German Credit Data Analysis :

The given German Credit dataset contains **1000** observations of **30** variables.
The variable RESPONSE indicates whether a certain applicant is “Good” or “Bad” with a 1 or a 0 respectively.
Proportion of Good to Bad = Sum (Good) / Sum(Good + Bad) = 700

- Steps taken In R,
  sum(GERMANCREDIT1_$RESPONSE) = 700

Hence, out of 1000 cases 70% of the applicants are good.

1. Dealing with Missing Values

Yes, there are missing values.
Even though the description of the dataset mentions the presence of 0’s and 1’s from NEW_CAR to RETRAINING, there were no 0’s present in the given dataset. Assuming the missing values were ‘0’, I decided to impute them into the dataset.

- Steps taken in R,

```
GERMANCREDIT1_$NEW_CAR[is.na(GERMANCRDIT$NEW_CAR)]□0
GERMANCREDIT1_$USED_CAR[is.na(GERMANCREDIT$USED_CAR)]□0
GERMANCREDIT1_$FURNITURE[is.na(GERMANCRDIT$FURNITURE)]□0
GERMANCREDIT1_$`RADIO/TV`[is.na(GERMANCRDIT$`RADIO/TV`)]□0
GERMANCREDIT1_$EDUCATION[is.na(GERMANCRDIT$ EDUCATION `)]□0
GERMANCREDIT1_$RETRAINING[is.na(GERMANCRDIT$ RETRAINING `)]□0
```

In column “age” there were values missing. Since it is numerical data, the median of all the values of AGE was taken and imputed into the missing values, this will preserve the natural median of the set.

- Steps taken in R

```
GERMANCREDIT1_$AGE[is.na(GERMANCREDIT1_$AGE)] <-
median(GERMANCREDIT1_$AGE, na.rm= TRUE)
```

In column “PERSONAL_STATUS”, there were 300 values missing. The mode of the values in Personal_Status was taken as it is a categorical data and imputed into the missing rows.

- Steps taken in R

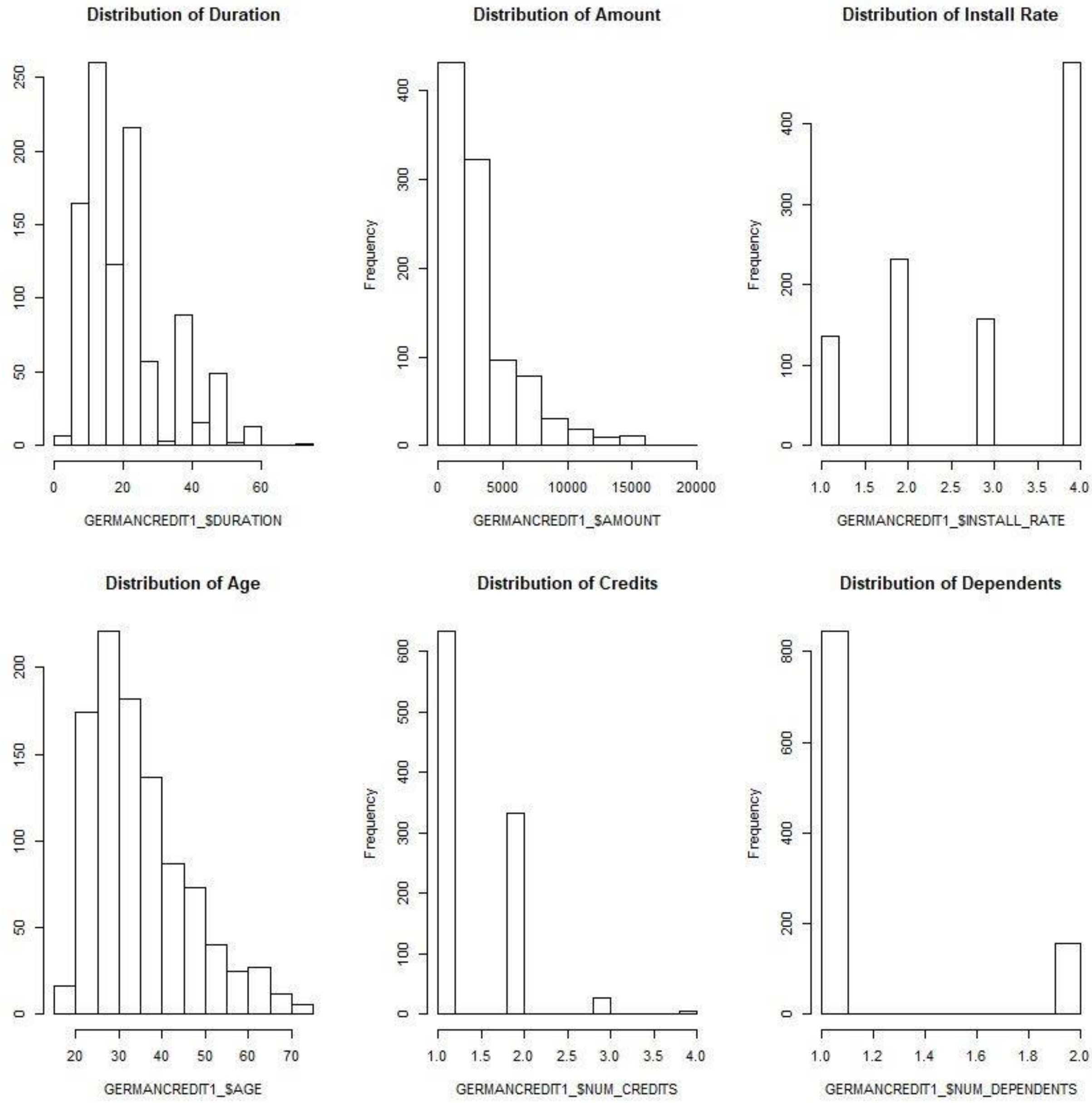
```
GERMANCREDIT1_$PERSONAL_STATUS[is.na(GERMANCREDIT1_$PERSONAL_STATUS)]
<- mode(GERMANCREDIT1_$PERSONAL_STATUS, na.rm=TRUE)
```

