Personal Loan Campaign Report 2019

Aug 10, 2019

Personal Loan Campaign Report

Authored By: Deepti Lobo

CONTENTS

Project Objective

This case is about a bank (Thera Bank) which has a growing customer base. Majority of these customers are liability customers (depositors) with varying size of deposits. The number of customers who are also borrowers (asset customers) is quite small, and the bank is interested in expanding this base rapidly to bring in more loan business and in the process, earn more through the interest on loans. In particular, the management wants to explore ways of converting its liability customers to personal loan customers (while retaining them as depositors). A campaign that the bank ran last year for liability customers showed a healthy conversion rate of over 9% success. This has encouraged the retail marketing department to devise campaigns with better target marketing to increase the success ratio with a minimal budget. The department wants to build a model that will help them identify the potential customers who have a higher probability of purchasing the loan. This will increase the success ratio while at the same time reduce the cost of the campaign.

Known Facts

The dataset has data on 5000 customers. The data include customer demographic information (age, income, etc.), the customer's relationship with the bank (mortgage, securities account, etc.), and the customer response to the last personal loan campaign (Personal Loan). Among these 5000 customers, only 480 (= 9.6%) accepted the personal loan that was offered to them in the earlier campaign.

Exploratory Data Analysis (EDA)

The given dataset consists of 5000 observations and 14 variables.

```
#Renamed the columns

names(theraDS)
```

```
##  [1] "ID"                "Age"                "Experience"
##  [4] "Income"            "ZIP_Code"           "Family_members"
##  [7] "CCAvg"             "Education"          "Mortgage"
## [10] "Personal_Loan"     "Securities_Account" "CD_Account"
## [13] "Online"            "CreditCard"
```

Attributes Details

Nominal variables:

- **ID** - The customer's ID does not provide any useful insights. There is no correlation between the ID and the loan. We can neglect this variable for our model prediction.

- **Zip Code** -Home Address ZIP code.

Ordinal variables:

- **Family Members** - Family size of the customer
- **Education** - The education Level of the customer. 1: Undergrad; 2: Graduate; 3: Advanced/Professional

Interval variables:

- **Age** - Customer's age in years.
- **Experience** - The years of professional experience of the customer.
- **Income** - The annual income of the customer ($000).
- **CCAvg** - The avg. spending on credit cards per month ($000).
- **Mortgage** - The value of the house mortgage if any. ($000)

Binary Variables:

- **Personal Loan** - Did this customer accept the personal loan offered in the last campaign? We will consider this as our target variable.
- **Securities Account** - Does the customer have a securities account with the bank?
- **CD Account** - Does the customer have a certificate of deposit (CD) account with the bank?
- **Online** - Does the customer use internet banking facilities?
- **Credit Card** - Does the customer use a credit card issued by the bank?

```
#Display the first six rows
head(theraDS)
```

```
##   ID Age Experience Income ZIP_Code Family_members CCAvg Education
##   1  25          1     49    91107              4   1.6         1
##   2  45         19     34    90089              3   1.5         1
##   3  39         15     11    94720              1   1.0         1
##   4  35          9    100    94112              1   2.7         2
##   5  35          8     45    91330              4   1.0         2
##   6  37         13     29    92121              4   0.4         2
##   Mortgage Personal_Loan Securities_Account CD_Account Online CreditCard
##          0             0                  1          0      0          0
##          0             0                  1          0      0          0
##          0             0                  0          0      0          0
##          0             0                  0          0      0          0
```

```
##         0             0                  0          0      0          1
##       155             0                  0          0      1          0
```

```
#Is there any values missing?

colSums(is.na(theraDS))
```

```
##                 ID                Age         Experience
##                  0                  0                  0
##             Income           ZIP_Code     Family_members
##                  0                  0                 18
##              CCAvg          Education           Mortgage
##                  0                  0                  0
##      Personal_Loan Securities_Account         CD_Account
##                  0                  0                  0
##             Online         CreditCard
##                  0                  0
```

The attribute Family Members has 18 NA's. For better analysis, we will be updating this column as 0. Assuming that, family size as 0 means, the person has no dependent and 1 means, he has a single dependent and so on.

Descriptive Analysis

```
#Data types of all the columns
str(theraDS)
```

```
## 'data.frame':    5000 obs. of  14 variables:
##  $ ID                : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ Age               : num  25 45 39 35 35 37 53 50 35 34 ...
##  $ Experience        : num  1 19 15 9 8 13 27 24 10 9 ...
##  $ Income            : num  49 34 11 100 45 29 72 22 81 180 ...
##  $ ZIP_Code          : num  91107 90089 94720 94112 91330 ...
##  $ Family_members    : num  4 3 1 1 4 4 2 1 3 1 ...
##  $ CCAvg             : num  1.6 1.5 1 2.7 1 0.4 1.5 0.3 0.6 8.9 ...
##  $ Education         : num  1 1 1 2 2 2 2 3 2 3 ...
##  $ Mortgage          : num  0 0 0 0 0 155 0 0 104 0 ...
##  $ Personal_Loan     : num  0 0 0 0 0 0 0 0 0 1 ...
##  $ Securities_Account: num  1 1 0 0 0 0 0 0 0 0 ...
##  $ CD_Account        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Online            : num  0 0 0 0 0 1 1 0 1 0 ...
##  $ CreditCard        : num  0 0 0 0 1 0 0 1 0 0 ...
```

```
#Changing Education, Credit Card, Securities Account, CD Account, Online and
Personal Loan as a factor.


#summary of the dataset
summary(theraDS)
```

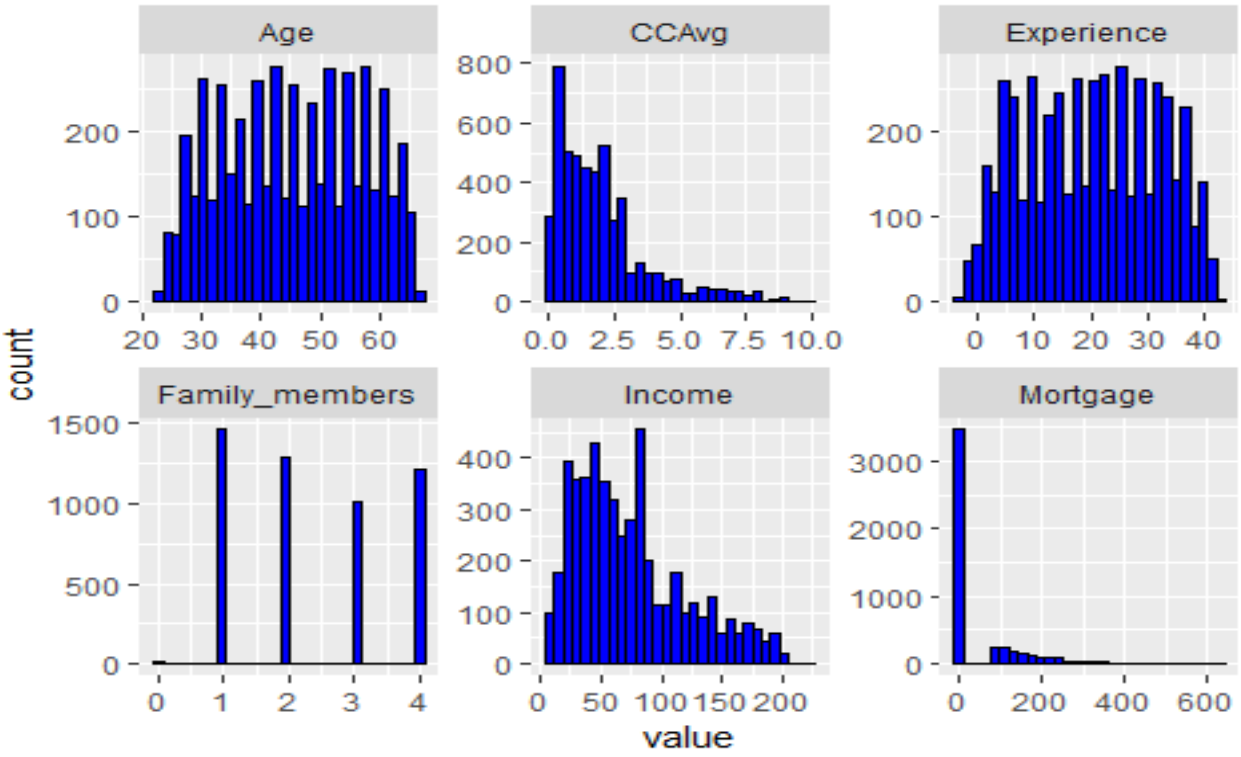
```
##        ID            Age          Experience       Income
##  Min.   :   1   Min.   :23.00   Min.   :-3.0   Min.   :  8.00
##  1st Qu.:1251   1st Qu.:35.00   1st Qu.:10.0   1st Qu.: 39.00
##  Median :2500   Median :45.00   Median :20.0   Median : 64.00
##  Mean   :2500   Mean   :45.34   Mean   :20.1   Mean   : 73.77
##  3rd Qu.:3750   3rd Qu.:55.00   3rd Qu.:30.0   3rd Qu.: 98.00
##  Max.   :5000   Max.   :67.00   Max.   :43.0   Max.   :224.00
##     ZIP_Code     Family_members      CCAvg        Education
##  Min.   : 9307   Min.   :0.000   Min.   : 0.000   1:2096
##  1st Qu.:91911   1st Qu.:1.000   1st Qu.: 0.700   2:1403
##  Median :93437   Median :2.000   Median : 1.500   3:1501
##  Mean   :93153   Mean   :2.389   Mean   : 1.938
##  3rd Qu.:94608   3rd Qu.:3.000   3rd Qu.: 2.500
##  Max.   :96651   Max.   :4.000   Max.   :10.000
##     Mortgage     Personal_Loan Securities_Account CD_Account Online
##  Min.   :  0.0   0:4520        0:4478             0:4698     0:2016
##  1st Qu.:  0.0   1: 480        1: 522             1: 302     1:2984
##  Median :  0.0
##  Mean   : 56.5
##  3rd Qu.:101.0
##  Max.   :635.0
##  CreditCard
##  0:3530
##  1:1470
```

```
#ID, ZIP Code, Family members, Education, Credit Card, Securities Account, CD Account, Online and Personal loan columns have not be considered.
##Summary Statistics Measure of central tendency and dispersion (Univariate Analysis)

describe(theraDS[,c(2,3,4,7,9)],na.rm = TRUE,
         quant = c(0.01,0.05,0.10,0.25,0.75,0.90,0.95,0.99),IQR=TRUE,check=TRUE)

##             vars    n  mean     sd median trimmed   mad min max range  skew
## Age            1 5000 45.34  11.46   45.0   45.38 14.83  23  67    44 -0.03
## Experience     2 5000 20.10  11.47   20.0   20.13 14.83  -3  43    46 -0.03
## Income         3 5000 73.77  46.03   64.0   68.83 43.00   8 224   216  0.84
## CCAvg          4 5000  1.94   1.75    1.5    1.65  1.33   0  10    10  1.60
## Mortgage       5 5000 56.50 101.71    0.0   32.98  0.00   0 635   635  2.10
##             kurtosis   se   IQR Q0.01 Q0.05 Q0.1 Q0.25 Q0.75  Q0.9 Q0.95
## Age            -1.15 0.16  20.0    25  27.0 30.0  35.0  55.0  61.0    63
## Experience     -1.12 0.16  20.0    -1   2.0  4.0  10.0  30.0  36.0    38
## Income         -0.05 0.65  59.0    10  18.0 22.0  39.0  98.0 145.0   170
## CCAvg           2.64 0.02   1.8     0   0.1  0.3   0.7   2.5   4.3     6
## Mortgage        4.75 1.44 101.0     0   0.0  0.0   0.0 101.0 200.0   272
##              Q0.99
## Age          65.00
## Experience   41.00
```

```
## Income     193.00
## CCAvg        8.00
## Mortgage   431.01
```

