THERA BANK PROJECT

Karthik Paranthaman

UNIVERSITY OF TEXAS

1. PROJECT OVERVIEW

The main aim of this project is to explore the data set provided by the client "Thera Bank" to find business insights to improve the percentage of loan offerings. First Chapter discusses the EDA analysis such as Univariate analysis and Bi-Variate analysis. 2nd Chapter illustrates the Clustering techniques to analyze the data. Chapter -3 will be discussing the CART model and the performance analysis. Chapter-4 talks about the Random Forest Techniques and the model performance evaluation is discussed. Chapter -5 provides a conclusive findings.

2. ASSUMPTIONS

In the experience attributes some of the observations were found to be negative. All negative rows were converted into ZERO. It was assumed that all information provided by the customers are accurate.

3. EXPLORATORY DATA ANALYSIS

In this Chapter we will be discussing about the data exploration approaches to find business insights. The exploratory process consists of the following stages.

1. Environment Setup and Data Import
2. Missing Value Identification
3. Missing Value Treatment.
4. Variable Identification
5. Univariate Analysis
6. Bi-Variate Analysis

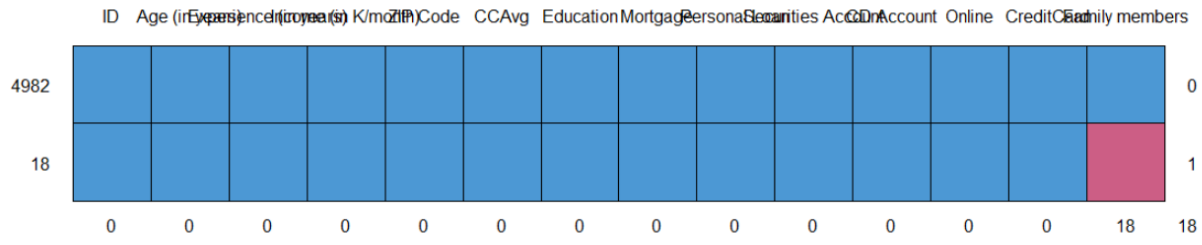
3.1 Environment Setup and Data Import

3.1.1 WORKING DIRECTORY

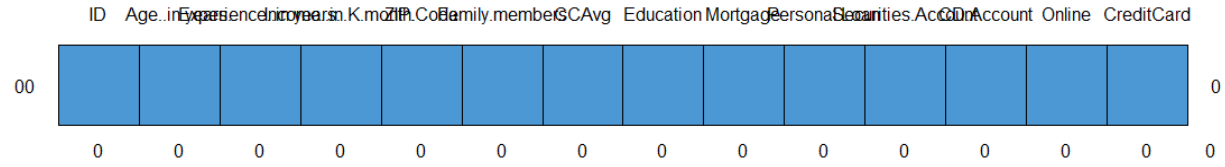
The working directory has been set. The dataset is called using the name "Customer_Details" in R-Studio (Refer to APPENDIX -1 – for R Code).

3.2 Missing Value Identification

The dataset contains 18 missing values in family members attribute. The graphical representation below shows the data distribution among the 12 attributes in the dataset.



"mice" library package is used to treat the missing values. After missing value treatment above graphical representation was run against the entire dataset. The below figure indicates that the missing value issue had been rectified.



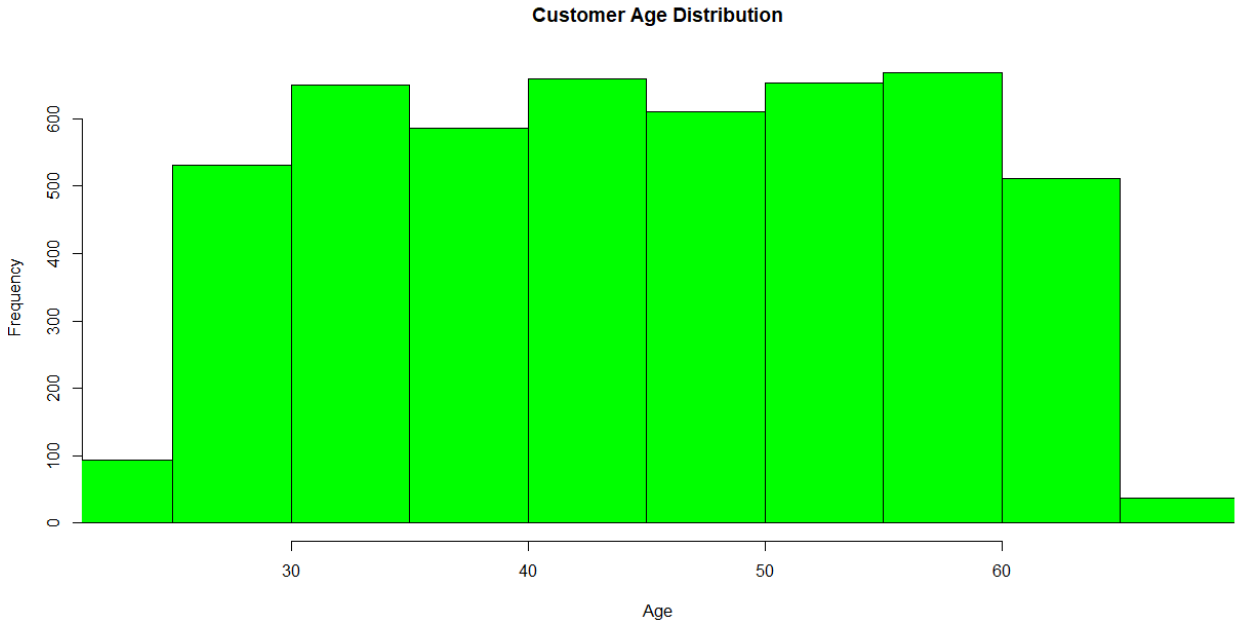
5 number summary is given below to obtain an idea of the data distribution among each attribute in dataset provided by Thera Bank.

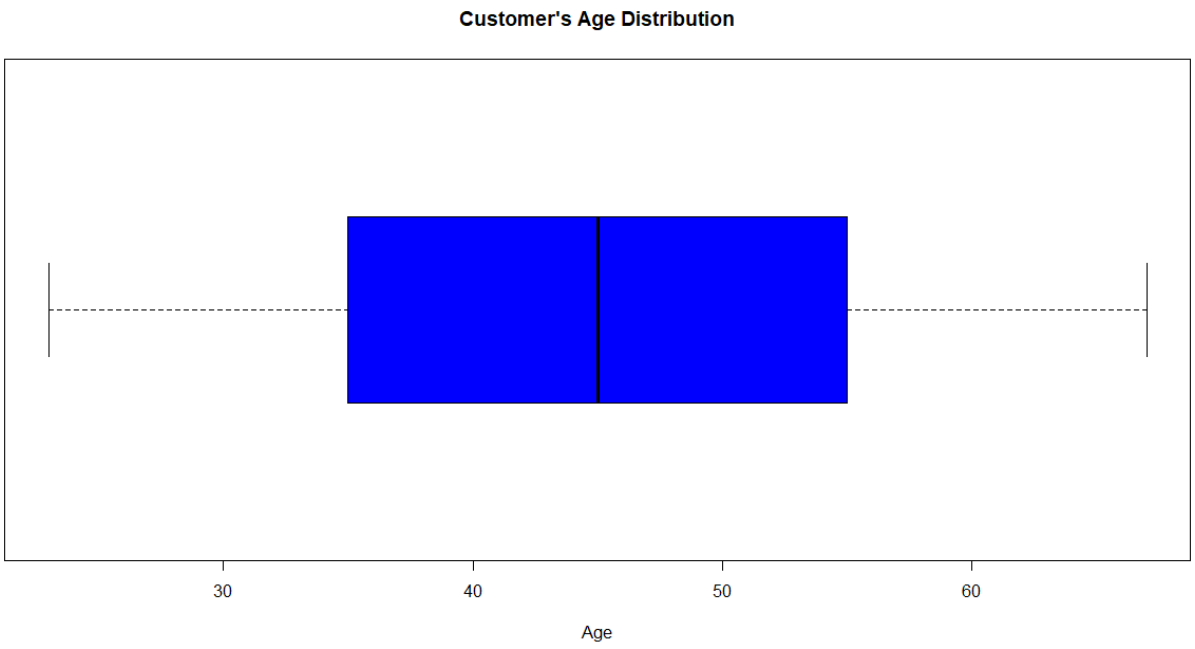
```
> summary(Customer_Details_Treated)
       ID        Age..in.years.  Experience..in.years. Income..in.K.month.     ZIP.Code
 1      :   1   Min.   :23.00   Min.   :-3.0          Min.   :  8.00       94720  : 169
 2      :   1   1st Qu.:35.00   1st Qu.:10.0          1st Qu.: 39.00       94305  : 127
 3      :   1   Median :45.00   Median :20.0          Median : 64.00       95616  : 116
 4      :   1   Mean   :45.34   Mean   :20.1          Mean   : 73.77       90095  :  71
 5      :   1   3rd Qu.:55.00   3rd Qu.:30.0          3rd Qu.: 98.00       93106  :  57
 6      :   1   Max.   :67.00   Max.   :43.0          Max.   :224.00       92037  :  54
 (Other):4994                                                              (Other):4406
 Family.members      CCAvg        Education    Mortgage      Personal.Loan
 Min.   :1.000   Min.   : 0.000   1:2096    Min.   :  0.0   0:4520
 1st Qu.:1.000   1st Qu.: 0.700   2:1403    1st Qu.:  0.0   1: 480
 Median :2.000   Median : 1.500   3:1501    Median :  0.0
 Mean   :2.399   Mean   : 1.938             Mean   : 56.5
 3rd Qu.:3.000   3rd Qu.: 2.500             3rd Qu.:101.0
 Max.   :4.000   Max.   :10.000             Max.   :635.0

 Securities.Account CD.Account Online   CreditCard
 0:4478             0:4698     0:2016   0:3530
 1: 522             1: 302     1:2984   1:1470
```