Description of numerical Variables:

V1: Duration, V2: Amount, V3: Install Rate, V3: Age, V4: Num Credits, V5: Dependants

	V1	V2	V3	V4	V5	V6
median	18.0000000	2.319500e+03	3.00000000	33.0000000	1.00000000	1.00000000
mean	20.9030000	3.271156e+03	2.97300000	35.4610000	1.40700000	1.15500000
SE.mean	0.3813332	8.925924e+01	0.03537686	0.3580289	0.01826704	0.01145016
CI.mean.0.95	0.7483059	1.751571e+02	0.06942149	0.7025749	0.03584617	0.02246912
var	145.4150060	7.967212e+06	1.25152252	128.1846637	0.33368468	0.13110611
std.dev	12.0588145	2.822625e+03	1.11871467	11.3218666	0.57765447	0.36208577
coef.var	0.5768940	8.628830e-01	0.37629152	0.3192766	0.41055755	0.31349417

Distribution of Numeric Variables:

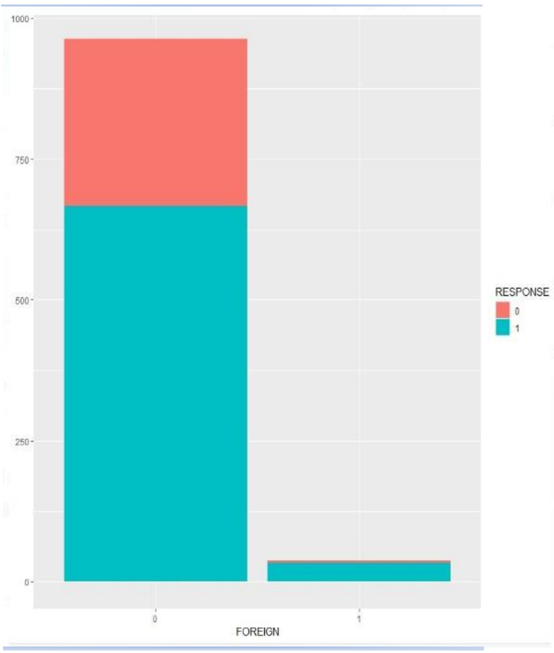


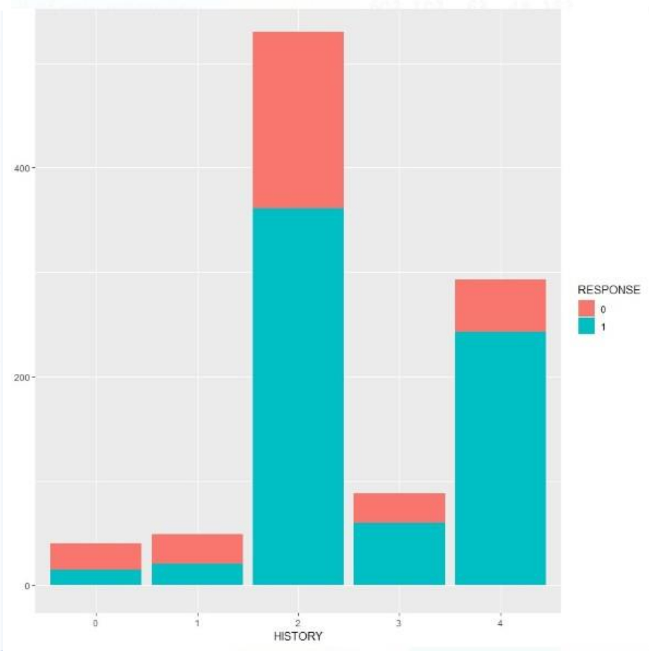
Distribution is skewed right for Duration, Amount, Age and Credits which indicates that the mean is greater that the median. Distribution of Dependents and Install Rate is a normally distributed curve

Frequencies of Categorical Variables:

```
> table(GERMANCREDIT1_$CHK_ACCT)

  0   1   2   3 
274 269  63 394 
> table(GERMANCREDIT1_$CHK_ACCT)

  0   1   2   3 
274 269  63 394 
> table(GERMANCREDIT1_$HISTORY)

  0   1   2   3   4 
 40  49 530  88 293 
> table(GERMANCREDIT1_$SAV_ACCT)

  0   1   2   3   4 
603 103  63  48 183 
> table(GERMANCREDIT1_$EMPLOYMENT)

  0   1   2   3   4 
 62 172 339 174 253 
> table(GERMANCREDIT1_$PRESENT_RESIDENT)

  1   2   3   4 
130 308 149 413 
```