Data Visualization

```
#Removing ID and Zip Code from the plot

#Histogram for numerical variables (Univariate Analysis)

theraDS[,-c(1,5)] %>% keep(is.numeric) %>% gather() %>%
  ggplot(aes(value)) + facet_wrap(~key, scales = "free") + geom_histogram(col
or = "black", fill = "blue")
```



Observation

- **Age** - Is normally distributed. Most of the customers fall between the age group of 30 to 60 yrs.
- **CCAvg** - Is positively skewed with the spending ranging from 0 to 10K. Most of the spending is less than 2.5K.

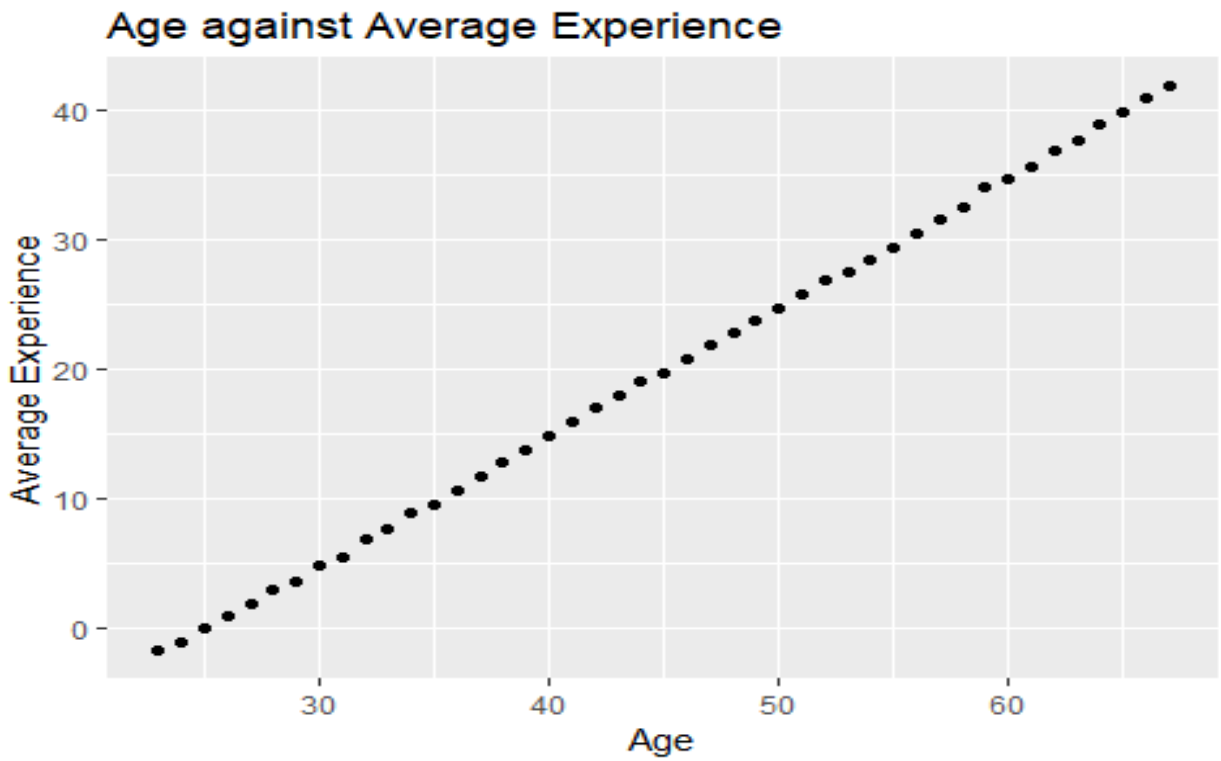
- **Experience** - Is normally distributed with an average experience of 20yrs. The min experience shows as -3yrs. This could be a data input error, as the experience cannot be negative, and it needs to be corrected.
- **Income** - Is positively skewed, with an average income of 73K and max income of 224K. Mortgage - The 50% of the customers have a mortgage of less than 56K. The max mortgage being of 635K.
- **Family_members** - are evenly distributed.
- **Mortgage** – Looks like may have outliers with max value as 0, indicating that majority of the people do not have a mortgage.

CD_Account, CreditCard, Online, Personal_Loan and Securities_Account have binary values of 1 and 0.

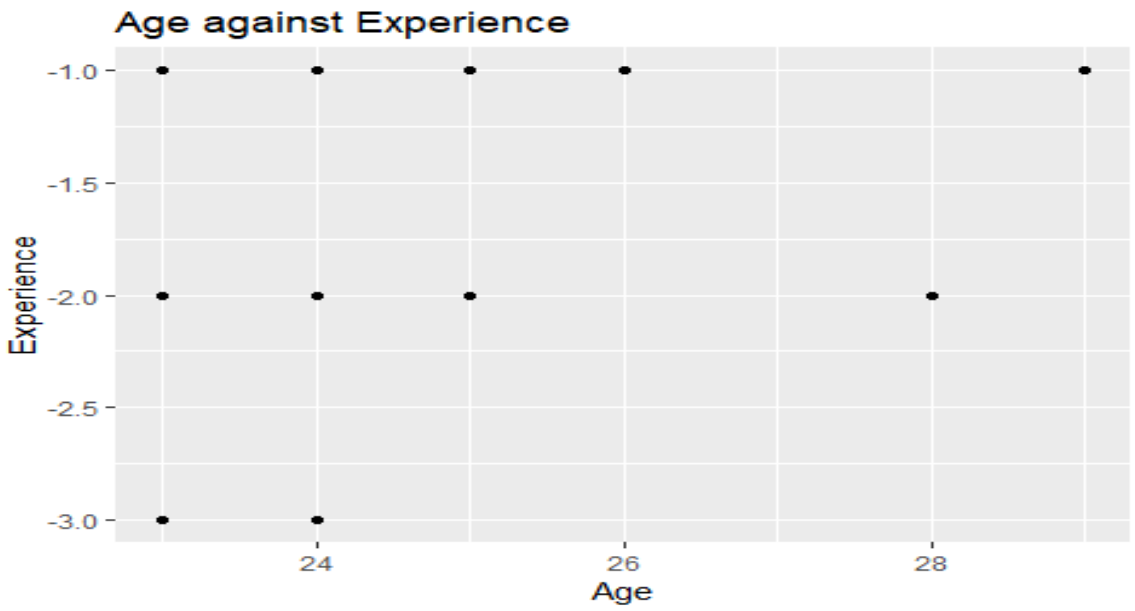
```
##Frequency distribution for categorical variable (Univariate Analysis)

table(theraDS[,c(8)]) #Education

##
##    1    2    3
## 2096 1403 1501

table(theraDS[,c(10)]) #Personal Loan

##
##    0    1
## 4520  480

table(theraDS[,c(11)]) #Securties Accoun

##
##    0    1
## 4478  522

table(theraDS[,c(12)]) #CD Account

##
##    0    1
## 4698  302

table(theraDS[,c(13)]) #Online

##
##    0    1
## 2016 2984

table(theraDS[,c(14)]) #Credit Card
```

```
##
##    0    1
## 3530 1470

#Count of customers having negative experience.
sum(theraDS$Experience < 0)

## [1] 52

#Education Vs Experience
aggregate(theraDS$Experience, list(theraDS$Education), mean)

##   Group.1        x
## 1       1 20.06536
## 2       2 19.77049
## 3       3 20.47169
```