3.3 Univariate Analysis

AGE:

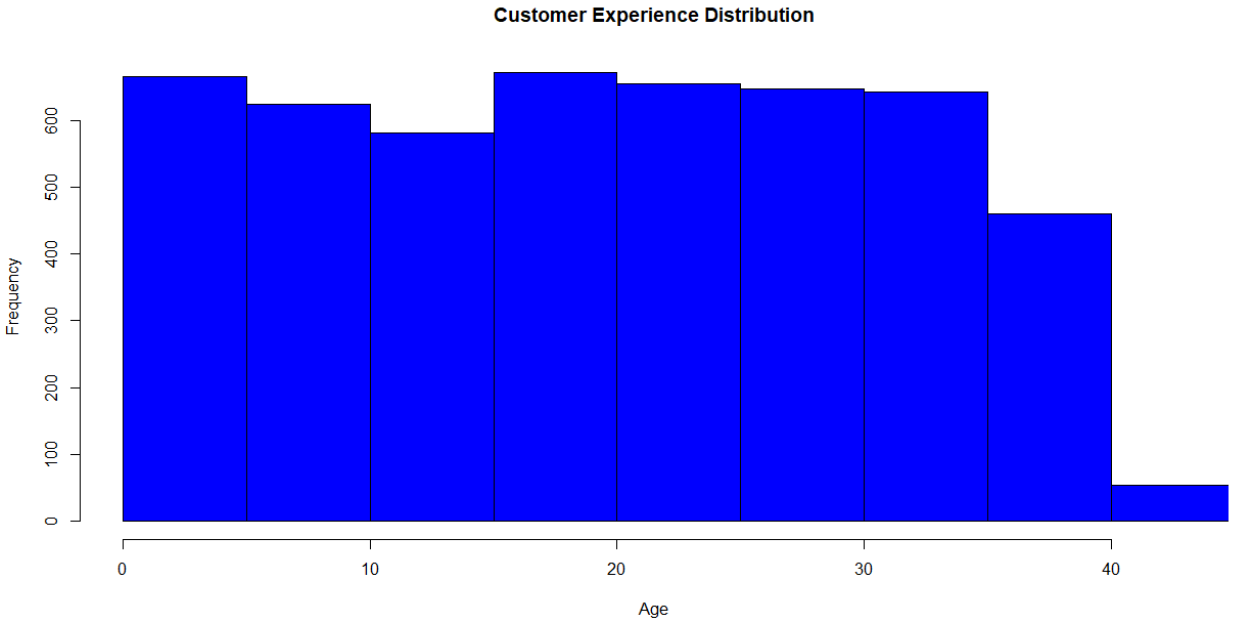
Range of age distribution is 23 to 67.

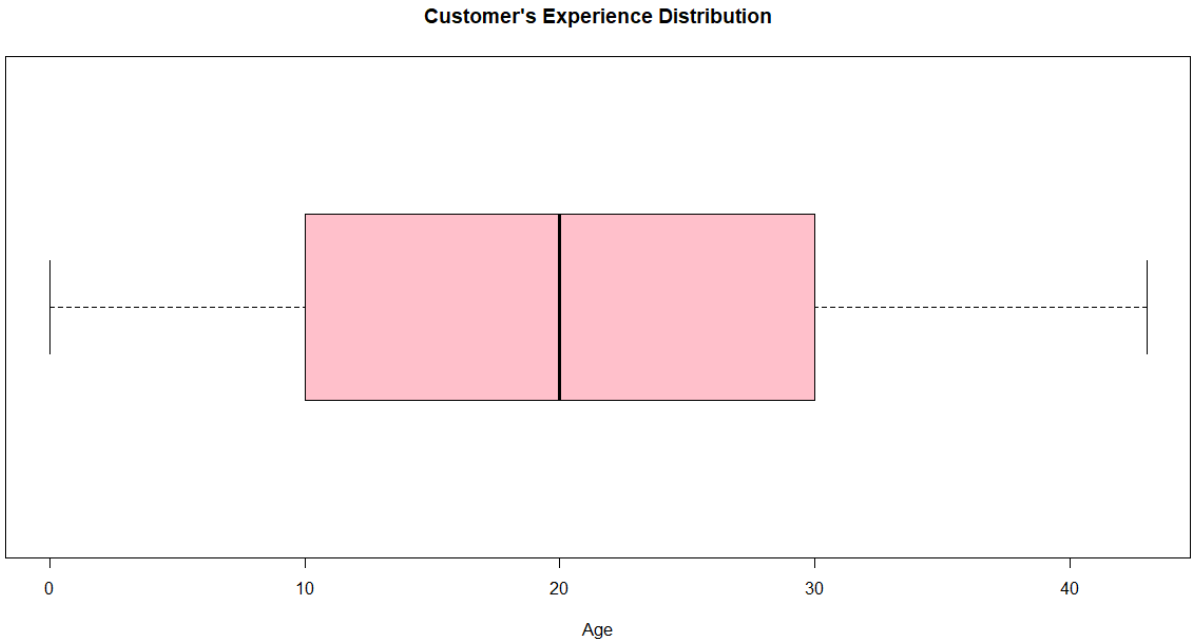




EXPERIENCE:

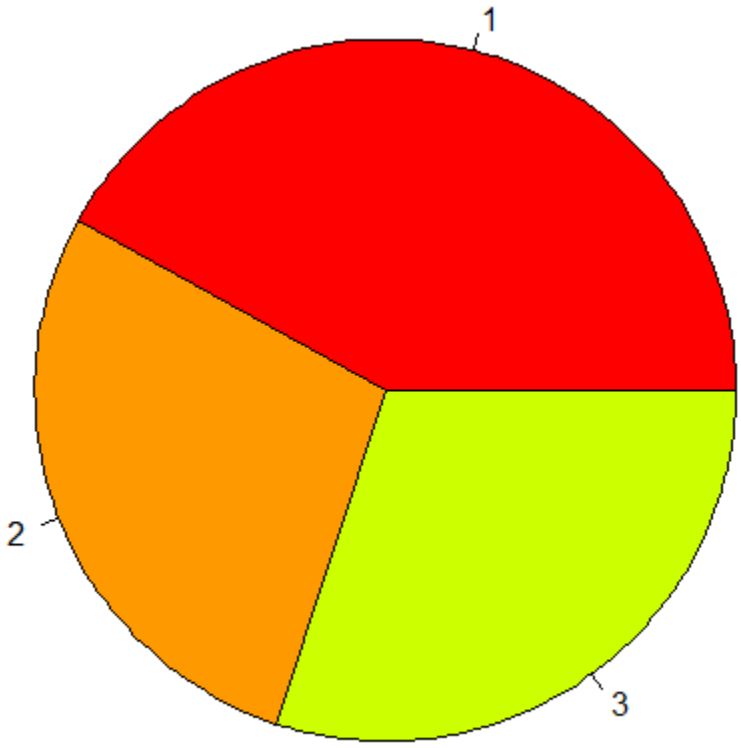
Range of Experience distribution is 0 to 43. All negative entries were converted into zero.





