DECISION TREE ANALYSIS

GERMAN CREDIT DATA

NISHANTH SINGAMSETTY

QUESTION 1

Explore the data: What is the proportion of "Good" to "Bad" cases? Are there any missing values – how do you handle these? Obtain descriptions of the predictor (independent) variables – mean, standard deviations, etc. for real-values attributes, frequencies of different category values. Examine variable plots. Do you notice 'bad' credit cases to be more prevalent in certain value-ranges of specific variables, and is this what one might expect (or is it more of a surprise)?What are certain interesting variables and relationships - explain why you think these are 'interesting'. From the data exploration, which variables do you think will be most relevant for the outcome of interest (and why)?

DATA SET

1000

Observations

30

Variables

7:3

Good: Bad Cases

German Credit data consists of data with 30 variables and 1000 instances of data describing the credit risk for 1000 applicants.

In this report, we have analyzed in detail the German credit data, which has led us to determine the associated risk involved in issuing a loan to new applicants. Using this data, we were able to infer the applicants with good credit vs bad credit, allowing us to differentiate the factors like socio-economic background of an individual, nature of their jobs, level of their education etc. that are responsible for classifying an applicant with good or bad credit.

The proportion of Good to Bad cases is 7:3, with 700 Good cases and 300 Bad cases of the total 1000 observations.

DATA CLEANING

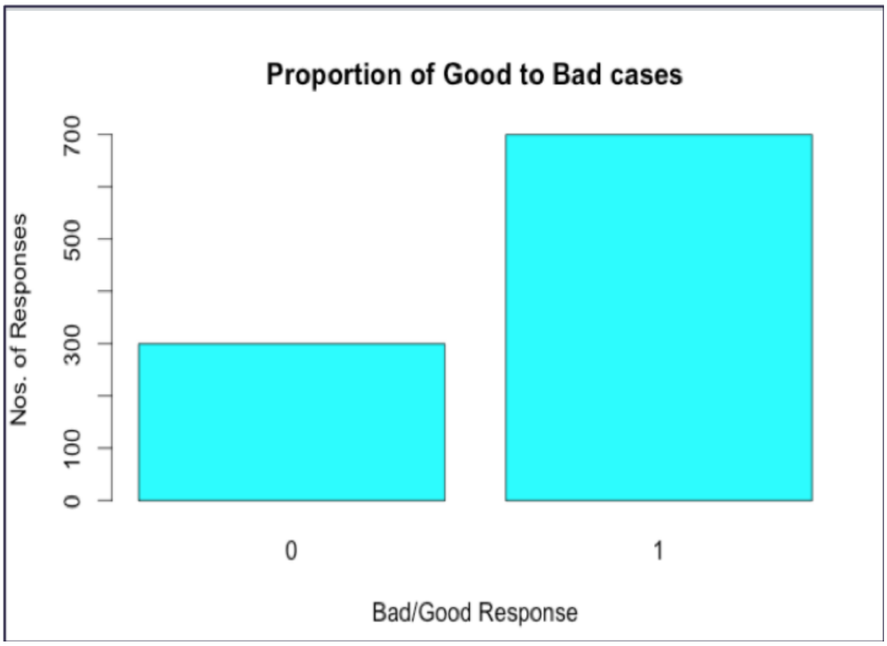
The proportion of Good to Bad cases is 7:3, with 700 Good cases and 300 Bad cases of the total 1000 observations.

The columns having NA values are New_Car, Used_Car, Furniture, Radio/Tv, Education, Retraining, Age and Personal Status

The missing values of New_Car, Used_Car, Furniture, Radio/Tv, Education, Retraining can be handled in the following ways –

1) Converting the NAs to 0
2) Combining the columns ("NEW_CAR","USED_CAR","FURNITURE","RADIO/TV","EDUCATION","RETRAINING") to one single column 'Purpose'
3) Removing these columns as they have maximum NA values.
4) Imputing the values depending on property of the column.