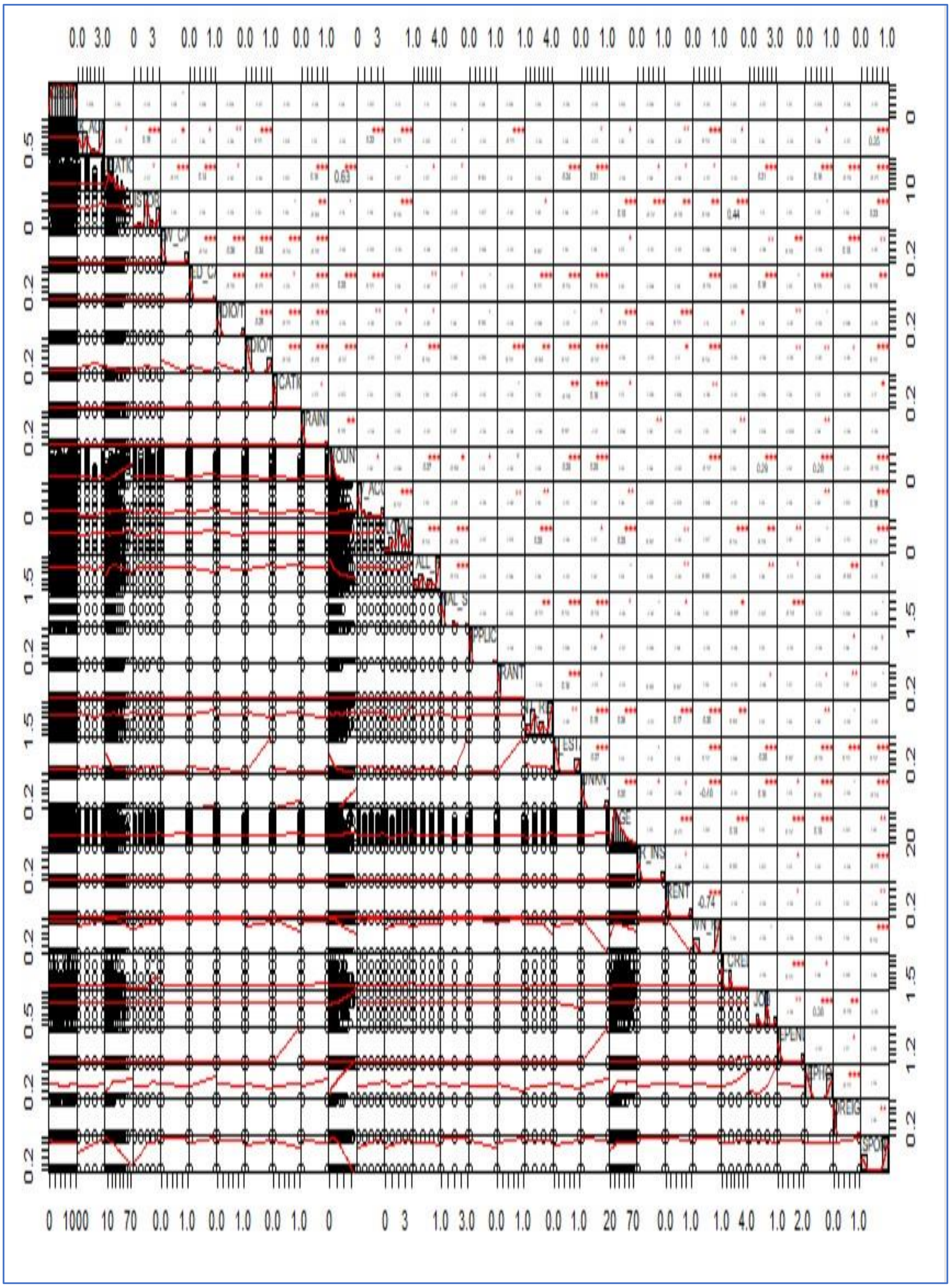
```
> table(GERMANCREDIT1_$NEW_CAR)

  0   1 
766 234 
> table(GERMANCREDIT1_$USED_CAR)

  0   1 
897 103 
> table(GERMANCREDIT1_$FURNITURE)

  0   1 
819 181 
> table(GERMANCREDIT1_$`RADIO/TV`)

  0   1 
720 280 
> table(GERMANCREDIT1_$EDUCATION)

  0   1 
950  50 
> table(GERMANCREDIT1_$RETRAINING)

  0   1 
903  97 
```





There is a huge proportion of applicants having the history of -existing credits paid back duly till now. The visualization shows the number of cases who have returned the credit duly till now and the ones who have not. This can be an important predictor to realize which applicants could be possible ‘bad’ credit cases. Also notice that there is a large base of new applicants without a checking account

Attempted to see which of the independent variables have the highest correlation so as to help formulate our model going forward. It can be seen that there are many highly correlated variables.
On the bottom of the diagonal the bivariate scatter plots with a fitted line are displayed, top half of the diagonal displays the significance level as stars



It can be suspect that the Credit history, Average balance in savings account, Instalment rate as % of disposable income and Checking account status will show at most importance in the outcome in Response. Since the variables are highly correlated this assumption could be incorrect.