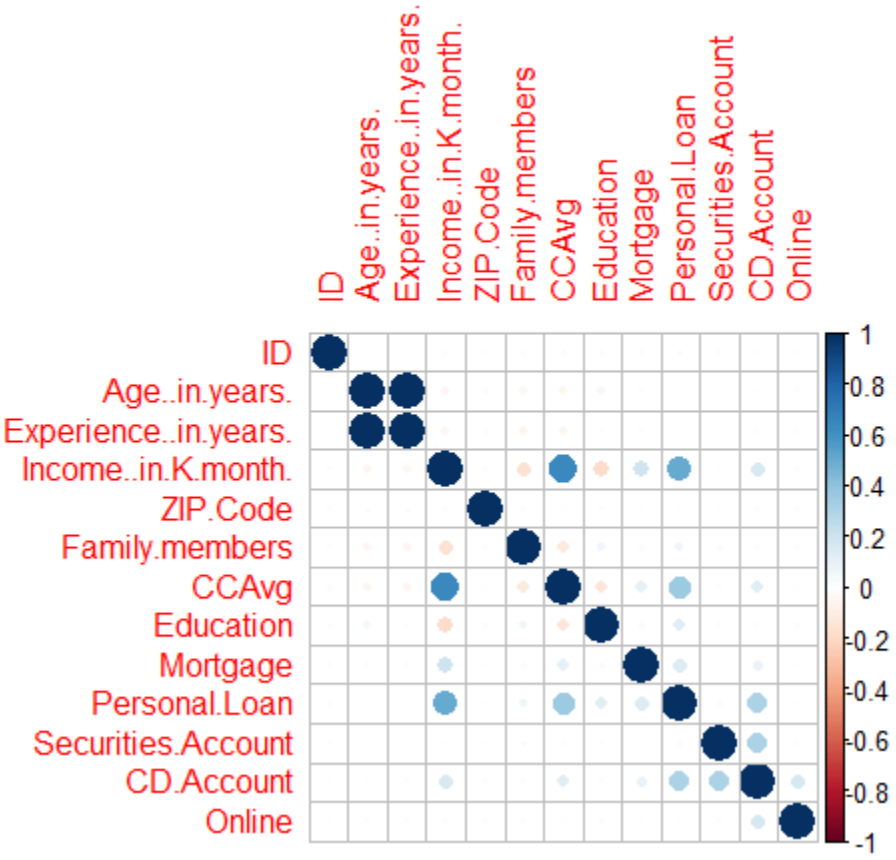
EDUCATION:

As shown in the pie chart below, the majority of the customers have an educational background of undergraduate around 2096. Second biggest customers hold an advanced professional qualification. This tells us that our customers are highly educated.



3.3 Multi Variate Analysis

Below correlation plot clearly indicates the co linearity between each group:

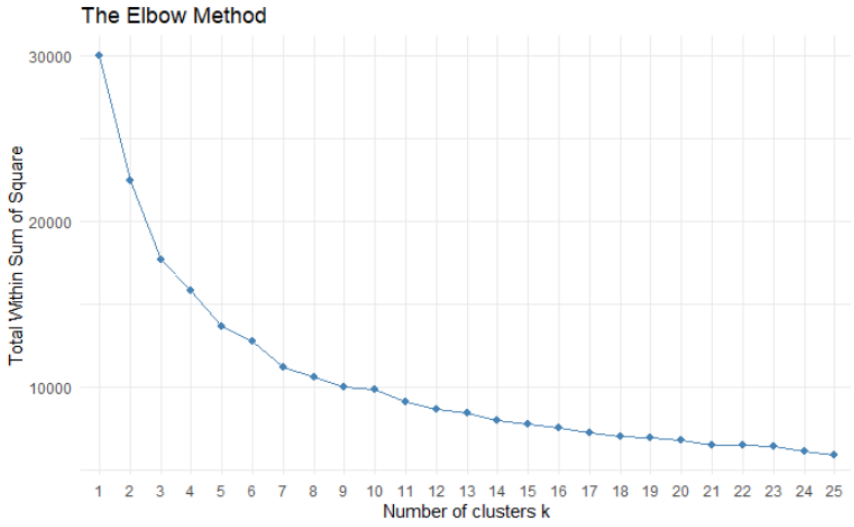


CLUSTERING

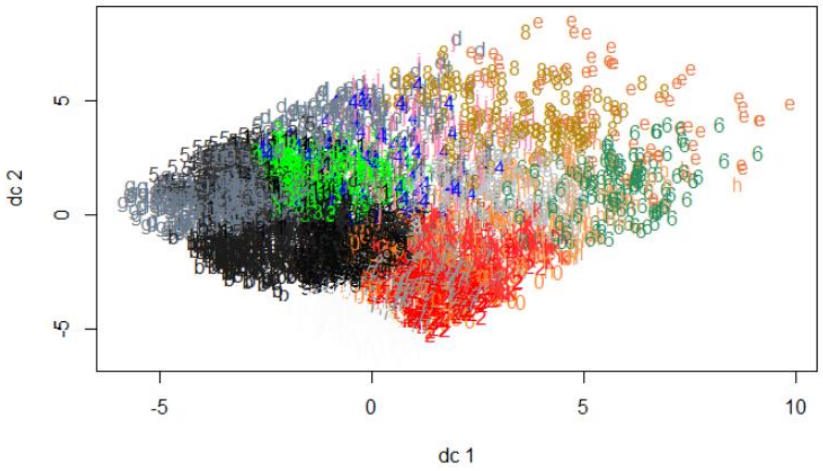
After the univariate and bivariate analysis it can be said that K- means clustering is most suitable and computationally less expensive. Also, the income attribute was observed to have very large difference between each values and as such scaling was done to improve analytics power of the dataset.

Set seed value to be 1000. This value will be maintained thorough out the clustering analytics in order to maintain the uniformity of the distribution.

An "Elbow-method " is used to find the optimum value of K. As clearly indicated in the figure below it can be said that for the first time an additional cluster does not reduce the Within Sum of Square when K= 21. Therefore, optimum number of clusters is 21.



Before we proceed with the discussion of a detailed partition algorithm, it's advisable to analyze how well clusters are separated with a minimal overlap. Below figure gives a graphical representation of the clustered data with a K value of 21.



It can be said that points not separated properly. There numerous overlapping can be obtained. Hence, let's run the "Nbclust" command to commute the suitable number of clusters. Inrder to avoid overlapping K value require a fine tuning.

CART MODELING

CART modelling is done by portioning the main data set into 70% and 30% using the following code and titled them as CART_Train and CART_Test respectively.

The R packages used in this sections are listed below:

install.packages("rpart")

library(rpart)

install.packages("rpart.plot")

library(rpart.plot)

library(caTools)

Data Partioning:

70% and 30% data split R code for CART model generation is shown below.

```
# DATA SPLIT - CART

library(caTools)
set.seed(1000) #Input any random number
spl = sample.split(Customer_Details_Treated$Personal.Loan, SplitRatio = 0.7)
CART_Train = subset(Customer_Details_Treated, spl == T)
dim(CART_Train)
CART_Test = subset(Customer_Details_Treated, spl == F)
dim(CART_Test)
```

Train data set consists of 336 of customer have accepted the Personal loan offer in the last campaign while only 144 positive customer from the test dataset.

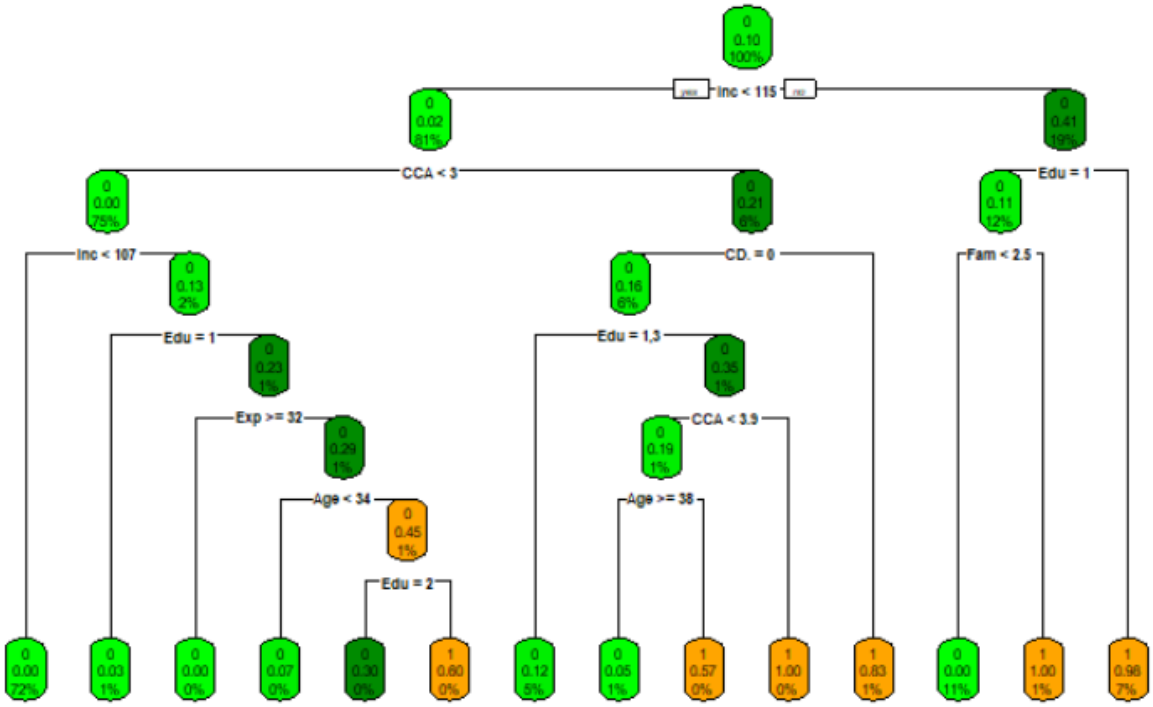
The cart model is developed and the out put is shown below.

```
cartParameters = rpart.control(minisplit=30,minibucket=10,cp=0,xval=10)
cartModel_Train=rpart(formula=Personal.Loan~.,data=CART_Train[,c(-1,-5)],method="class",
                control=cartParameters)
cartModel_Train
```