As we can see, there is no much difference in the average of experience based on the education.

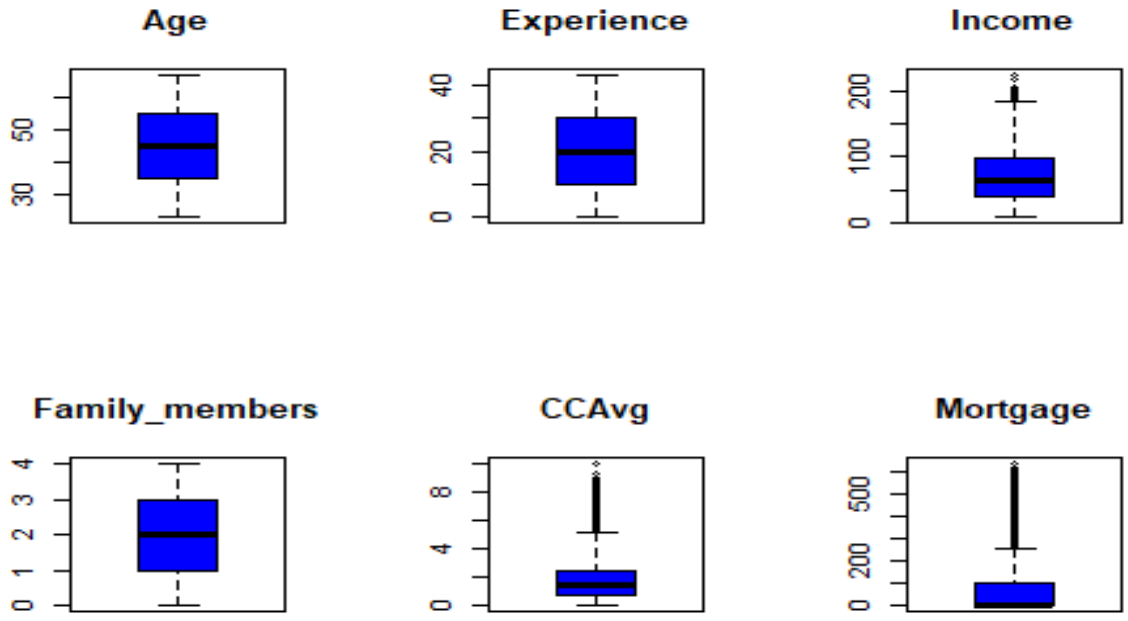


As we can see Experience is linearly proportional to Age.



As we can see the negative range of experience, ranges from 23 to 29 yrs and has the values of -1, -2 and -3. We have seen that experience increases with age. Therefore, we can assume that the negative experience was a data entry error and take its absolute value.

Boxplot



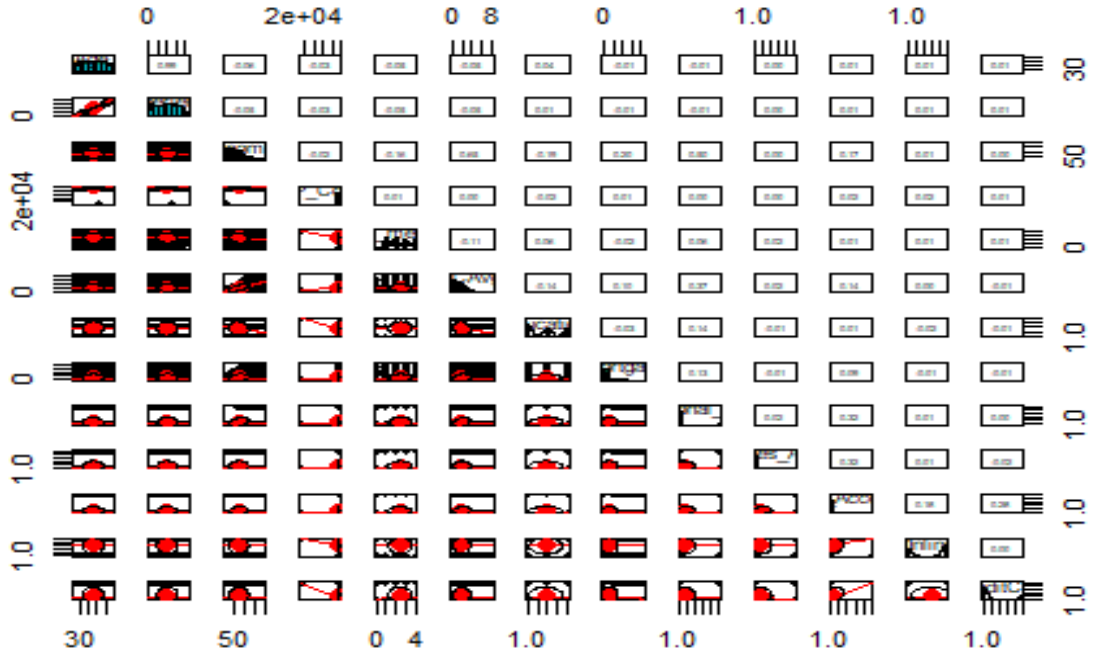
The boxplot shows that Income, CCAvg and Mortgage has outliers.

```
summary(theraDS)
```

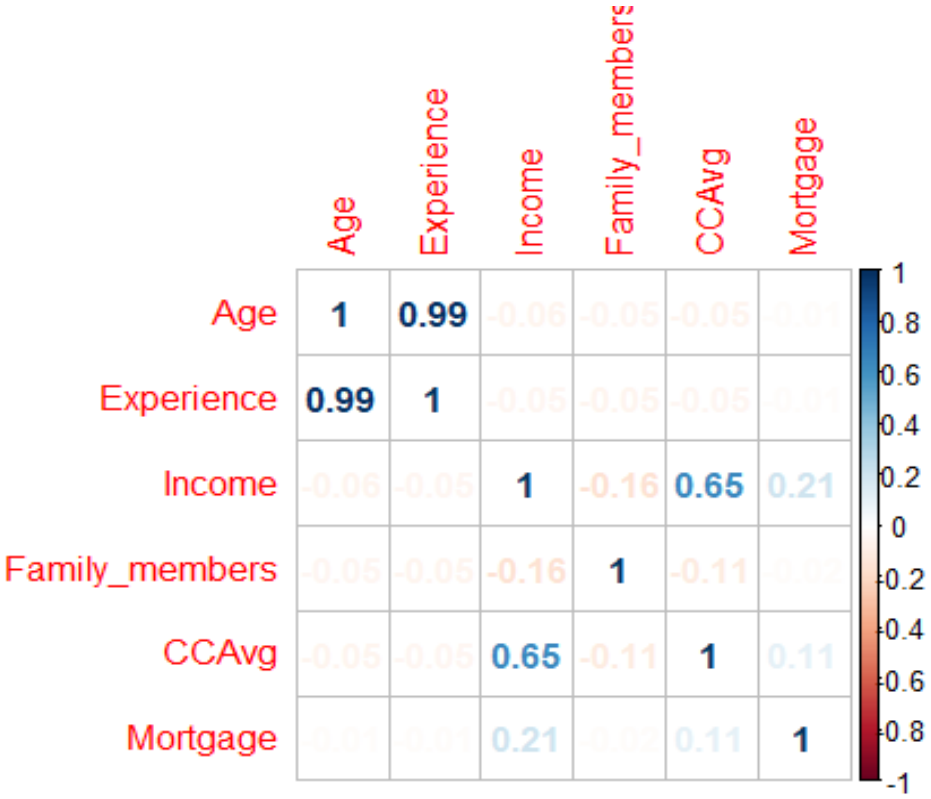
```
##        ID            Age          Experience        Income
##  Min.   :   1   Min.   :23.00   Min.   : 0.00   Min.   :  8.00
##  1st Qu.:1251   1st Qu.:35.00   1st Qu.:10.00   1st Qu.: 39.00
##  Median :2500   Median :45.00   Median :20.00   Median : 64.00
##  Mean   :2500   Mean   :45.34   Mean   :20.13   Mean   : 73.72
##  3rd Qu.:3750   3rd Qu.:55.00   3rd Qu.:30.00   3rd Qu.: 98.00
##  Max.   :5000   Max.   :67.00   Max.   :43.00   Max.   :193.00
##     ZIP_Code     Family_members      CCAvg        Education
##  Min.   : 9307   Min.   :0.000   Min.   :0.000   1:2096
##  1st Qu.:91911   1st Qu.:1.000   1st Qu.:0.700   2:1403
##  Median :93437   Median :2.000   Median :1.500   3:1501
##  Mean   :93153   Mean   :2.389   Mean   :1.933
##  3rd Qu.:94608   3rd Qu.:3.000   3rd Qu.:2.500
##  Max.   :96651   Max.   :4.000   Max.   :8.000
##     Mortgage      Personal_Loan Securities_Account CD_Account Online
##  Min.   :  0.00   0:4520        0:4478             0:4698     0:2016
##  1st Qu.:  0.00   1: 480        1: 522             1: 302     1:2984
##  Median :  0.00
##  Mean   : 55.69
##  3rd Qu.:101.00
##  Max.   :431.01
##  CreditCard
##  0:3530
##  1:1470
```