Decision Tree:

30 Decision trees were developed on the full data. Different combinations of minsplit, maxdepth and cp were applied. The top four with the highest accuracy is given in the data below. The second model shows the highest accuracy with 75%

Models Generated	**Accuracy**
Model 1	0.715
Model 2	0.75
Model 3	0.715
Model 4	0.73

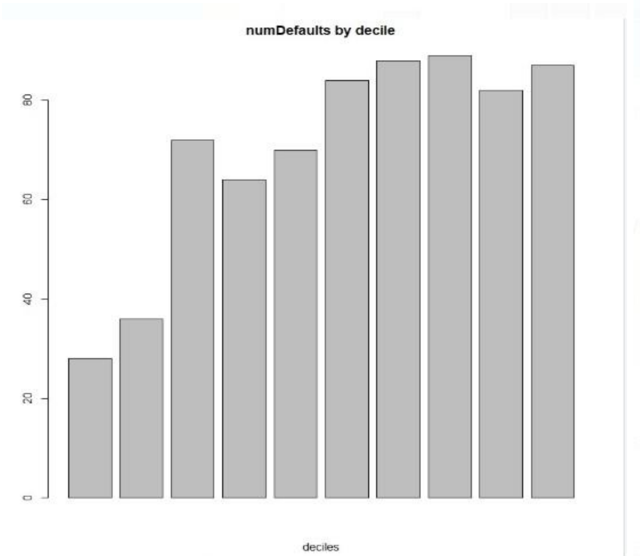
The parameters were chosen such that the (maxdepth)max levels of the trees were long enough for the classification to be accurate but not at the risk of overfitting. I also wanted (minsplit) the minimum number of observations that must exist in a node for a spilt to be attempted to be low along with moderate range of cp which decides whether to split if the result does not produce any improvement in fit.

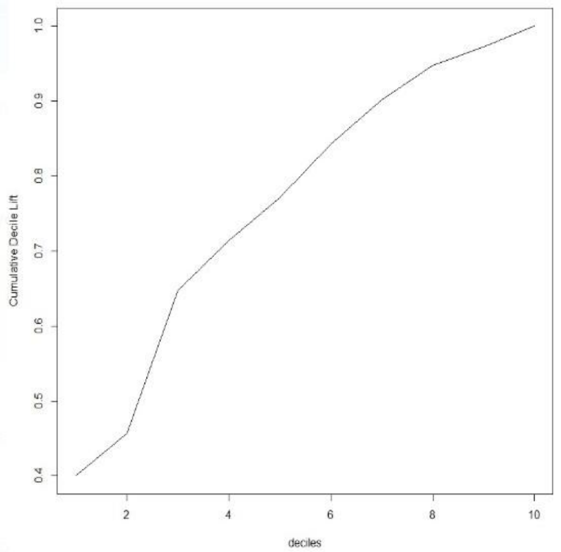
The parameters present in the top four chosen models are given in the below table.

Models Generated	Parameters Chosen
Model 1	rpModel1<- rpart(RESPONSE~., data = GERMANCREDIT1_, method=”class”)
Model 2	rpModel2<- rpart(RESPONSE~., data = GERMANCREDIT1_, method=”class”,maxdepth =15, minspilt=15, xval =10,cp=.01,parms=list(split=’information’)
Model 3	rpModel3<- rpart(RESPONSE~., data = GERMANCREDIT1_, method=”class”,maxdepth =15, minspilt=40, xval =10,parms=list(split=’gini’)
Model 4	rpModel4<- rpart(RESPONSE~., data = GERMANCREDIT1_, method=”class”,maxdepth =5, minspilt=50, xval =15,cp=.02,parms=list(split=’information’)

	CP	nsplit	rel error	xerror	xstd
1	0.05166667	0	1.0000000	1.0000000	0.04830459
2	0.04666667	3	0.8400000	0.9900000	0.04816534
3	0.01833333	4	0.7933333	0.9533333	0.04763332
4	0.01000000	6	0.7566667	0.9033333	0.04685189

During simulation it was focused primarily on the minsplit and maxdepth on our training dataset. Having a minsplit greater than 15 lowered the accuracy given and had more levels which increased the complexity.





The decile chart and lift curve obtained was the mirror image of what is ideally used to asses them. The decile chart obtained starts with lowest to highest instead of it being the other way around. The same is seen in the lift curve , the curve has an increasing slope instead of a curve where the slope is decreasing

Pruning Table of Model 2: This table tells us the information about the pruning done by the algorithm

	CP	nsplit	rel error	xerror	xstd
1	0.05166667	0	1.0000000	1.0000000	0.04830459
2	0.04666667	3	0.8400000	0.9900000	0.04816534
3	0.01833333	4	0.7933333	0.9533333	0.04763332
4	0.01000000	6	0.7566667	0.9033333	0.04685189

No, this is not a reliable model as it hasn't been tested on unseen data yet.

Tried two different approaches at splitting. One using information and another using gini. As it can be seen from the table below, both of the, have similar accuracy. Hence we chose information tables to be carried on in our study.
Result of Decision trees with ‘information’ at a 50/50 split on data

Models	CP	Maxdepth	Minspilt	Train Accuracy	Test Accuracy
Model1	0.01	10	10	.81	.715
Model2	.01	15	15	.76	.75
Model3	.0001	15	40	.80	.715
Model4	.02	5	50	.74	.75

Result of Decision trees with ‘gini’ at a 50/50 split on data

Models	CP	Maxdepth	Minspilt	Train Accuracy	Test Accuracy
Model1	0.01	10	10	.805	.715
Model2	.01	15	15	.7675	.745
Model3	.0001	15	40	.803	.715
Model4	.02	5	50	.74	.73

Performance Measures:
It can be seen from the table that the model with the highest AUC is Model 2.