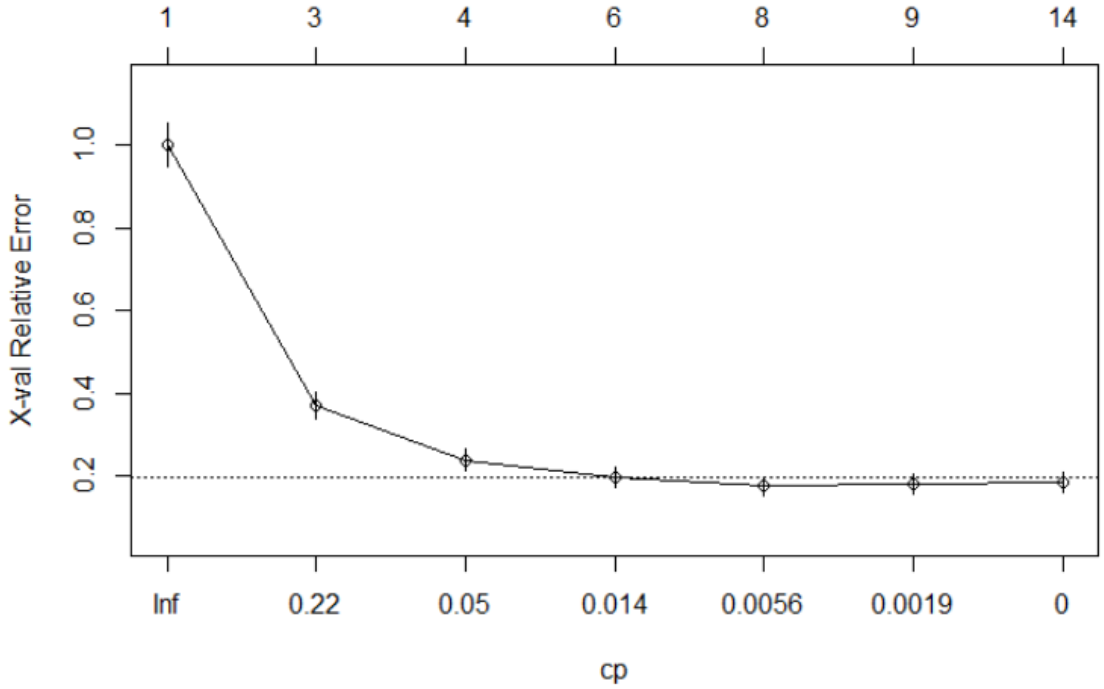
```
n= 3500

node), split, n, loss, yval, (yprob)
      * denotes terminal node

  1) root 3500 336 0 (0.904000000 0.096000000)
    2) Income..in.K.month.< 114.5 2827  57 0 (0.979837283 0.020162717)
      4) CCAvg< 2.95 2613  11 0 (0.995790279 0.004209721)
        8) Income..in.K.month.< 106.5 2530   0 0 (1.000000000 0.000000000) *
        9) Income..in.K.month.>=106.5 83  11 0 (0.867469880 0.132530120)
         18) Education=1 39   1 0 (0.974358974 0.025641026) *
         19) Education=2,3 44  10 0 (0.772727273 0.227272727)
           38) Experience..in.years.>=31.5 10   0 0 (1.000000000 0.000000000) *
           39) Experience..in.years.< 31.5 34  10 0 (0.705882353 0.294117647)
             78) Age..in.years.< 33.5 14   1 0 (0.928571429 0.071428571) *
             79) Age..in.years.>=33.5 20   9 0 (0.550000000 0.450000000)
```

Based on the CART model output it can be said that there 336 GOOD predictions out of 3500 datasets have been made. This shows an unbalance nature of the dataset which could lead to a bias decision. Probability of good prediction is about 9.6% while 90.4% probability of bad prediction.

Using “rpart.plot” function a decision tree is plotted. Refer to the diagram below.



The relative error of the CART model needs to be evaluated and as such Complexity Parameter vs Relative Error plot is generated to find the perfect CP value.



The root node error for good prediction is about 9.6% .It can be observed that relative error decreases as the number of split value increases.

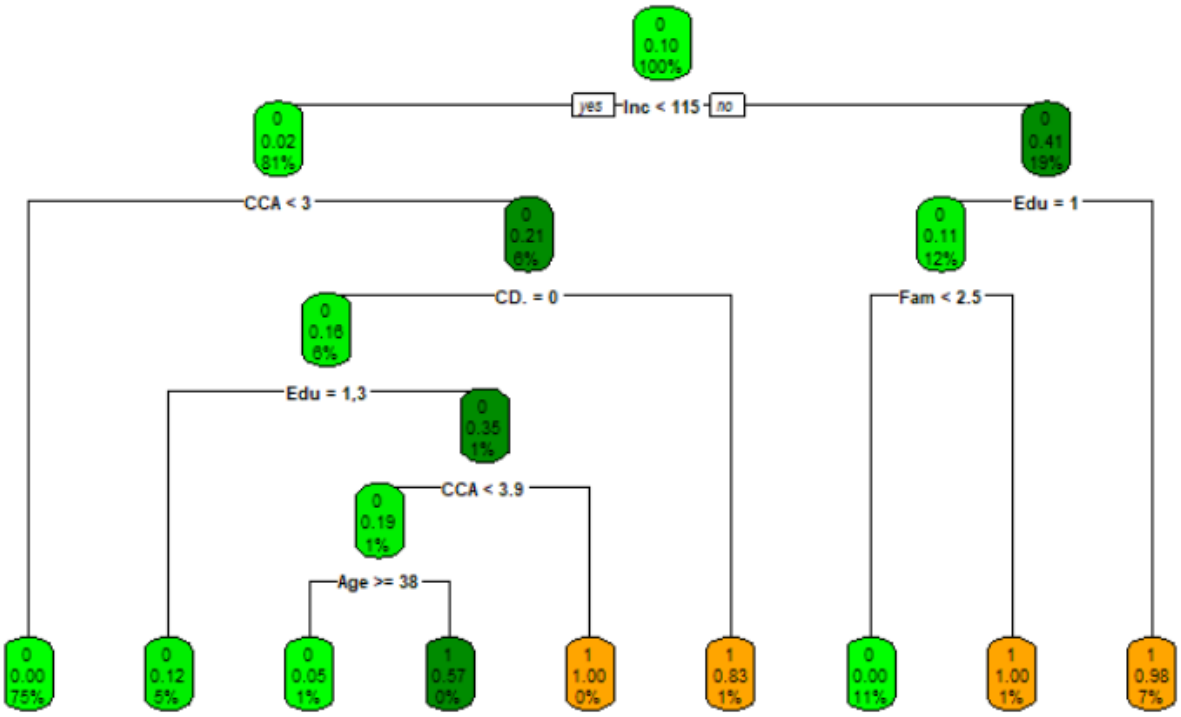
```
Classification tree:
rpart(formula = Personal.Loan ~ ., data = CART_Train[, c(-1,
    -5)], method = "class", control = cartParameters)

Variables actually used in tree construction:
[1] Age..in.years.        CCAvg                 CD.Account
Education
[5] Experience..in.years. Family.members        Income..in.K.month.

Root node error: 336/3500 = 0.096

n= 3500

         CP nsplit rel error  xerror     xstd
1 0.3377976      0   1.00000 1.00000 0.051870
2 0.1398810      2   0.32440 0.37202 0.032675
3 0.0178571      3   0.18452 0.23810 0.026314
4 0.0104167      5   0.14881 0.19643 0.023950
5 0.0029762      7   0.12798 0.17560 0.022667
6 0.0011905      8   0.12500 0.18155 0.023041
7 0.0000000     13   0.11905 0.18452 0.023226
```

Let's use the R inbuilt function to find the best CP value for the above tree and start pruning the tree. That is about 0.00297619.

```
bestcp=cartModel_Train$cptable[which.min(cartModel_Train$cptable[,"xerror"]),"CP"]
bestcp
```

```
[1] 0.00297619
```

The pruned tree with new CP is displayed below.