We have imputed age by the median value and NA values of Personal status are set to 'Other'

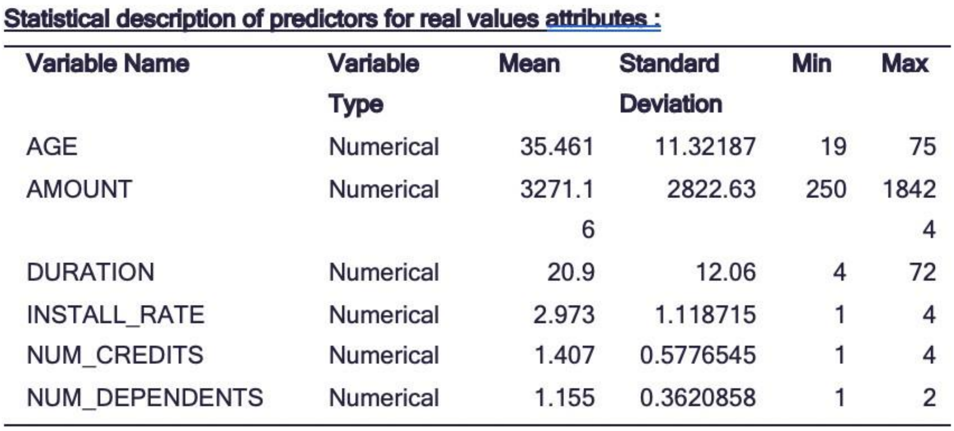


7:3

Good:Bad Cases

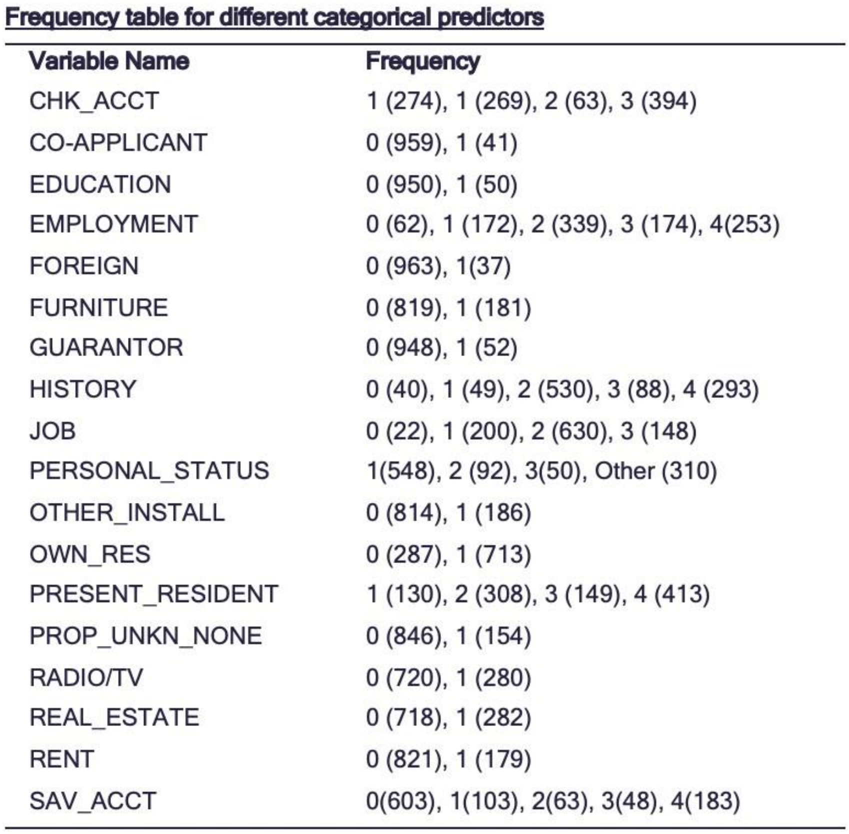
DESCRIPTIVE STATISTICS

Descriptive statistical information of the predictor (independent) variables:



Statistical description of predictors for real values attributes :

Variable Name	Variable Type	Mean	Standard Deviation	Min	Max
AGE	Numerical	35.461	11.32187	19	75
AMOUNT	Numerical	3271.16	2822.63	250	18424
DURATION	Numerical	20.9	12.06	4	72
INSTALL_RATE	Numerical	2.973	1.118715	1	4
NUM_CREDITS	Numerical	1.407	0.5776545	1	4
NUM_DEPENDENTS	Numerical	1.155	0.3620858	1	2