Models	Precision		Sensitivity		Recall		AUC
	0	1	0	1	0	1	
Model1	.5757	.7425	.400	.8131	.3064	.8985	.6519
Model2	.6222	.7806	.5283	.8259	.4516	.8768	.6764
Model3	.5531	.7647	.4770	.8041	.4193	.8478	.6920
Model4	.6111	.7560	.4489	.8211	.3548	.8985	.6736

In Model 2 contains these following parameters. Having minspilt at 15 prunes the tree without over-fitting it. It also performs better on unseen data.

CP=.01	Maxdepth=15	Minsplit=15

C5.0

Experimented with many different types of parameters on the test data only the rules provided below showed some changes to our accuracy measures.

Results:

Tree1:

Accuracy= .7

	True	
Predicted	0	1
0	21	19
1	41	119

Tree2:

Accuracy= .705

	True	
Predicted	0	1
0	17	14
1	45	124

Tree3:

Accuracy= .705

	True	
Predicted	0	1
0	17	14
1	45	125

Chose Tree2 out of Tree2 and Tree3 since they displayed almost the same results. Considered Tree 2 to be the best model and below are the rules defined by the given model.

Iterated through multiple parameters and through different accuracies. The precision, recall and F1 measure of our Trees.

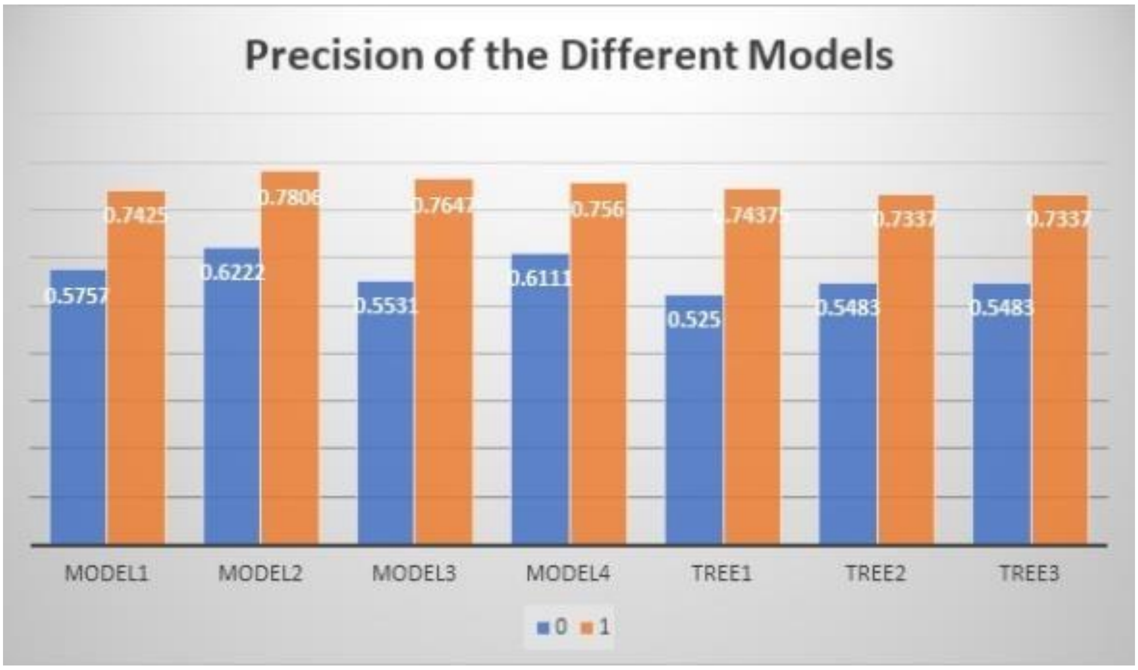
Trees	Precision		Recall		F1		AUC
	0	1	0	1	0	1	
Tree1	.525	.74375	.3337	.8625	.4117	.7986	.6519
Treel2	.5483	.7337	.2741	.8985	.3655	.8078	.6764
Tree3	.5483	.7337	.2741	.8985	.3655	.8078	.6920

Result Rules:

```
Rule 1: (12, lift 3.1)
        CHK_ACCT in {0, 1}
        HISTORY = 0
        OWN_RES = 0
        -> class 0  [0.929]

Rule 2: (23/1, lift 3.1)
        CHK_ACCT in {0, 1}
        DURATION > 47
        EDUCATION = 0
        SAV_ACCT = 0
        -> class 0  [0.920]
```

```
Rule 3: (9, lift 3.1)
        DURATION > 15
        DURATION <= 21
        USED_CAR = 0
        SAV_ACCT = 0
        EMPLOYMENT = 2
        GUARANTOR = 0
        PROP_UNKN_NONE = 0
        NUM_CREDITS <= 1
        NUM_DEPENDENTS <= 1
        -> class 0  [0.909]

Rule 4: (8, lift 3.0)
        OBS > 111
        CHK_ACCT in {0, 1}
        HISTORY in {1, 2, 3}
        SAV_ACCT = 0
        EMPLOYMENT = 3
        REAL_ESTATE = 0
        JOB = 2
        NUM_DEPENDENTS <= 1
        -> class 0  [0.900]
```

Included a bar graph that is able to provide visually clearer changes in precision among the different rpart trees and C5.0 trees. There very minute changes among the trees.