Now the pruned decision tree will be used, to do the prediction and the probability of the success rate on the personal loan attribute.

Securities.Account <fctr>	CD.Account <fctr>	Online <fctr>	CreditCard <fctr>	predict.class <fctr>	prob_1 <dbl>
0	0	0	0	1	0.978902954
0	0	0	0	0	0.004209721
0	0	0	0	1	0.978902954
1	0	0	1	0	0.004209721
0	0	1	0	0	0.004209721
0	0	1	0	0	0.004209721
0	0	1	0	0	0.004209721
1	0	0	0	0	0.004209721
0	0	1	0	0	0.004209721
0	0	0	0	0	0.004209721

As shown in the above table, it can be said that all personal loan 1 prediction have higer probability in the range of approximately 97%.

PERFORMANCE MEASURE ON THE TRAINING DATASET – CART MODEL

The following performance measures were measured to identify the accuracy of our model.

1. Rank Ordering Method
2. Kolomogorov-Smirnov (K-S Chart)
3. Area Under Curve (AUC)
4. GINI Coefficient
5. Concordance – Discordance Ratio

As explained above, The main purpose to find out how well this model would perform: The first step of this process is to start with the confusion matrix on the Actual Vs Predicted Value on Personal Loan from the pruned decision trees.

CONFUSION MATRIX:

The output of the confusion matrix is given below:

```
      0    1
  0 3153   11
  1   31  305
[1] 0.012
```

The mis-classification through the off diagonal entries. The error rate of the confusion matrix to be 1.2%.

THE RANK ORDERING TABLE:

Now we chop all the data into various buckets. But based on the decision tree output there are limited number of possibilities of chopping can be done with the end node. First let's find all the percentile and analyze whether data could be chopped up into different basket.

```
 [1] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
         0%         10%         20%         30%         40%         50%         60%
0.000000000 0.000000000 0.004209721 0.004209721 0.004209721 0.004209721 0.004209721
        70%         80%         90%        100%
0.004209721 0.004209721 0.117283951 1.000000000
```

So, the output above illustrates the number chops that can be done in the CART_Train dataset. The bucket combination is listed below:

Bucket 1: 0% to 80%

Bucket 2: 80% > 90%

Bucket 3:90% > 100%

Let's compute the deciles using R "cut" function. The output can be viewed at the train dataset.

CD.Account <fctr>	Online <fctr>	CreditCard <fctr>	predict.class <fctr>	prob_1 <dbl>	deciles <fctr>
0	0	0	1	0.978902954	[0.117,1]
0	0	0	0	0.004209721	[0.00421,0.117)
0	0	0	1	0.978902954	[0.117,1]
0	0	1	0	0.004209721	[0.00421,0.117)
0	1	0	0	0.004209721	[0.00421,0.117)
0	1	0	0	0.004209721	[0.00421,0.117)
0	1	0	0	0.004209721	[0.00421,0.117)
0	0	0	0	0.004209721	[0.00421,0.117)
0	1	0	0	0.004209721	[0.00421,0.117)
0	0	0	0	0.004209721	[0.00421,0.117)

As mentioned before, the all entries with a decile value of 0 to 0.00421 will be put in one bucket. Values which are greater than 0.00421 and up to 0.117 will be grouped separately. Last but not the least values above 0.117 and up to 1 will be grouped into a separate group.

R packages used in to analyze decile using table function. These packages provides us options to various changes that we could do in databases.

library(data.table)

library(scales)

The figure below shows the count based on each deciles.

Deciles	Cnt	Cnt_ Loan1	Cnt_ Loan0	Res%	Cum_ Res	Cum_ nonRes	Cum_rel_ Resp	Cum_rel_ nonResp	KS
[0.117,1]	478	324	154	67.87	324	154	96.4	4.87	91.56
[0.0021,0.117)	2633	12	2621	0.46	336	2775	100.00	87.71	12.29
[0,0.00421)	389	0	389	0.00	336	3164	100.00	100.00	0.00

The above summary shows the 1s and 0s of the personal loan column based on different deciles. Majority of the observations have the decile range of 0.00421 to 0.117. Additionally, the table illustrates on the response rate, cumulative response rate of the customers. The first decile contains the maximum number of response rate. The third decile shows a very poor response rate.

THE KS & AREA UNDER CURVE:

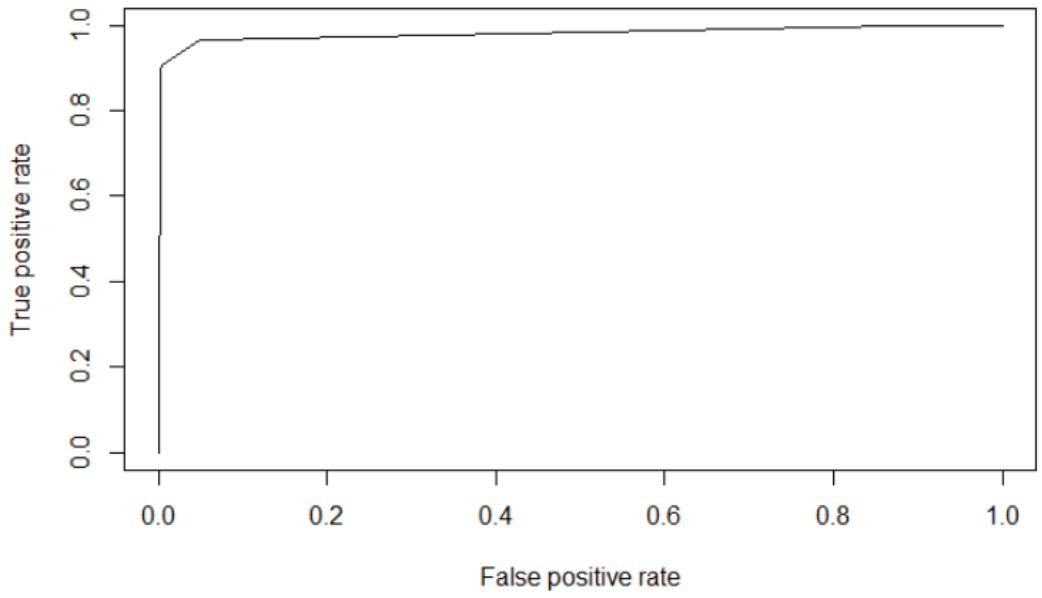
R packages used in this sections are listed below.

library(ROCR)

library(ineq)

library(InformationValue) # To measure concordance and Discordance

The below graph illustrates the False Positive Rate Vs True Positive Rate.



THE SUMMARY OF MODEL PERFORMCE MEASURE ON TRAIN DATA SET

MODEL PERFORMANCE TOOLS	GOODNESS OF THE TRAINING MODEL
KS	91.56%
AUC	98.24%
GINI	87.23%
CONCORDANCE	96.71%
ACCURACY	98.8%
CLASSIFICATION ERROR RATE	1.2%

PERFORMANCE MEASURE ON THE TEST DATASET – CART MODEL

The procedure is exactly the same to carry on the performance measure on the test dataset of the CART model. Also, same type of packages have been used for this procedure.

THE CONFUSION MATRIX OUTPUT:

```
      0    1
  0 1351    5
  1   25  119
[1] 0.02
```

Error rate due to mis-classification is about 2%.

The columns of the CART_Test dataset with three deciles are illustrated in the table below.