We have done the outlier treatment for 99 percentiles. Post the outlier treatment, if we see the summary. The max of Income has changed from 224 to 193. Similarly, the max of CCAVG has changed from 10 to 8. The max Mortgage has changed from 635 to 431.01.

Correlation graph based on Pearson for Bivariate Analysis




```
##correlation between continuous variables (Bivariate Analysis)
round(cor(theraDS[,c(2,3,4,6,7,9)]),2)

##                    Age Experience Income Family_members CCAvg Mortgage
## Age               1.00       0.99  -0.06          -0.05 -0.05    -0.01
## Experience        0.99       1.00  -0.05          -0.05 -0.05    -0.01
## Income           -0.06      -0.05   1.00          -0.16  0.65     0.20
## Family_members   -0.05      -0.05  -0.16           1.00 -0.11    -0.02
## CCAvg            -0.05      -0.05   0.65          -0.11  1.00     0.10
## Mortgage         -0.01      -0.01   0.20          -0.02  0.10     1.00
```



From the graph as well as the correlation matrix, we can see that Experience and Age are highly correlated. CCAvg and Income are also correlated.

Chi Square test for Categorical variables (Bivariate Analysis)

Chi-Square test is a statistical method which is used to determine if two categorical variables have a significant correlation between them. Here we are taking a significance level of 0.05 and comparing the p-value to accept or reject the Null Hypothesis.

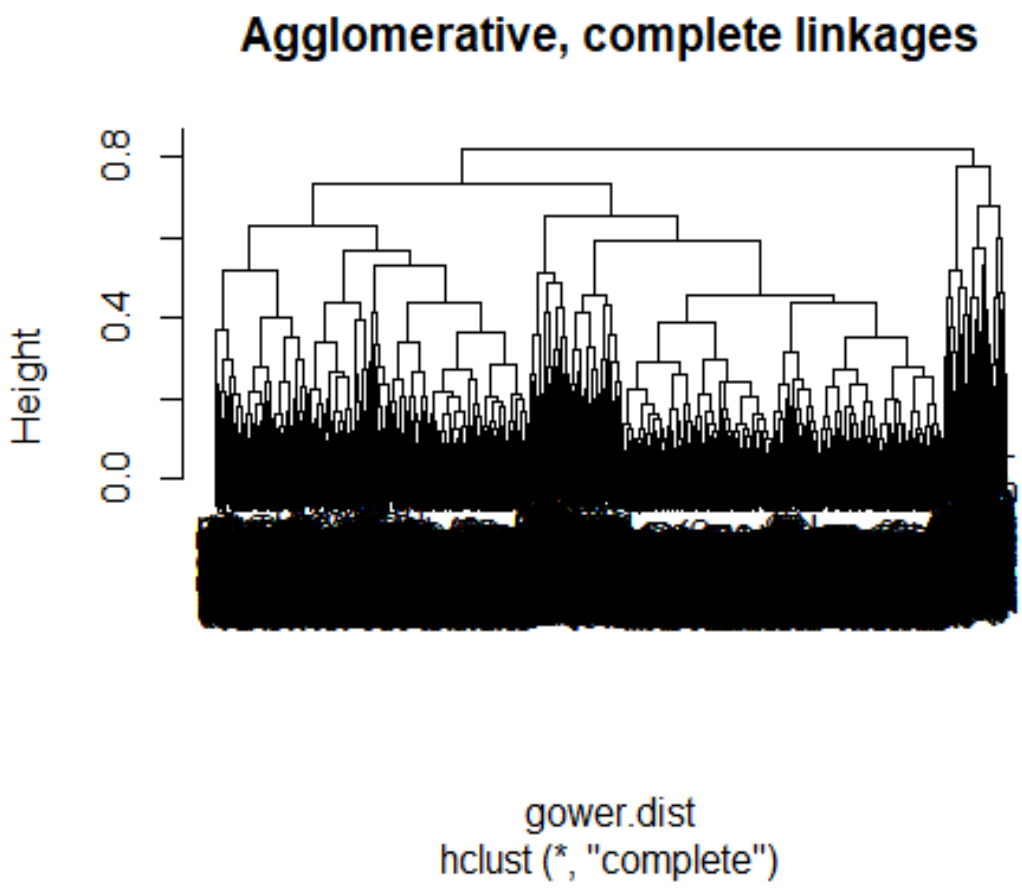
Categorical Variables	P-Value	Analysis
Education and Personal loan	`p-value < 2.2e-16`	Null hypothesis is rejected.
Education and Security Account	`p-value = 0.68`	Failed to reject the Null hypothesis.
Education and CD Account	`p-value = 0.58`	Failed to reject the Null hypothesis.

Education and Online	p-value = 0.1698	Failed to reject the Null hypothesis.
Education and Credit Card	p-value = 0.5377	Failed to reject the Null hypothesis.
Personal loan and Security Account	p-value = 0.1405	Failed to reject the Null hypothesis.
Personal loan and CD Account	p-value < 2.2e-16	Null hypothesis is rejected.
Personal loan and Online	p-value = 0.6929	Failed to reject the Null hypothesis.
Personal loan and Credit Card	p-value = 0.8844	Failed to reject the Null hypothesis.
Security Account and CD Account	p-value < 2.2e-16	Null hypothesis is rejected.
Security Account and Online	p-value = 0.3977	Failed to reject the Null hypothesis.
Security Account and Credit Card	p-value = 0.3116	Failed to reject the Null hypothesis.
CD Account and Online	p-value < 2.2e-16	Null hypothesis is rejected.
CD Account and Credit Card	p-value < 2.2e-16	Null hypothesis is rejected.
Online and Credit Card	p-value = 0.7902	Failed to reject the Null hypothesis.

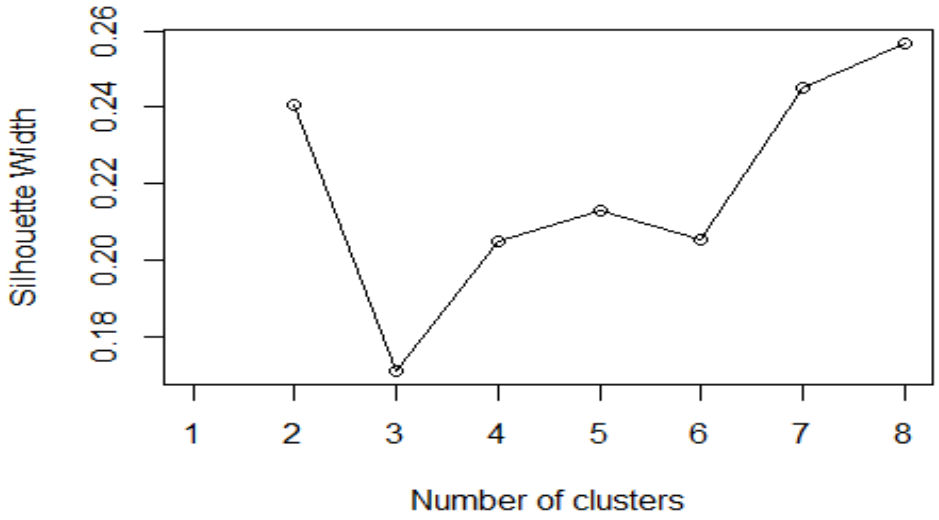
Clustering

As this is a mixed dataset [i.e] it consists of numerical and categorical variables, we are using two step clustering. Using Daisy and Gower for scaling the dataset. We have scaled the numerous variables and then binded it with the categorical variables.