Precision of the Different Models
0.7425
0.7806
0.7647
0.756
0.74375
0.7337
0.7337
0.5757
0.6222
0.5531
0.6111
0.525
0.5483
0.5483
MODEL1
MODEL2
MODEL3
MODEL4
TREE1
TREE2
TREE3
0
1

Yes, there is instability in the models. Which is - using different seeds the accuracy changes along with changes in the confusion matrix. For demonstration purpose, Tree1 is used

Tree1: (Original)
Seed = 123
Accuracy= .7

	True	
Predicted	0	1
0	21	19
1	41	119

Tree1.1
Seed = 321
Accuracy= .716

	True	
Predicted	0	1
0	50	31
1	41	78

Tree1.2
Seed = 276
Accuracy= .722

	True	
Predicted	0	1
0	61	26
1	27	86

Tree1.3
Seed = 500
Accuracy= .722

	True	
Predicted	0	1
0	59	26
1	46	69

Tree1.4
Seed = 453
Accuracy= .708

	True	
Predicted	0	1
0	46	36
1	49	69

As we can see some of the trees provide the same accuracy, while some do not. However the range of accuracies given are close to the original

Variable importance for rpart and C5.0 models

Partitioned the data into three different splits: 50/50, 70/30 & 80/20. The measurements of the different variables taken across the model and tree have similar values. It can be concluded that a large training dataset is not a necessary factor, one can work with only the 50% of the dataset and still get similar results.

rpart:

	Accuracy	**Precision**		**Recall**		**F1**		**AUC**	**Optimal Threshold**
		0	1	0	1	0	1		
50/50	.71	.60	.7222	.1935	.9420	.2926	.8176	.6077	.4749
70/30	.705	.5319	.7581	.4032	.8405	.4587	.7972	.6361	.0695
80/20	.685	.4938	.8951	.6451	.7028	.5594	.7548	.7034	.0857

C5.0

	Accuracy	**Precision**		**Recall**		**F1**		**AUC**	**Optimal Threshold**
		0	1	0	1	0	1		
50/50	.72	.5460	.765	.345	.876	.422	.786	.587	.1405
70/30	.715	.514	.714	.326	.845	.424	.756	.567	.1405
80/20	.718	.5250	.7437	.3387	.8623	.4117	.7986	.5785	.1405

Model 1: Variable Importance using the same parameters but with a 50/50 split

```
Variable importance
    CHK_ACCT         DURATION   HISTORY   AMOUNT   SAV_ACCT      AGE   OBS
          27               12        11       11          9        6     4
INSTALL_RATE PRESENT_RESIDENT   NEW_CAR USED_CAR EMPLOYMENT  RADIOTV   JOB
           4                3         2        2          2        2     1
 NUM_CREDITS   PROP_UNKN_NONE
           1                1
```

Model 2: Variable Importance using the same parameters but with a 70/30 split

```
Variable importance
     CHK_ACCT    HISTORY            OBS         DURATION    SAV_ACCT  AMOUNT     AGE
           34         13             10                9           8       5       4
   RETRAINING  GUARANTOR       USED_CAR PRESENT_RESIDENT NUM_CREDITS RADIOTV NEW_CAR
            2          2              2                2           2       2       1
OTHER_INSTALL EMPLOYMENT PROP_UNKN_NONE
            1          1              1
```

Model 3: Variable Importance using the same parameters but with a 80/20 split

```
variable importance
        CHK_ACCT         HISTORY        SAV_ACCT             OBS        DURATION          AMOUNT       GUARANTOR
              45              12              10               8               7               5               4
PRESENT_RESIDENT         RADIOTV     NUM_CREDITS      EMPLOYMENT  PROP_UNKN_NONE     REAL_ESTATE
               3               2               1               1               1               1
```

Analysis: The model almost has all the top three variables in common. The top variables according us are- Checking Account, History & Duration. I agree with this logic as customers with more money in the checking account and a good history with the bank is likely to be a ‘good’ credit holder.

Probability Threshold	Accuracy	Accuracy of Model2 with cost
.3	.738	.74
.4	.741	,773
.6	.741	.773
.7	.741	.665
.8	.701	.637

The accuracy of the model reduces with the increase in the prediction threshold value. This can also be validated by the given cost matrix where there is a loss of 500DM incorrectly predicted values.

The optimal threshold taken from the ROC curve for the model with cost matrix is 0.773 and the accuracy is 0.745.

		Predicted	
		Good	Bad
Actual	Good	0	100DM
	Bad	500DM	0

The theoretical threshold is 0.8 and the accuracy for this is 0.685, which is lower than what we computed in the answer above.