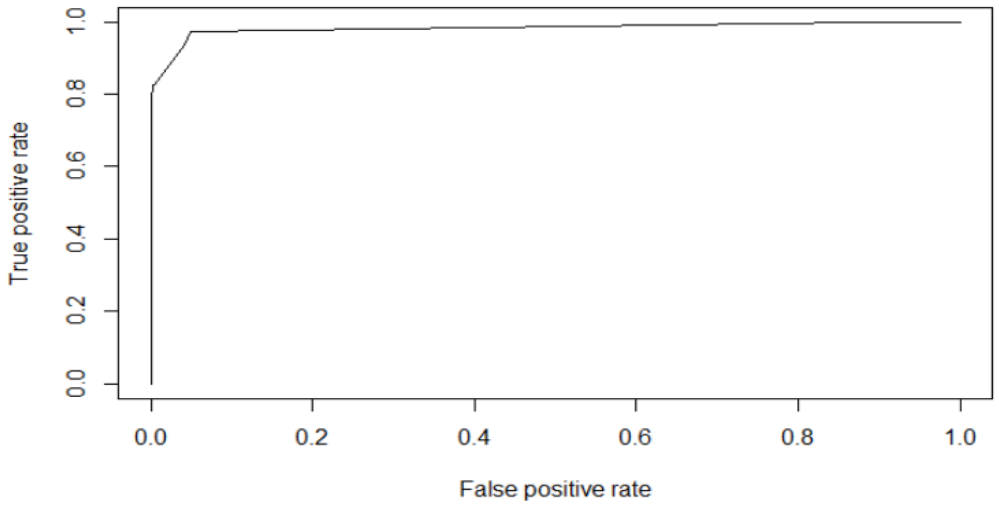
CD.Account <fctr>	Online <fctr>	CreditCard <fctr>	predict.class <fctr>	prob_1 <dbl>	deciles <fctr>
0	0	0	0	0.004209721	[0.00421,0.117)
0	1	0	0	0.004209721	[0.00421,0.117)
0	0	0	1	0.978902954	[0.117,1]
0	0	0	0	0.117283951	[0.117,1]
0	0	0	0	0.004209721	[0.00421,0.117)
0	1	1	0	0.004209721	[0.00421,0.117)
0	0	1	0	0.000000000	[0,0.00421)
0	1	0	0	0.004209721	[0.00421,0.117)
0	0	0	0	0.004209721	[0.00421,0.117)
0	0	0	0	0.004209721	[0.00421,0.117)

The below summary shows the 1s and 0s of the personal loan column based on different deciles.

Additionally, the "rrate" column illustrates on the response rate of the customers. The first decile contains the maximum number of response rate. The second and third deciles shows a very poor response rate.

Deciles	Cnt	Cnt_ Loan1	Cnt_ Loan0	Res%	Cum_ Res	Cum_ nonRes	Cum_rel_ Resp	Cum_rel_ nonResp	KS
[0.117,1]	192	135	57	70.31	135	57	93.75	4.20	89.55
[0.0021,0.117)	1143	9	1134	0.79	144	1191	100.00	87.83	12.17
[0,0.00421)	165	0	165	0.00	144	1356	100.00	100.00	0.00

The below graph illustrates the False Positive Rate Vs True Positive Rate.



THE SUMMARY OF MODEL PERFORMCE MEASURE ON TEST DATA SET

MODEL PERFORMANCE TOOLS	GOODNESS OF THE TRAINING MODEL
KS	92.43%
AUC	98.27%
GINI	87.64%
CONCORDANCE	96.87%
ACCURACY	98.0%
CLASSIFICATION ERROR RATE	2.0%

DISCUSSION ON CART MODEL PERFORMANCE MEASURE

MODEL PERFORMANCE TOOLS	Train	Test	% Difference
KS	91.56%	92.43%	-0.87
AUC	98.24%	98.27%	-0.03
GINI	87.23%	87.64%	-0.41
CONCORDANCE	96.71%	96.87%	-0.16
ACCURACY	98.8%	98.0%	0.8%
CLASSIFICATION OF ERROR	1.2%	2.0%	-0.8%

The variation between the Train and Test model performance values lies less than +/- 10% which is a good indication of a good model. Hence, we can say the developed CART model have good prediction making abilities.

RANDOM FOREST MODELING

As explained in the previous chapter, original data is divided into 70% into 30% ratio. Beginning of this chapter will discuss the building of Random Forest in "Customer_Train" dataset.

Set seed value is 1000 throughout the model development process to maintain a uniform random number generation.

```
Call:
 randomForest(formula = Personal.Loan ~ ., data = Customer_Train[,      c(-1, -5)], ntree = 501,
mtry = 10, nodesize = 10, importance = TRUE)
               Type of random forest: classification
                     Number of trees: 501
No. of variables tried at each split: 10

        OOB estimate of  error rate: 1.4%
Confusion matrix:
     0   1 class.error
0 3151  13 0.004108723
1   36 300 0.107142857
```

Out of Bag (OOB) error rate is about 1.4%. This is the primary measure of success return.

CONFUSION MATRIX

Every Mis-classification is on the OFF DIAGONAL i.e: 13 and 36.

Every Right classification is on the DIAGONAL i.e: 3151 and 300.

OBB error rate of the top 10 and bottom 10 of the dataset is shown below.

```
             OOB           0         1
 [1,] 0.01392111 0.004240882 0.1140351
 [2,] 0.01772880 0.005319149 0.1304348
 [3,] 0.01945080 0.006339814 0.1406250
 [4,] 0.01689189 0.005233645 0.1263158
 [5,] 0.01635220 0.005571031 0.1168831
 [6,] 0.01659626 0.006008011 0.1163522
 [7,] 0.01453575 0.004594683 0.1080247
 [8,] 0.01551977 0.004857513 0.1162080
 [9,] 0.01506373 0.004809234 0.1111111
[10,] 0.01641232 0.006371456 0.1107784
              OOB           0         1
[492,] 0.01400000 0.004108723 0.1071429
[493,] 0.01400000 0.004108723 0.1071429
[494,] 0.01428571 0.004424779 0.1071429
[495,] 0.01371429 0.004108723 0.1041667
[496,] 0.01400000 0.004108723 0.1071429
[497,] 0.01371429 0.004108723 0.1041667
[498,] 0.01371429 0.004108723 0.1041667
[499,] 0.01371429 0.004108723 0.1041667
[500,] 0.01371429 0.004108723 0.1041667
[501,] 0.01400000 0.004108723 0.1071429
```

As the rows get larger and larger it can be seen that OOB error rate column approaches the 1.4% as given in OOB error rate from the classification output.

The Error rate plot is illustrated below.

The x-axis is number of trees and y-axis is the Error rate. Green graph is the error for predicting target equal 1. Red graph is error rate for predicting target equal 0. OBB error rate plot tells the number of trees that any trees more than 110 is not valuable.

Mean Decreasing Accuracy & Mean Decrease Gini:

	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
Age..in.years.	22.040478	-5.4530753	21.315733	9.1971242
Experience..in.years.	14.158326	-4.0340172	12.782511	5.8614022
Income..in.K.month.	265.487368	140.8449598	292.019757	198.8970408
Family.members	174.489744	74.1966379	183.424016	87.5405515
CCAvg	37.492484	43.2906572	44.193131	36.4056265
Education	232.177194	101.0682376	246.215930	220.5966483
Mortgage	5.243700	-3.3654478	3.813042	3.4353127
Securities.Account	-3.280842	-0.3061908	-2.697117	0.3214228
CD.Account	18.436806	22.7711125	26.560453	12.9284071
Online	1.542013	0.4203794	1.671824	0.7910091
CreditCard	-1.248429	-0.8266217	-1.587408	0.4749191

The "importance" R function talks about accuracy.

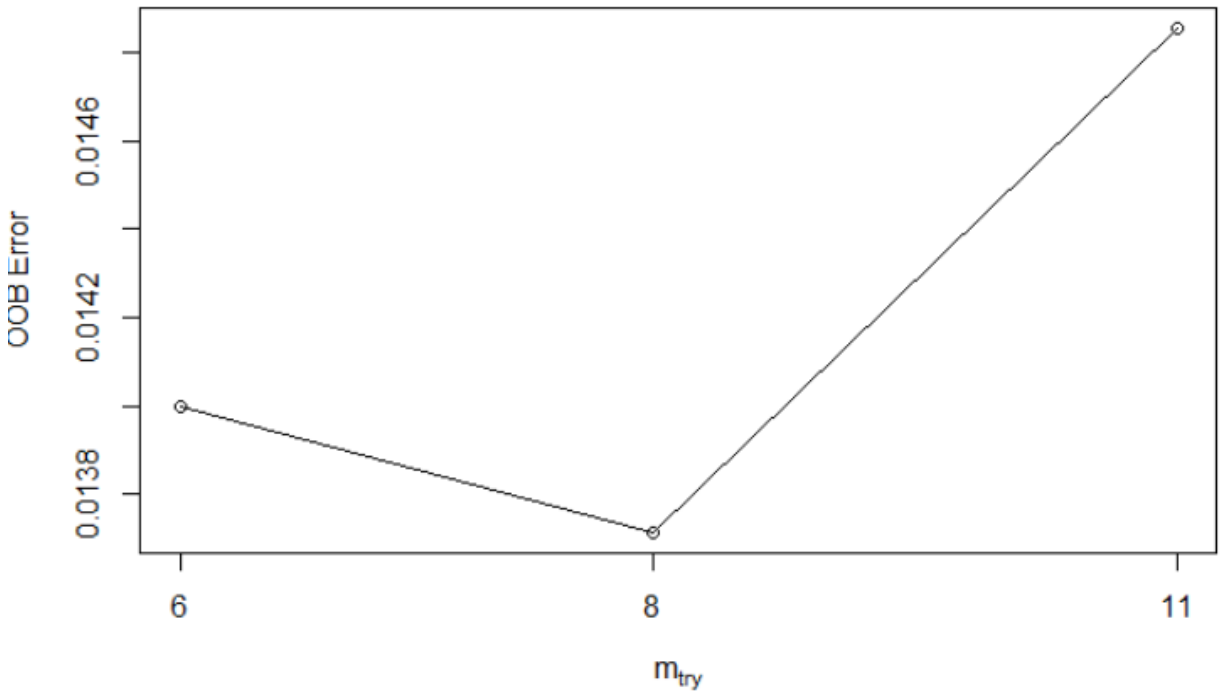
Coloumn 1 - Decreasing accuracy for target predicting variable 0