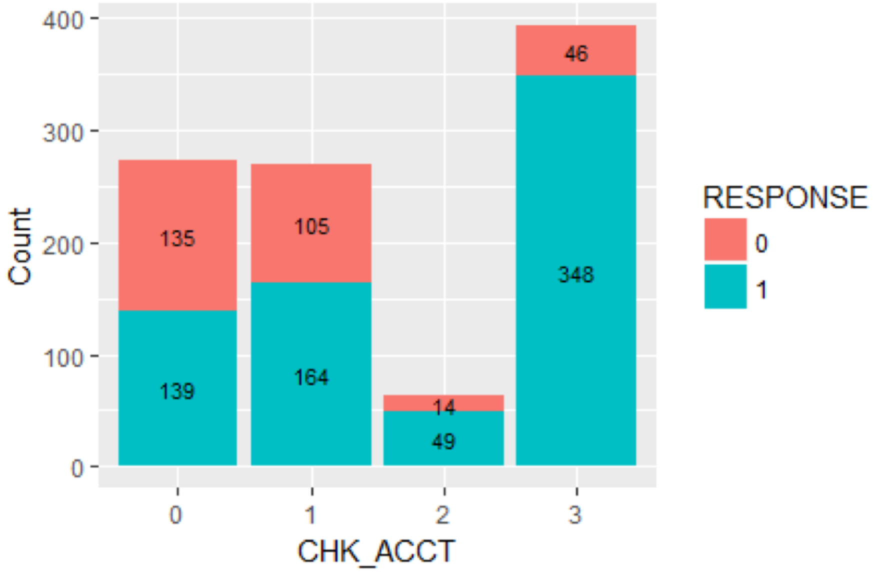
Frequency table for different categorical predictors

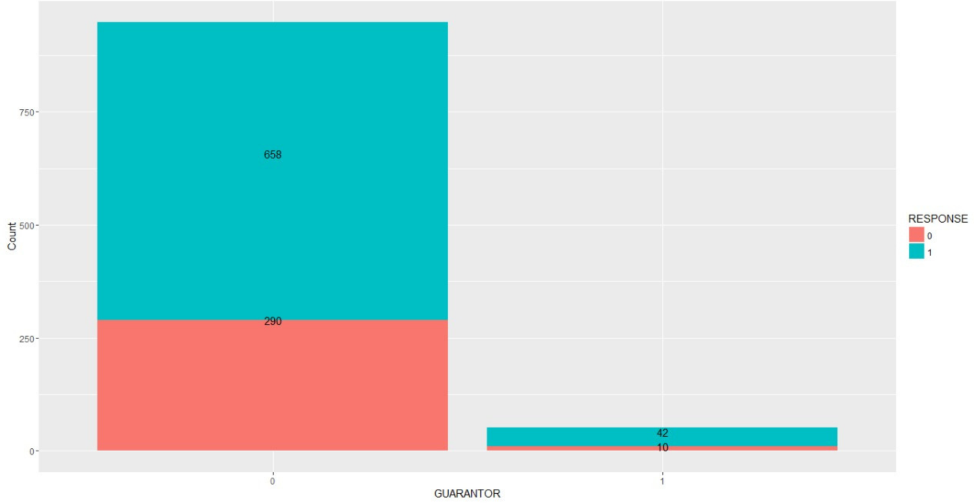
Variable Name	Frequency
CHK_ACCT	1 (274), 1 (269), 2 (63), 3 (394)
CO-APPLICANT	0 (959), 1 (41)
EDUCATION	0 (950), 1 (50)
EMPLOYMENT	0 (62), 1 (172), 2 (339), 3 (174), 4(253)
FOREIGN	0 (963), 1(37)
FURNITURE	0 (819), 1 (181)
GUARANTOR	0 (948), 1 (52)
HISTORY	0 (40), 1 (49), 2 (530), 3 (88), 4 (293)
JOB	0 (22), 1 (200), 2 (630), 3 (148)
PERSONAL_STATUS	1(548), 2 (92), 3(50), Other (310)
OTHER_INSTALL	0 (814), 1 (186)
OWN_RES	0 (287), 1 (713)
PRESENT_RESIDENT	1 (130), 2 (308), 3 (149), 4 (413)
PROP_UNKN_NONE	0 (846), 1 (154)
RADIO/TV	0 (720), 1 (280)
REAL_ESTATE	0 (718), 1 (282)
RENT	0 (821), 1 (179)
SAV_ACCT	0(603), 1(103), 2(63), 3(48), 4(183)