After comparing performance of the two new models,

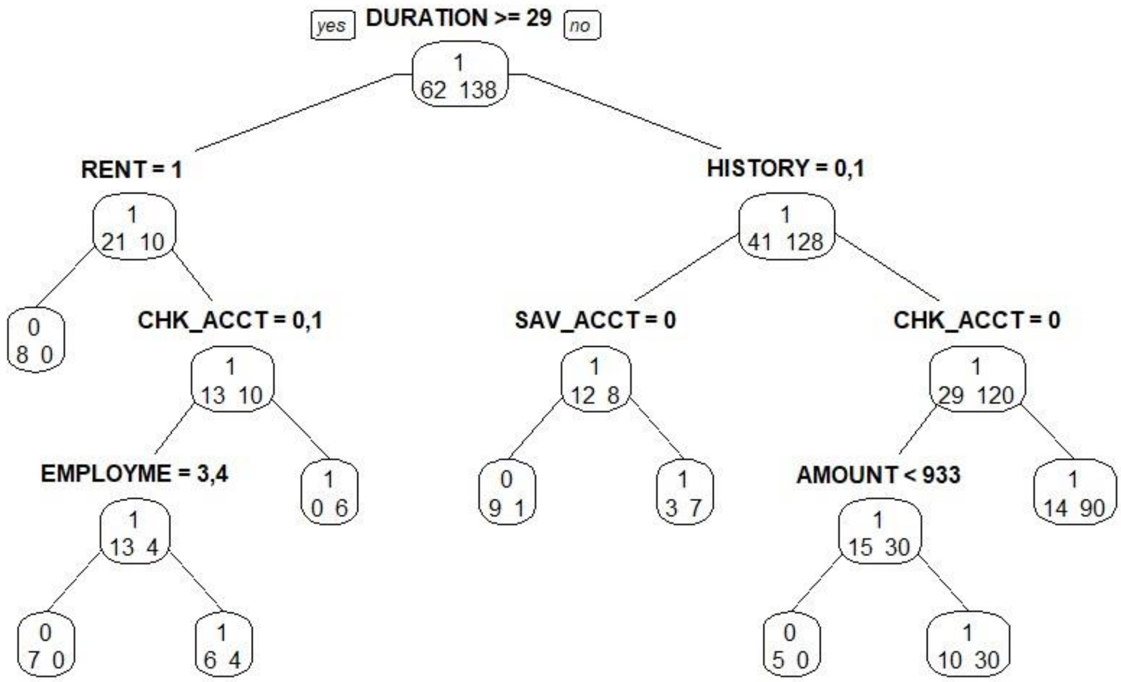
Model2 with a 70/30 split in Rptree category.
With cost matrix = .7667, Without cost matrix= .804

Tree2 with 50/50 split in C5.0 category.
With cost matrix = .7543, Without cost matrix= .801

The cost matrix depicts the cost for each of the combinations of predicted and actual categories. By default the costs of misclassification is set to 1, this reduces a model’s performance

The best Decision tree was obtained by considering various parameters like the split as Information, Cost matrix, Maximal depth of the tree as 10. The variable importance is as given below.

```
Variable importance
     AMOUNT     CHK_ACCT     SAV_ACCT     DURATION      HISTORY         RENT       EMPLOYMENT
         17           12           12           11           11            8                6
        OBS   RETRAINING      OWN_RES          AGE    TELEPHONE     USED_CAR PRESENT_RESIDENT
          6            4            4            4            2            2                1
```



The path followed by two pure leaf nodes is as given below:

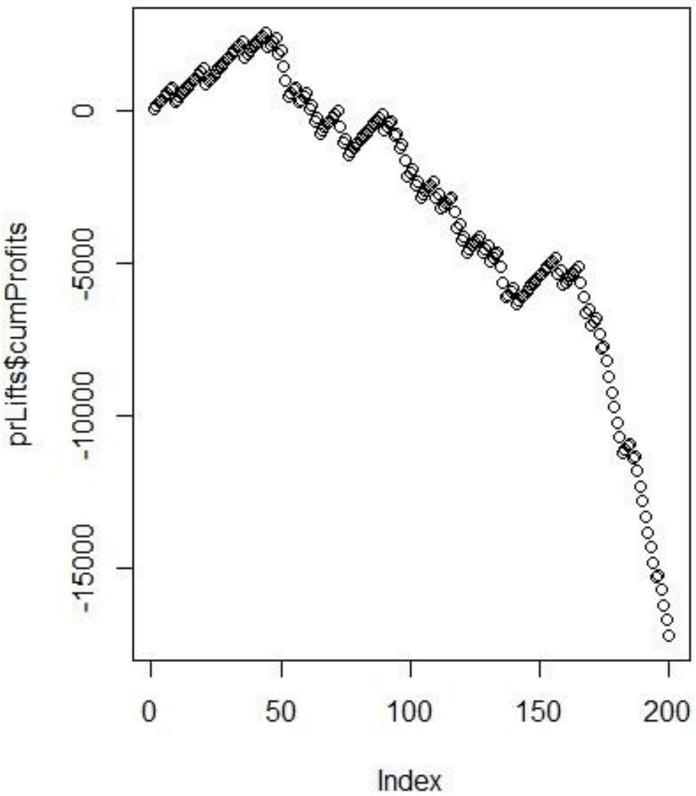
DURATION >>HISTORY>>CHK_ACCT>>AMOUNT – It has a size of 5 which means that there are 5 bad cases and 0 good cases. Probability of good case is 0 and bad case is 1.

DURATION>>RENT>>CHK_ACCT>>EMPLOYMENT-It has a size of 6 which means there are 0 bad cases and 6 good cases. Probability of good cases is 1 and bad case is 0.

The model is implemented based on the predicted probabilities. The data is sorted based on the predicted probability of "Good" Credit and then a cut-off probability is determined based on this list. The values above the cut-off probability are considered acceptable risk values.