Coloumn 2 - Decreasing accurcy for target predicting variable 1

Coloumn 3 - Mean Decreasing Accuracy

Coloumn 4 - Mean Decrease Accuracy GINI

Using an original dataset we created a random forest. In order to measure the accuracy of the random forest we require certain technique by evaluating how important each variables are. In other words, we are looking at if one of the variable gets shuffled up the original order in what percentage our accuracy gets effected. Mean Decreasing accuracy indicates that larger the number more important the variable is. Based on the above output, "Education", "Income per month" and Family. Members" are the highly influencing variables in predicting the target variable "Personal. Loan". The "Mean Decrease Gini" measure indicates the affected rate on the accuracy on the prediction if one of the variables are been removed. According to this measure "Income", "Education" and "Family members" variables are most important variable in predicting the potential customers who could be converted into a potential customer for a personal loan scheme.



The above tuned random forest plot tells that error rate increases at mtry= 6 & 11. Therefore, it stops at 8.

Tuned Random forest was run with mtry=8, and Mean and Gini decreasing accuracy was tabulated below.

	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
Age..in.years.	12.461606541	-2.8337554	12.0918426	3.6926225
Experience..in.years.	4.331263610	-3.5499981	3.2205284	1.6741022
Income..in.K.month.	178.671313887	116.0150566	193.3578300	178.7302347
Family.members	100.815289750	50.8557068	99.7102811	80.7304640
CCAvg	23.711307016	24.7324644	27.0655331	45.4025881
Education	197.665304951	88.1201182	198.2135462	195.5599292
Mortgage	4.361054232	-5.7464785	3.0932409	2.0222795
Securities.Account	-0.005315552	-1.1425587	-0.9144144	0.1068576
CD.Account	12.024433716	16.0828768	17.8135682	15.7995675
Online	1.462957488	-1.7364671	1.2762488	0.1495830
CreditCard	1.546910904	0.2109917	1.6946505	0.1519546

It can be said that, “Income”, ”Family Members” and “Education” attributes play vital role if the observations were shuffled or removed. The marketing team must focus on these attribute and information of the customer and redesign their marketing strategies.

Now the random forest trees will be used, to do the prediction and the probability of the success rate on the personal loan attribute. The below screenshots shows the 10 observations with their respective predictive class and probability of success.

Personal.Loan <fctr>	Securities.Account <fctr>	CD.Account <fctr>	Online <fctr>	CreditCard <fctr>	Predict.class <fctr>	Prob_1 <dbl>
1	0	0	0	0	1	1.000
0	0	0	0	0	0	0.000
1	0	0	0	0	1	0.990
0	1	0	0	1	0	0.000
0	0	0	1	0	0	0.000
0	0	0	1	0	0	0.000
0	0	0	1	0	0	0.000
0	1	0	0	0	0	0.000
0	0	0	1	0	0	0.000
0	0	0	0	0	0	0.000

We next find the probability threshold that for the top decile. The choice of what threshold you use is quite subjective and depends on the benefits of having Personal Loan=1 vs the cost of sending out, say mailers, to each customer. Since the threshold for the top decile is lower than 0.5, I decided to use 0.5.

```
    0%    10%    20%    30%    40%    50%    60%    70%    80%    90%   100%
0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0020 0.2162 1.0000
```

Based on the above prediction and probability column output for first 10 rows it can be said that, for rows with probability with a low value has prediction of "ZERO". Evidently, for the 11th row with really high probability value of 99% gives a prediction of 1.

```
mean((Customer_Train$Personal.Loan[Customer_Train$Prob_1>threshold])=="1")
```

```
[1] 0.9899329
```

Based on the output we could say that by sending customers who have threshold value of more than 50% has high chance of responding. Probability of response is about almost 98.99%.

From this analysis marketing team will be able to forecast the response rate of customers with certain threshold value. Very impressive tool to preplan the marketing campaign.

PERFORMANCE MEASURE ON THE TRAINING DATASET – RANDOM FOREST MODEL

The following performance measures were measured to identify the accuracy of our model.

1. Rank Ordering Method
2. Kolomogorov-Smirnov (K-S Chart)
3. Area Under Curve (AUC)
4. GINI Coefficient
5. Concordance – Discordance Ratio

As explained above, the main purpose to find out how well this model would perform: The first step of this process is to start with the confusion matrix on the Actual Vs Predicted Value on Personal Loan from the pruned decision trees.

CONFUSION MATRIX:

The output of the confusion matrix is given below:

```
      0    1
0  3161    3
1    38  298
[1] 0.01171429
```

The mis-classification through the off diagonal entries. The error rate of the confusion matrix to be approximately 1.2%.

RANK ORDERING METHOD:

```
 [1] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
    0%    10%    20%    30%    40%    50%    60%    70%    80%    90%   100%
0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0020 0.1882 1.0000
```

So, the output above illustrates the number chops that can be done in the Customer_Train dataset. The bucket combination is isted below:

Bucket 1: 0% to 80%

BUcket 2: 80% > 90%

Bucket 3 :90% > 100%

Let's compute the deciles using R "cut" function.

CD.Account <fctr>	Online <fctr>	CreditCard <fctr>	Predict.class <fctr>	Prob_1 <dbl>	deciles <fctr>
0	0	0	0	0.000	[0,0.002)
0	0	0	0	0.000	[0,0.002)
0	0	0	0	0.002	[0.002,0.188)
0	0	1	0	0.000	[0,0.002)
0	1	0	0	0.000	[0,0.002)
0	0	1	0	0.000	[0,0.002)
0	1	0	0	0.000	[0,0.002)
0	0	0	0	0.008	[0.002,0.188)
0	1	0	0	0.000	[0,0.002)
0	1	0	0	0.000	[0,0.002)

As mentioned before, the all entries with a decile value of 0 to 0.002 will be put in one bucket. Values which are greater than 0.002 and up to 0.188 will be grouped separately. Last but not the least, values above 0.118 and up to 1 will be grouped into a separate group.

Numeric count per decile is given below.

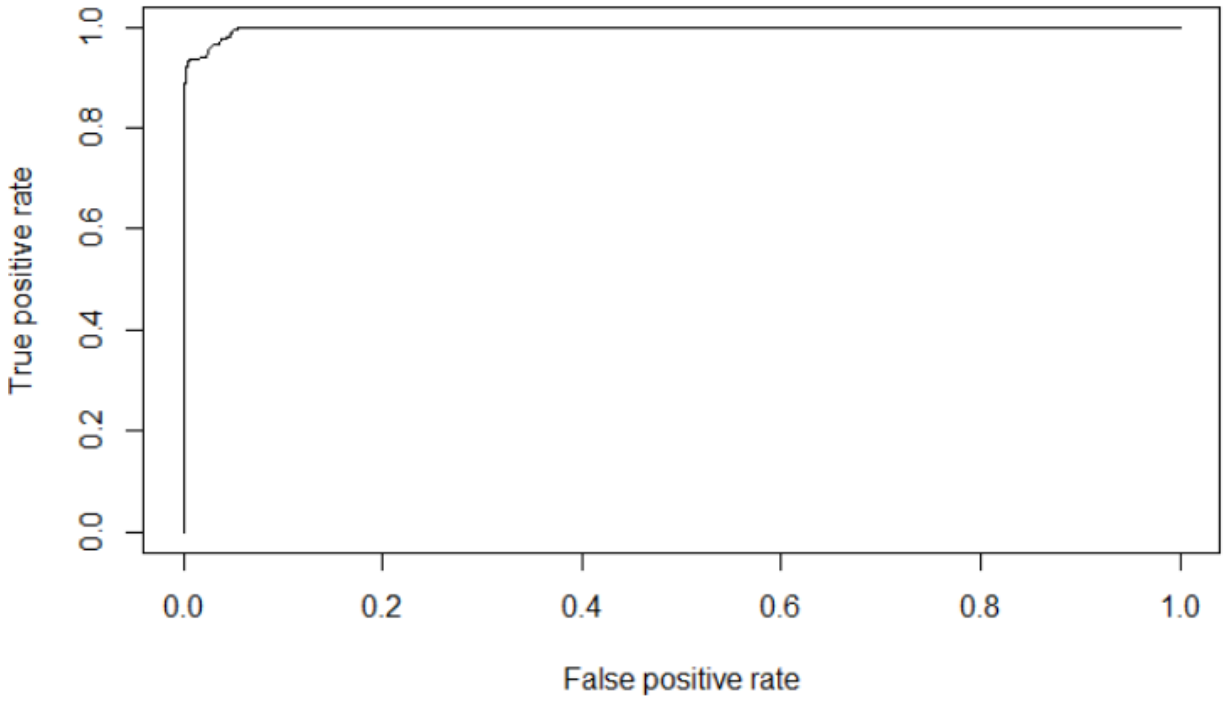
deciles <fctr>	cnt <int>
[0.188,1]	350
[0.002,0.188)	398
[0,0.002)	2752

Additionally, the "rrate" column illustrates on the response rate of the customers. The first decile contains the maximum number of response rate. The second and third deciles shows a very poor response rate.