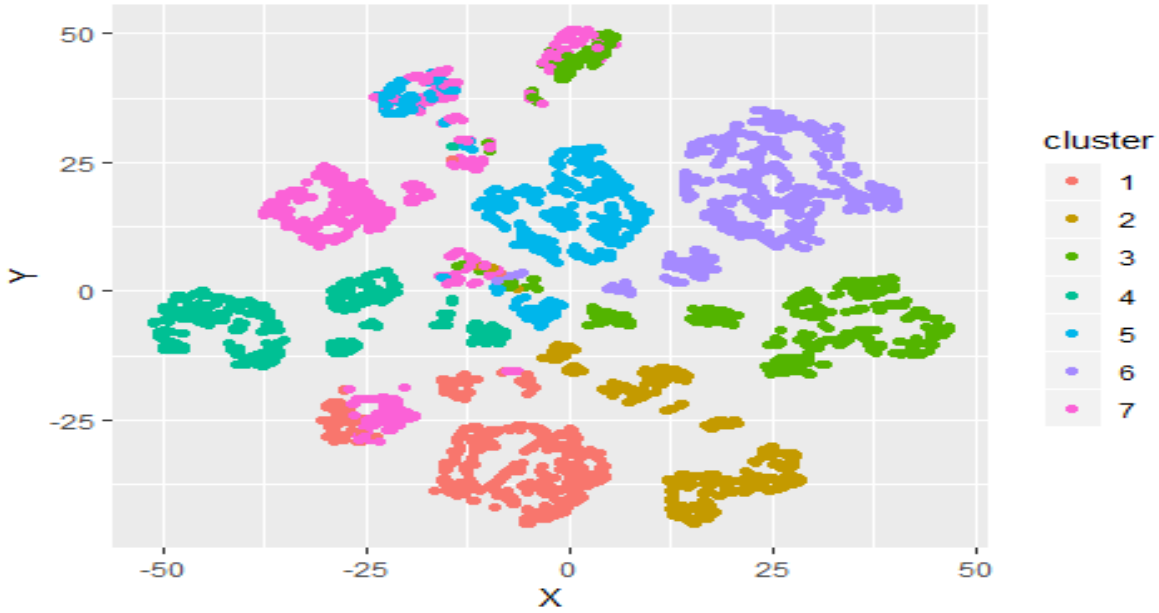
```
#distance calculation using daisy
class(gower.dist)

## [1] "dissimilarity" "dist"

summary(gower.dist)

## 12497500 dissimilarities, summarized :
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0000106 0.2078200 0.2744300 0.2828300 0.3488700 0.8189000
## Metric :  mixed ;  Types = I, I, I, I, I, I, I, N, N, N, N, N, N
## Number of objects : 5000
```



From the dendrogram, we can see that multiple clusters have been formed.



From the graph, we can see 8 clusters have the highest silhouette width. Since 7 is simpler and almost as good, lets pick k = 7.



We can see the scatter plot distribution for each cluster.

Cluster 1 and Cluster 6 has much of them with Education level as 1.

Cluster 2 and Cluster 3 has much of them with Education level as 2.

Cluster 4 and Cluster 5 has much of them with Education level as 3.

Cluster 7 has the education level distributed.

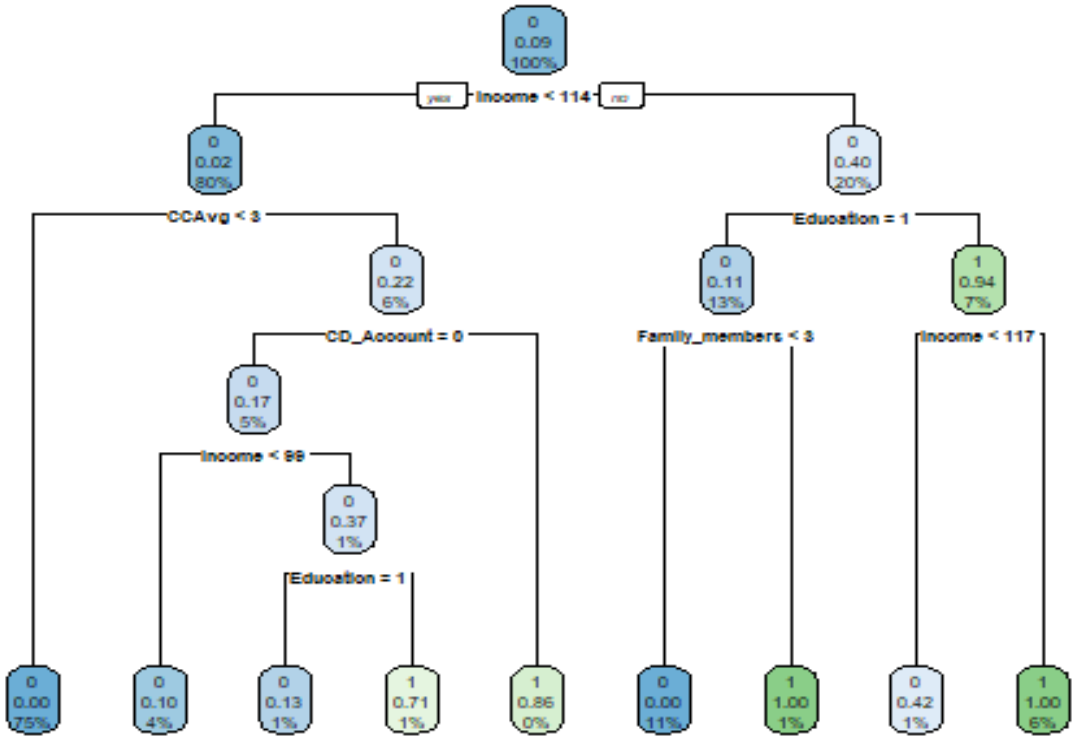
Similarly, other variables have also been distributed.

CART Model

The dataset has been split into train and test dataset in the ratio of 70:30. Only 9% of the dataset has the value as 1 [i.e] only 9% has opted for personal loan. This is an imbalanced dataset. The first model is built with minbucket as 10.

```
## n= 3500
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 3500 320 0 (0.908571429 0.091428571)
##    2) Income< 113.5 2816  49 0 (0.982599432 0.017400568)
##      4) CCAvg< 2.95 2619   5 0 (0.998090874 0.001909126) *
##      5) CCAvg>=2.95 197  44 0 (0.776649746 0.223350254)
##       10) CD_Account=0 183  32 0 (0.825136612 0.174863388)
##         20) Income< 98.5 131  13 0 (0.900763359 0.099236641) *
##         21) Income>=98.5 52  19 0 (0.634615385 0.365384615)
##           42) Education=1 31   4 0 (0.870967742 0.129032258) *
##           43) Education=2,3 21   6 1 (0.285714286 0.714285714) *
##       11) CD_Account=1 14   2 1 (0.142857143 0.857142857) *
##    3) Income>=113.5 684 271 0 (0.603801170 0.396198830)
##      6) Education=1 447  48 0 (0.892617450 0.107382550)
##       12) Family_members< 2.5 399   0 0 (1.000000000 0.000000000) *
##       13) Family_members>=2.5 48   0 1 (0.000000000 1.000000000) *
##      7) Education=2,3 237  14 1 (0.059071730 0.940928270)
##       14) Income< 116.5 24  10 0 (0.583333333 0.416666667) *
##       15) Income>=116.5 213   0 1 (0.000000000 1.000000000) *
```

The tree has been divided into two branches based on Income of lesser than 113.5. Let's plot the tree for a better idea. From the tree we can see that Income got the highest Gini Gain.