Cost figures can be used to determine the cut-off probability. We can calculate the actual cost for each predicted probability of the validation case. Then the cumulative net cost is calculated for each case. We can find out the maximum net cost from the model and also the cut off value for the predicted probabilities for the future credit applicants.

A graph has been plotted based on the above guidelines.



The graph shows that we are getting a max profit of 5000DM with a probability of 0.868

Various Random Forest models were developed and tested. The table below summarizes the various parameters considered for selecting the best fit model.

Model	**No. of trees (mtry)**	**OOR error rate (%)**	**Training Set Response**	**Validation set Response**
Model1	5	24.01	129 0 0 350	11 2 51 147
Model2	7	25.05	129 0 0 350	13 3 49 146
Model3	9	26.51	129 0 0 350	17 3 45 146
Model4	16	26.51	129 0 0 350	22 7 40 142
Model5	18	26.3	129 0 0 350	22 5 40 144

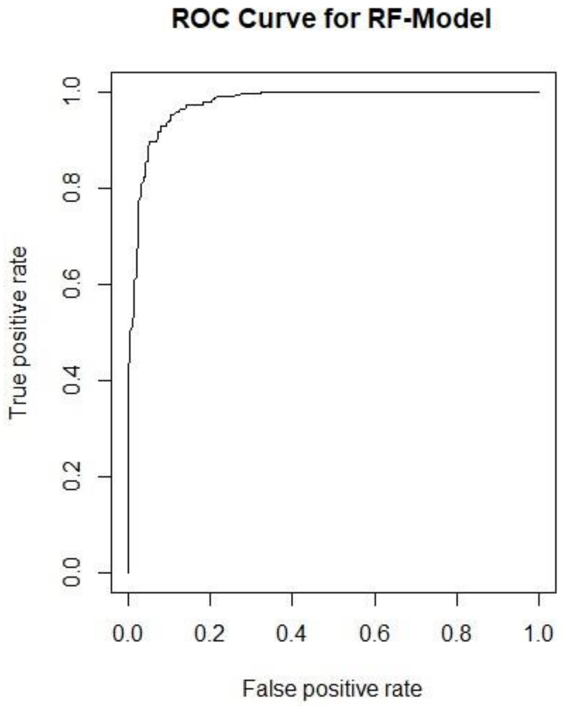
The table given below summarises the performance of each model described above. Based on the analysis, it can be observed that Model5 is the best fit model with an accuracy of 0.7800 and with mtry value of 18.

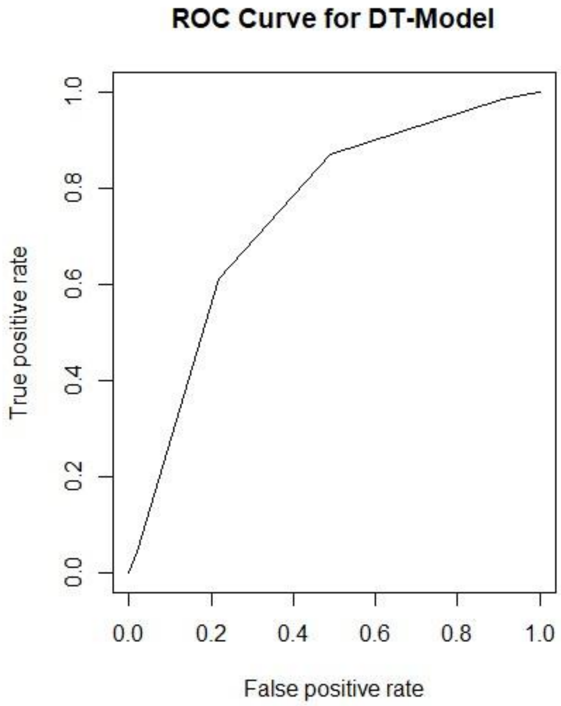
Model	**Accuracy of the Validation data (%)**	**Precision** **0 1**	**Recall** **0 1**	**Sensitivity** **0 1**
Model1	0.7488	0.8181 0.7238	0.1956 0.9797	0.3157 0.8326
Model2	0.7535	0.8333 0.7293	0.2173 0.9797	0.3448 0.8362
Model3	0.7722	0.8125 0.7441	0.2826 0.9696	0.4193 0.8421
Model4	0.7777	0.7894 0.7539	0.3260 0.9595	0.4615 0.8444
Model5	0.7800	0.7619 0.7580	0.3482 0.9494	0.4776 0.8430

<u>Comparing Random Forest Model with Decision Tree Model :</u>

The best fit Decision Tree model based on analysis is the Model2. Now comparing the performance of the best fit Decision Tree Model with the best fit Random Forest model obtained above, we notice that the Random Forest model has a better accuracy than the Decision tree model. The accuracy of the Random Forest model is 0.7772 where as the accuracy of the Decision Tree model is 0.6383.

The ROC plots have been plotted for the best fit Random Forest Model and the best fit Decision Tree model. It can be seen that the false positives increases with in Decision Tree model where as in the Random Forest model it remains constant.





Average profit graphs have been plotted for the best fit Random Forest model and the best fit Decision Tree model. The max-profit obtained for the Random Forest model is 3400 where as for the Decision Tree model it is 1200. This further proves that the Random Forest method is a more efficient method than the Decision Tree method.

```
maxProfit  scoreTst
 3400.000     0.748
```

```
   maxProfit      scoreTst
1200.0000000    0.8780488
```