Deciles	Cnt	Cnt_ Loan1	Cnt_ Loan0	Res%	Cum_ Res	Cum_ nonRes	Cum_rel_ Resp	Cum_rel_ nonResp	KS
[0.188,1]	350	315	35	90.00	315	35	93.75	1.11	92.64
[0.002 ,0.188)	398	21	377	5.28	336	412	100.00	13.02	86.98
[0,0.002)	2752	0	2752	0.00	336	3164	100.00	100.00	0.00

THE KS & AREA UNDER CURVE:

As discussed in the previous chapter, the R library packages used in this cahpter are same as above.



THE SUMMARY OF MODEL PERFORMCE MEASURE ON TRAIN DATA SET

MODEL PERFORMANCE TOOLS	GOODNESS OF THE TRAINING MODEL
KS	94.71%
AUC	99.76%
GINI	90.62%
CONCORDANCE	99.75%
ACCURACY	98.83%
CLASSIFICATION ERROR RATE	1.17%

PERFORMANCE MEASURE ON THE TESTING DATASET – RANDOMFOREST MODEL

CONFUSION MATRIX:

The output of the confusion matrix is given below:

```
      0    1
0  1353    3
1    30  114
[1] 0.022
```

The mis-classification through the off diagonal entries. The error rate of the confusion matrix to be approximately 2.2%.

RANK ORDERING METHOD:

```
 [1]  0.0  0.1  0.2  0.3  0.4  0.5  0.6  0.7  0.8  0.9  1.0
    0%    10%    20%    30%    40%    50%    60%    70%    80%    90%   100%
0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0020 0.1464 1.0000
```

So, the output above illustrates the number chops that can be done in the Customer_Test dataset. The bucket combination is listed below:

Bucket 1: 0% to 80%

BUcket 2: 80% > 90%

Bucket 3 :90% > 100%

Let's compute the deciles using R "cut" function.

CD.Account <fctr>	Online <fctr>	CreditCard <fctr>	Predict.class <fctr>	Prob_1 <dbl>	deciles <fctr>
0	0	0	0	0.000	[0,0.002)
0	1	0	0	0.000	[0,0.002)
0	0	0	1	0.992	[0.146,1]
0	0	0	0	0.474	[0.146,1]
0	0	0	0	0.000	[0,0.002)
0	1	1	0	0.000	[0,0.002)
0	0	1	0	0.000	[0,0.002)
0	1	0	0	0.000	[0,0.002)
0	0	0	0	0.000	[0,0.002)
0	0	0	0	0.000	[0,0.002)

As mentioned before, the all entries with a decile value of 0 to 0.002 will be put in one bucket. Values which are greater than 0.002 and up to 0.146 will be grouped separately. Last but not the least, values above 0.146 and up to 1 will be grouped into a separate group.

Numeric count per decile is given below.

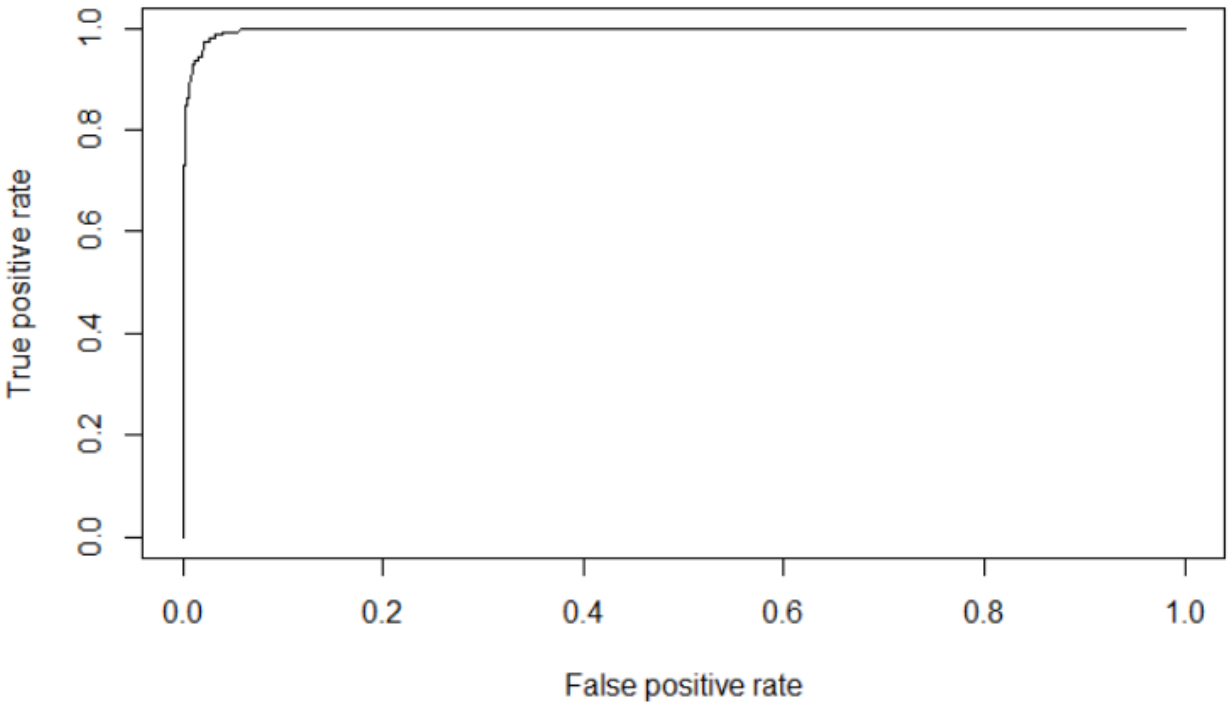
deciles <fctr>	cnt <int>
[0.146,1]	150
[0.002,0.146)	180
[0,0.002)	1170

Additionally, the "rrate" column illustrates on the response rate of the customers. The first decile contains the maximum number of response rate. The second and third deciles shows a very poor response rate.

Deciles	Cnt	Cnt_ Loan1	Cnt_ Loan0	Res%	Cum_ Res	Cum_ nonRes	Cum_rel_ Resp	Cum_rel_ nonResp	KS
[0.146,1]	150	134	16	89.33	134	16	93.06	1.18	91.88
[0.002 ,0.146)	180	10	170	5.56	144	186	100.00	13.72	86.28
[0,0.002)	1170	0	1170	0.00	144	1356	100.00	100.00	0.00

THE KS & AREA UNDER CURVE:

As discussed in the previous chapter, the R library packages used in this cahpter are same as above.



THE SUMMARY OF MODEL PERFORMCE MEASURE ON TRAIN DATA SET

MODEL PERFORMANCE TOOLS	GOODNESS OF THE TESTING MODEL
KS	95.44%
AUC	99.73%
GINI	91.07%
CONCORDANCE	99.73%
ACCURACY	97.8%
CLASSIFICATION ERROR RATE	2.2%

DISCUSSION ON CART MODEL PERFORMANCE MEASURE

MODEL PERFORMANCE TOOLS	Train	Test	% Difference
KS	94.71%	95.44%	-0.73%
AUC	99.76%	99.73%	0.03%
GINI	90.62%	91.07%	-0.45%
CONCORDANCE	99.75%	99.73%	0.02%
ACCURACY	98.83%	97.8%	1.03%
CLASSIFICATION OF ERROR	1.17%	2.2%	-1.03%

The variation between the Train and Test model performance values lies less than +/- 10% which is a good indication of a good model. Hence, we can say the developed CART model have good prediction making abilities.

6. CONCLUSION

Based on the overall findings it can be said that output of both performance measure of CART model and Random Forest Model shows a similar in nature. Both models can used to predict and identify the potential customers. However, I recommend the marketing team to focus theor campaign based on the financial background and educational background. The number of family members also an influencing attribute when it comes to marketing for personal loan. By considering all these aspects, if the marketing campaigns could be designed there is very high probability of getting a higher response rate on Thera Bank Personal loan scheme.