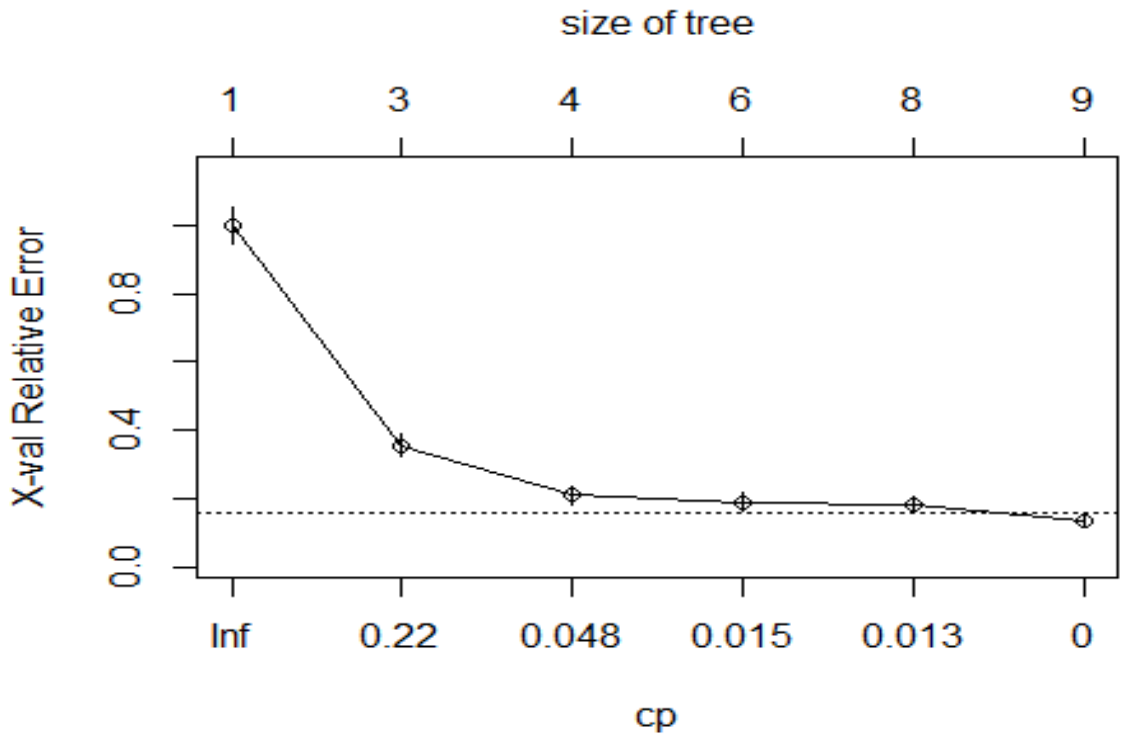
The tree has been divided based on the income <114. The tree is further divided based on CCAvg <3 and Education = 1.

```
## Classification tree:
## rpart(formula = Personal_Loan ~ ., data = CARTtrain, method = "class",
##     minbucket = 10, cp = 0)
##
## Variables actually used in tree construction:
## [1] CCAvg          CD_Account     Education      Family_members
## [5] Income
##
## Root node error: 320/3500 = 0.091429
##
## n= 3500
##
##         CP nsplit rel error  xerror     xstd
## 1 0.326563      0   1.00000 1.00000 0.053285
## 2 0.150000      2   0.34688 0.35625 0.032818
## 3 0.015625      3   0.19688 0.20938 0.025333
## 4 0.014063      5   0.16563 0.19063 0.024193
## 5 0.012500      7   0.13750 0.18438 0.023800
## 6 0.000000      8   0.12500 0.13750 0.020598
```

The variables used in the construction of the tree are CCAvg, CD Account, Education, Family Members and Income.

The tree has a misclassification rate of 0.091429 * 0.13750 * 100% = 1.26% in cross-validation (i.e. 98.74% of prediction accuracy). Now this tree can be used to produce confusion matrices and tree structure plots.

The cross-validation error is reducing, and we can see that the least value is, 0.13750. The CP value corresponding to this is, 0.000000.

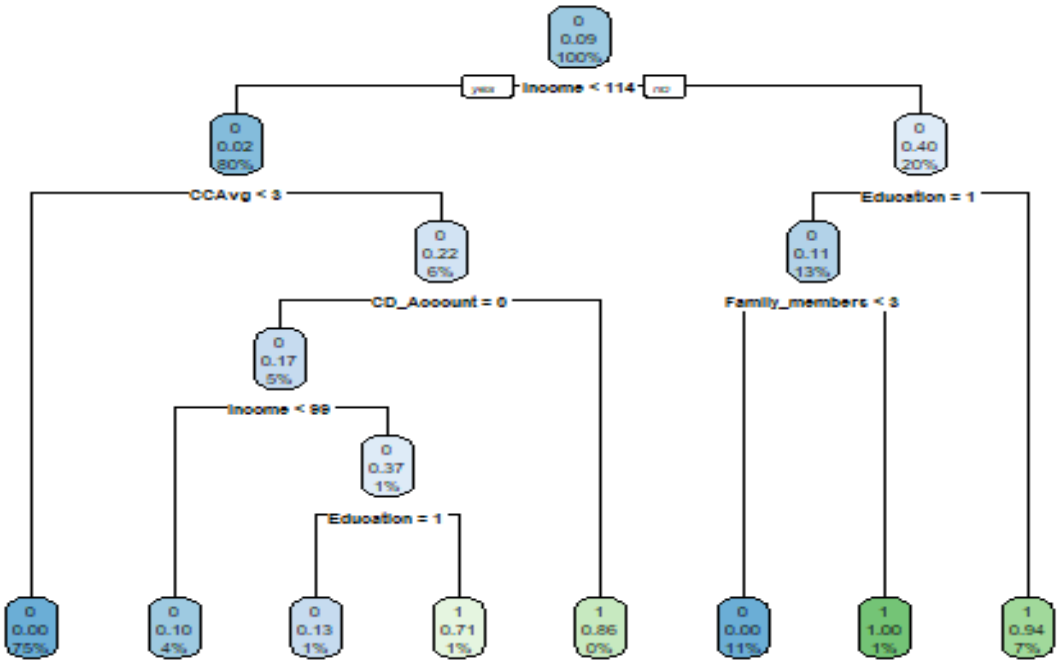


The plot shows the same. Since the CP value is 0, we will take the value above it for pruning.

```
ptree = prune(tree,cp=0.012500,"CP")

## n= 3500
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##  1) root 3500 320 0 (0.908571429 0.091428571)
##    2) Income< 113.5 2816  49 0 (0.982599432 0.017400568)
##      4) CCAvg< 2.95 2619   5 0 (0.998090874 0.001909126) *
##      5) CCAvg>=2.95 197  44 0 (0.776649746 0.223350254)
##       10) CD_Account=0 183  32 0 (0.825136612 0.174863388)
##         20) Income< 98.5 131  13 0 (0.900763359 0.099236641) *
```

```
##         21) Income>=98.5 52  19 0 (0.634615385 0.365384615)
##           42) Education=1 31   4 0 (0.870967742 0.129032258) *
##           43) Education=2,3 21   6 1 (0.285714286 0.714285714) *
##       11) CD_Account=1 14   2 1 (0.142857143 0.857142857) *
##    3) Income>=113.5 684 271 0 (0.603801170 0.396198830)
##      6) Education=1 447  48 0 (0.892617450 0.107382550)
##       12) Family_members< 2.5 399   0 0 (1.000000000 0.000000000) *
##       13) Family_members>=2.5 48   0 1 (0.000000000 1.000000000) *
##      7) Education=2,3 237  14 1 (0.059071730 0.940928270) *
```

The root is still divided based on Income again, but the other parameters have changed.

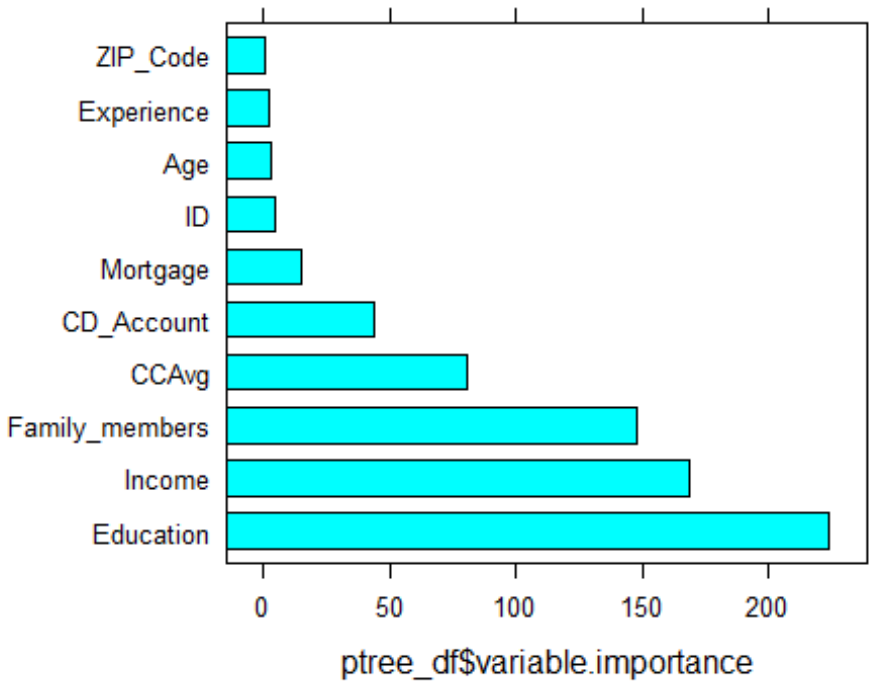


We can see the difference in the above graph.

```
## Classification tree:
## rpart(formula = Personal_Loan ~ ., data = CARTtrain, method = "class",
##     minbucket = 10, cp = 0)
##
## Variables actually used in tree construction:
## [1] CCAvg          CD_Account     Education      Family_members
## [5] Income
##
## Root node error: 320/3500 = 0.091429
##
## n= 3500
```

```
##
##         CP nsplit rel error  xerror     xstd
## 1 0.326563      0   1.00000 1.00000 0.053285
## 2 0.150000      2   0.34688 0.35625 0.032818
## 3 0.015625      3   0.19688 0.20938 0.025333
## 4 0.014063      5   0.16563 0.19063 0.024193
## 5 0.012500      7   0.13750 0.18438 0.023800
```

```
## Variable importance
##      Education         Income Family_members          CCAvg     CD_Account
##             32             24             21             12              6
##       Mortgage             ID
##              2              1
```



A bar chart is drawn to show the importance of each variable, with Education being the highest and Zip_Code being the lowest.

