_BAL+

EDUCATION+Sept+Aug+Jul+Jun+May+Apr+

BILL\_Sep+BILL\_Aug+BILL\_Jul+BILL\_Jun+BILL\_May+BILL\_Apr+AGE+SEX+MARRIAGE,

data= train\_df, method="class",

control = rpart.control(cp = 0.0001,minsplit = 30, minbucket = 30\*2,

maxsurrogate = 5, usesurrogate = 2,xval=10, maxdepth = 30))

printcp(mymod)

#Let's have a look at the variable importance and drop

#the variables that are not of much importance.

mymod$variable.importance

#Let's re-run the model after dropping the

#variables "EDUCATION","SEX","MARRIAGE"

mymod<-rpart(default.payment.next.month ~ PAY\_Apr+PAY\_May+PAY\_Jun+PAY\_Jul+PAY\_Aug+PAY\_Sep+LIMIT\_BAL+

EDUCATION+Sept+Aug+Jul+Jun+May+Apr+

BILL\_Sep+BILL\_Aug+BILL\_Jul+BILL\_Jun+BILL\_May+BILL\_Apr+AGE,

data= train\_df, method="class",

control = rpart.control(cp = 0.0001,minsplit = 30, minbucket = 30\*2,

maxsurrogate = 5, usesurrogate = 2,xval=10, maxdepth = 30))

printcp(mymod)

#We will prune the tree based on the best complexity

#parameter that has the least error.

set.seed(1234)

bestcp <- mymod$cptable[which.min(mymod$cptable[,"xerror"]),"CP"]

# Prune the tree using the best cp.

tree.pruned <- prune(mymod, cp = bestcp)

#Let us evaluate and the model accuracy

# confusion matrix (training data)

conf.matrix <- table("Predicted"=predict(tree.pruned,type="class"),"Actual"=train\_df$default.payment.next.month)

print(conf.matrix)

Accuracy\_dt<-sum(diag(conf.matrix))/sum((conf.matrix))

print(Accuracy\_dt \* 100)

library(rpart.plot)

rpart.plot(mymod,fallen.leaves = F ,extra=3)

conf.matrix\_test <- table("Predicted"=predict(tree.pruned,test\_df, type="class"),"Actual"=test\_df$default.payment.next.month)

print(conf.matrix\_test)

Accuracy\_test<-sum(diag(conf.matrix\_test))/sum((conf.matrix\_test))

print(Accuracy\_test \* 100)

#The accuracy achieved using the data set aside

#for validation is also similar hence our model

#fits the data well. Overall, we see some improvement in

#the accuracy of the model using the Decision Tree approach.

#Let's try random forest to understand

#if any further imrovement can be brought to the model.

seed<-7

mtry <- floor(sqrt(ncol(train\_df[,c("PAY\_Sep","PAY\_Aug","PAY\_Jul","PAY\_Jun","PAY\_May",

"PAY\_Apr","LIMIT\_BAL","EDUCATION","Sept","Aug","Jul",

"Jun","May","Apr","BILL\_Sep","BILL\_Aug","BILL\_Jul",

"BILL\_Jun","BILL\_May","BILL\_Apr","SEX","AGE","MARRIAGE")])))

metric <- "Accuracy"

control <- trainControl(method="repeatedcv", number=10, repeats=1, search="grid")

tunegrid <- expand.grid(.mtry=c(1:mtry))

modellist <- list()

for (ntree in c(500,1000, 1500, 2000, 2500)) {

set.seed(seed)

fit <- train(default.payment.next.month~.-(ID+PAY\_RATIO\_APR+PAY\_RATIO\_MAY+PAY\_RATIO\_JUNE+

PAY\_RATIO\_JULY+PAY\_RATIO\_AUG+PAY\_RATIO\_SEPT),

data=train\_df, method="rf", metric=metric,

tuneGrid=tunegrid, trControl=control, ntree=ntree)

key <- toString(ntree)

modellist[[key]] <- fit

}

# compare results

results <- resamples(modellist)

summary(results)

dotplot(results)