Random Forest Model

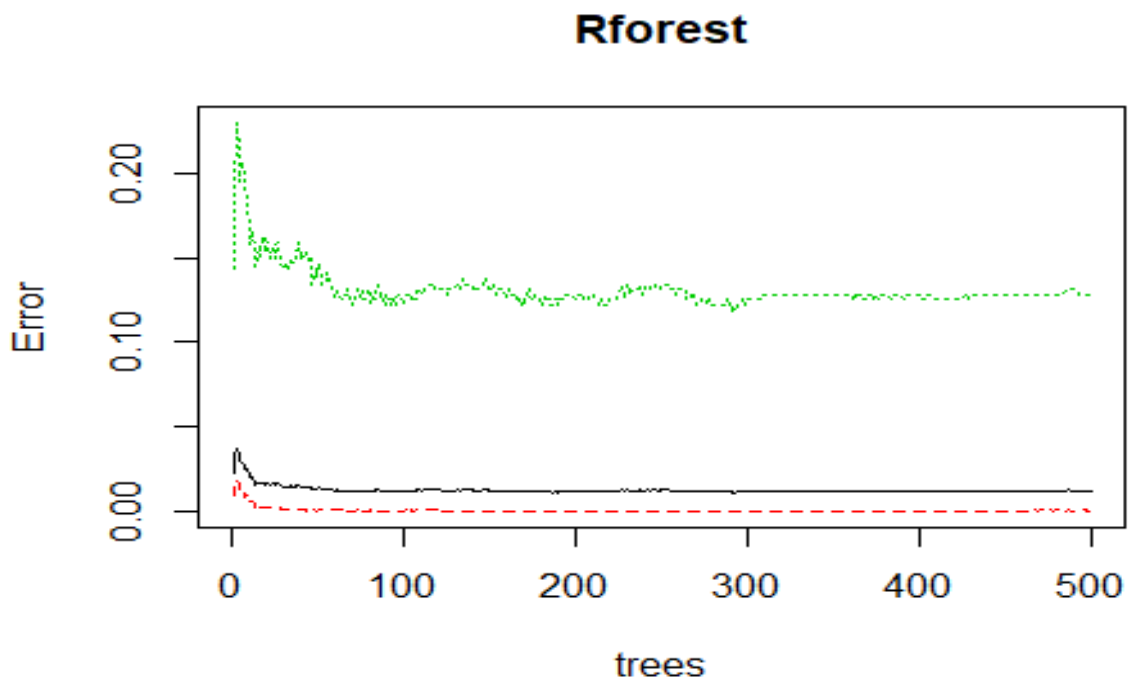
A random forest consists of multiple random decision trees. We have split the dataset into train and test in the ratio of 70:30.

We created a classification type of random forest with no of trees as 500 and no of variables considered for each tree as 3.

```
## Call:
##  randomForest(formula = Personal_Loan ~ ., data = RFtrain, importance = TR
UE)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 1.26%
## Confusion matrix:
##      0   1  class.error
## 0 3178   2 0.0006289308
## 1   42 278 0.1312500000
```

In the confusion matrix, we can see that around 3456 are predicted correctly and 44 are having wrong predictions. The OOB is 1.26%. The Out of Bag (OOB) error is a method of measuring the prediction error of random forests. If we grow 500 trees, then on average a record will be OOB for about 0.0126 * 500 = 6.3 ~ 6 trees.

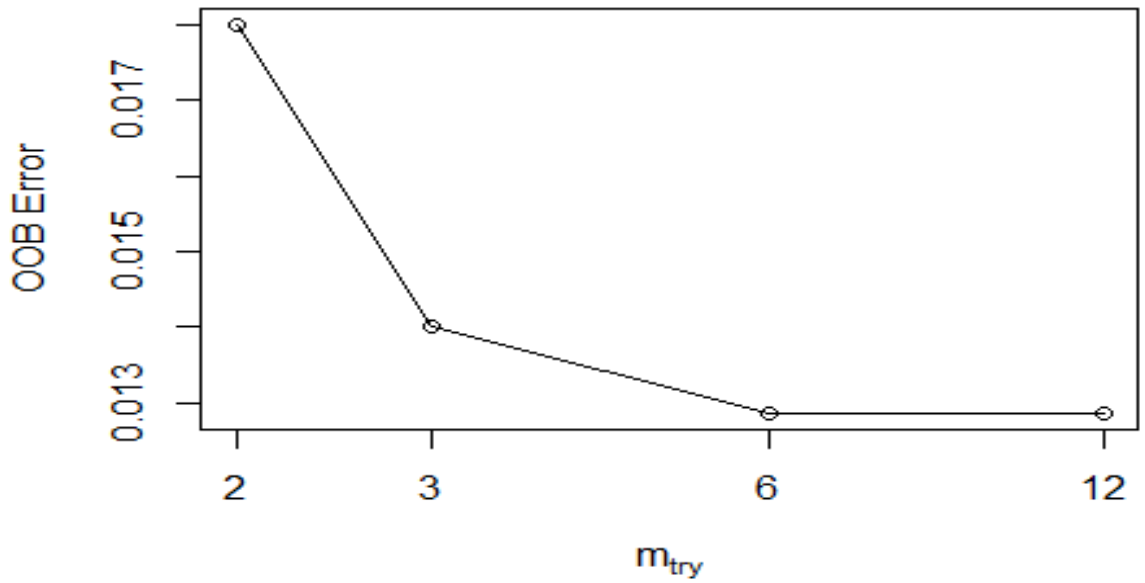
The RF is plotted to know the optimum number of tress.



We can assume after 100 trees the OOB has plateaued, so optimum trees can be 100. Let's tune up the RF model to find the best mtry.

```
## mtry = 3  OOB error = 1.4%
## Searching left ...
## mtry = 2     OOB error = 1.8%
```

```
## -0.2857143 1e-04
## Searching right ...
## mtry = 6     OOB error = 1.29%
## 0.08163265 1e-04
## mtry = 12    OOB error = 1.29%
## 0 1e-04
```



We can see that the OOB error is least for 6 mtry with an OOB rate of 1.29%.

The refined RF model is built.

```
## Call:
##  randomForest(formula = Personal_Loan ~ ., data = RFtrain, ntree = 100,
mtry = 6, nodesize = 10, importance = TRUE)
##                Type of random forest: classification
##                      Number of trees: 100
## No. of variables tried at each split: 6
##
##         OOB estimate of  error rate: 1.23%
## Confusion matrix:
##      0   1 class.error
## 0 3171   9 0.002830189
## 1   34 286 0.106250000
```

The error rate has improved from 1.26% to 1.23%. There is an improvement of only 0.03%.

```
temp %>% arrange(desc(MeanDecreaseGini))

##              0           1 MeanDecreaseAccuracy MeanDecreaseGini
## 1  63.4672881 45.29813681          69.36952362      170.9066001
## 2  75.2200286 34.49068019          76.60492989      152.0244085
## 3  54.5120293 24.09625253          52.84932661       77.3841432
## 4  13.0634984 13.64855814          14.97476011       68.5030796
## 5   4.9538367  5.15686033           7.39992417       27.1428184
## 6   0.2211146 -2.45502968          -1.11724313        8.5603964
## 7  -0.7286519 -1.31308151          -1.53742340        8.1296733
## 8   5.3444969 -0.01964271           5.20370747        8.1223132
## 9   3.8901149 -2.94285743           3.36852516        7.2555117
## 10  4.7536044 -0.18767278           4.35019822        6.9027164
## 11  0.3979092  1.41187783           1.21194708        1.5276220
## 12  2.7174631 -0.39971007           3.04816795        1.1975409
## 13 -0.3502029  0.71906961          -0.01319021        0.5963092
```

If we check the MeanDecreaseGini, Income has the highest value of 170.91. Education has the value of 152.02. CCAvg has a value of 77.38. Family members has a value of 68.50.

Confusion Matrix

A confusion matrix is a technique for summarizing the performance of a classification algorithm.

Matrix for CART Model for train.

	0	1
0	3158	22
1	22	298

We can see that around 3158 out of 3180 are predicted as 0 while 298 out of 320 are predicted as 1. About 3456 are predicted correctly and 44 are having wrong predictions. The False Negative or Type II error is 22. The False Positive or Type I error is 22.

Matrix for CART Model for test.

	0	1
0	1330	10
1	20	140

We can see that around 1330 out of 1350 are predicted as 0 while 140 out of 150 are predicted as 1. About 1470 are predicted correctly and 30 are having wrong predictions. The False Negative or Type II error is 10. The False Positive or Type I error is 20.

Matrix for Random Forest for train.

	0	1
0	3171	9
1	34	286

We can see that around 3171 out of 3205 are predicted as 0 while 286 out of 295 are predicted as 1. About 3457 are predicted correctly and 43 are having wrong predictions. The False Negative or Type II error is 9. The False Positive or Type I error is 34.

Matrix for Random Forest for test.

	0	1
0	1336	4
1	22	138

We can see that around 1336 out of 1358 are predicted as 0 while 138 out of 142 are predicted as 1. About 1474 are predicted correctly and 26 are having wrong predictions. The False Negative or Type II error is 4. The False Positive or Type I error is 22.

Performance Measure Parameters

Model Name	Samp-le	KS	AUC	GINI	Conc-orda-nce	Accu-racy	Sensi-tivity	Speci-ficity	Class Error
CART	Train	0.9319	0.9892	0.8890	0.9804	0.9874	0.9313	0.9931	0.0126
CART	Test	0.8925	0.9728	0.8811	0.9501	0.9800	0.8750	0.9852	0.0200
Random Forest	Train	0.9333	0.9942	0.8966	0.9928	0.9877	0.8938	0.9893	0.0123
Random Forest	Test	0.9344	0.9944	0.8854	0.9938	0.9827	0.8625	0.9838	0.0173

The KS has returned 93% for both the CART and RF model for train data. There is a variation for the test data, with CART giving 89% while RF giving 93%.

The AUC (Area Under the Curve) shows that both the models are returning 99% for the trained dataset. But for test dataset, CART gives 97% whereas RF is giving 99%. Showing RF is a better model.

The GINI index gives 89% for the train data for both the models. In case of test, CART gives 88% for both.

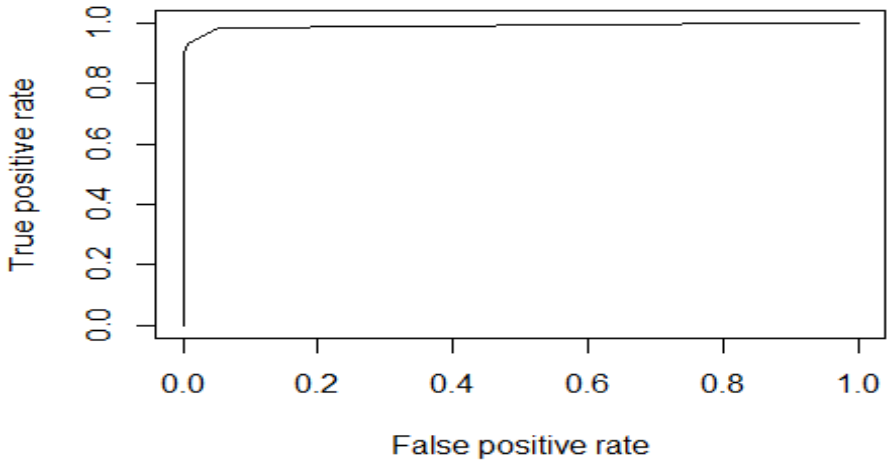
A confusion matrix is a technique for summarizing the performance of a classification algorithm. Here we are comparing CART verse the Random Forest algorithm.

The classification accuracy is the ratio of correct predictions to total predictions made. As we can see, both the models have given an accuracy of more than 98%.

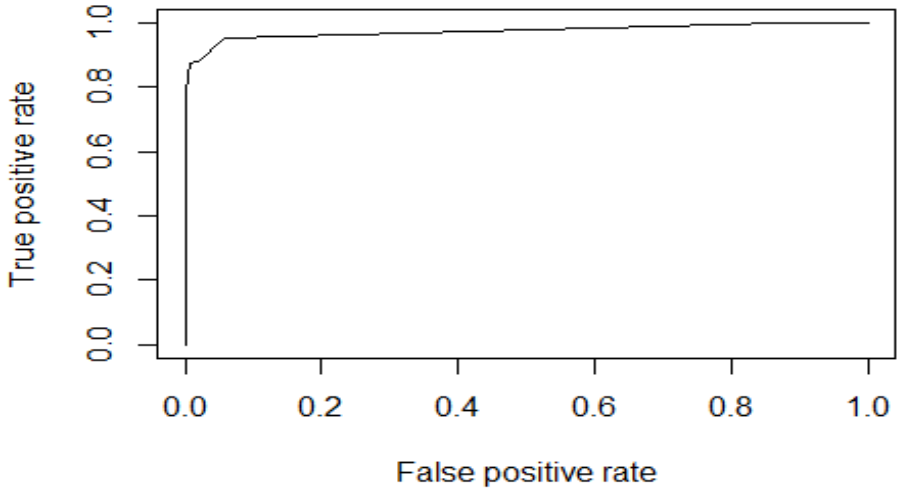
Sensitivity is the percentage of actual 1's correctly predicted by the model. Here we can see that CART model has 93.13% and 87.50% for training and test dataset respectively. While RF model has 89.38% and 86.25% for training and test dataset respectively.

Specificity is the proportion of actual 0's that were correctly predicted. For all the models it is almost 99%.

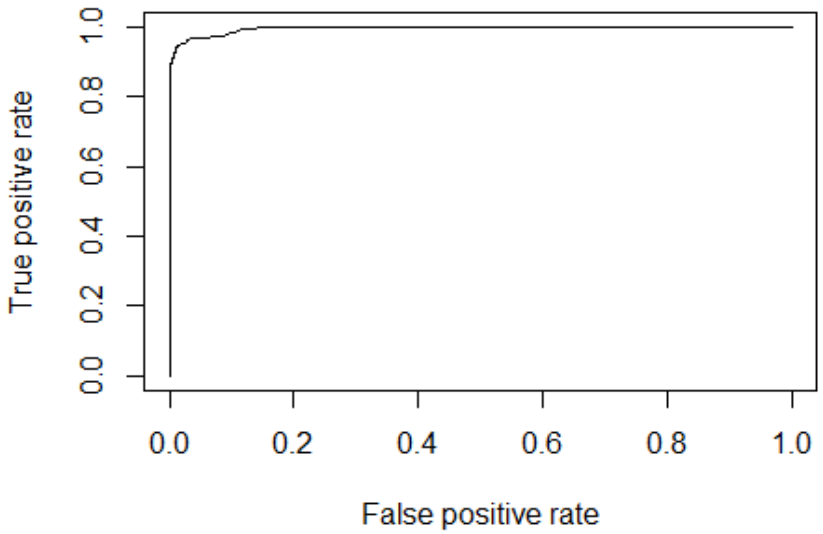
The class error rate is the inverting value of the classification accuracy, the error rate is around 1.2% for both the models for the train dataset. While the test data is showing around 2% for CART and 1.7% for Random Forest.



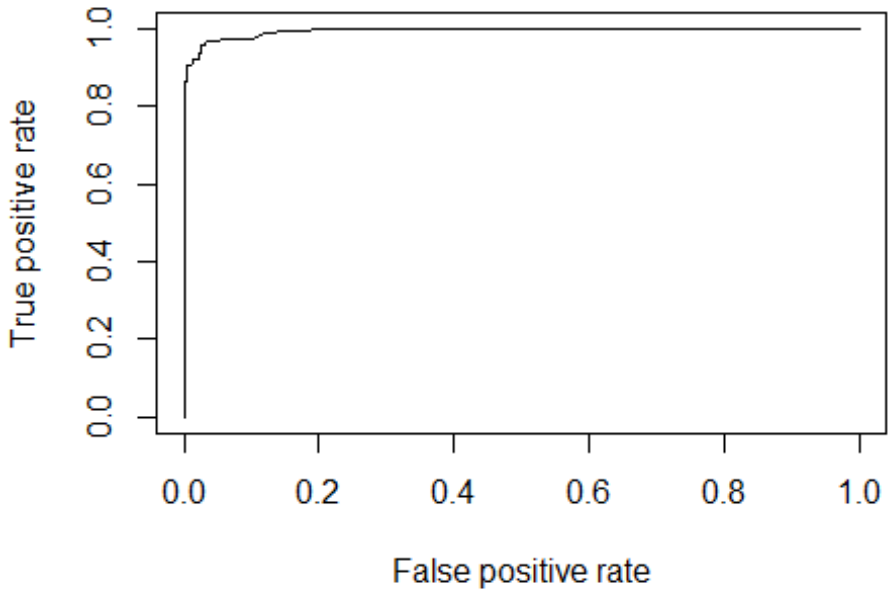
CART performance map for train dataset. The AUC is almost equal to 1.



CART performance map for test dataset. The AUC is almost equal to 1.



Random Forest performance map for train dataset. The AUC is almost equal to 1.



Random Forest performance map for the test dataset. The AUC is almost equal to 1.

Conclusion

The aim of Thera Bank is to convert its liability customers to loan customers. They want to set up a new marketing campaign. Therefore, they need information about the connection between the variables and the given data. From the above analysis we may consider, the main variables to be Income, Education, CC Average and Family Members.

Two classification algorithms, CART and Random Forest were used for this study. Both the models provided an accuracy of 98%. All the other performance parameters also show that there is only a slight difference between both the models.

From the above table, we can conclude that overall, Random Forest has given a slightly better performance than CART Model. Therefore, we can choose Random Model as our final model.