EXAMINATION OF VARIABLES

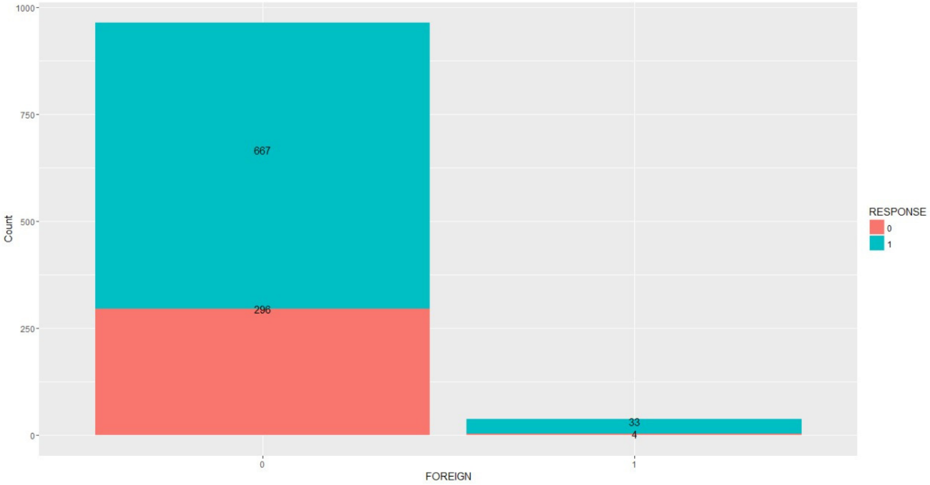
By observing the plots for all the independent variables, we decided to use Checking Account, Guarantor, Foreign, Savings A/C, History, Employment as they are good predictor (independent) variables for our model.
We examined these variables with the response variable.

Checking Account
As we can see in the plot below, out of 700 good cases, 348 cases are the ones that do not have a checking account (categorical value 3) have their loan approved. Thus, this variable can be a good predictor. Almost 88.3% of applicants having no checking account are good creditors, which is a surprise. Checking account status 2 has the least percentage, meaning applicants with more than 200 DM have the lowest percentage, which is surprising.





Guarantor
Though the applicants having a guarantor is very less there more number are good creditors. 69.4% of applicants having no guarantor are good creditors too, which is a surprise

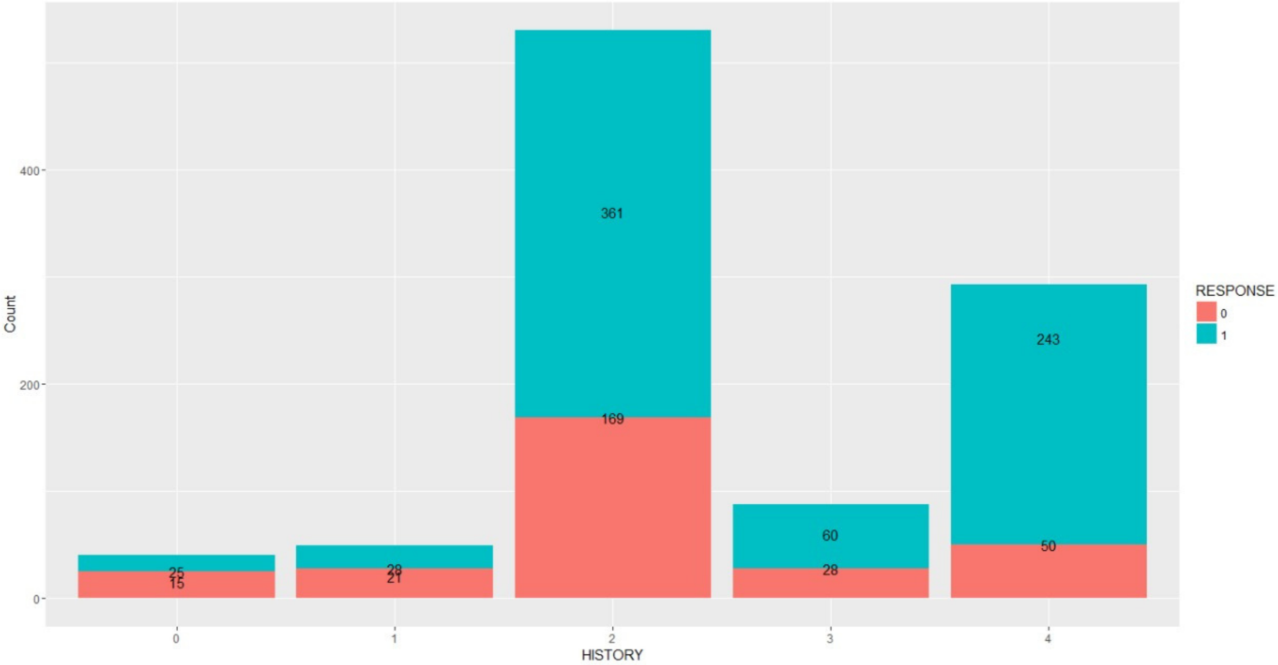


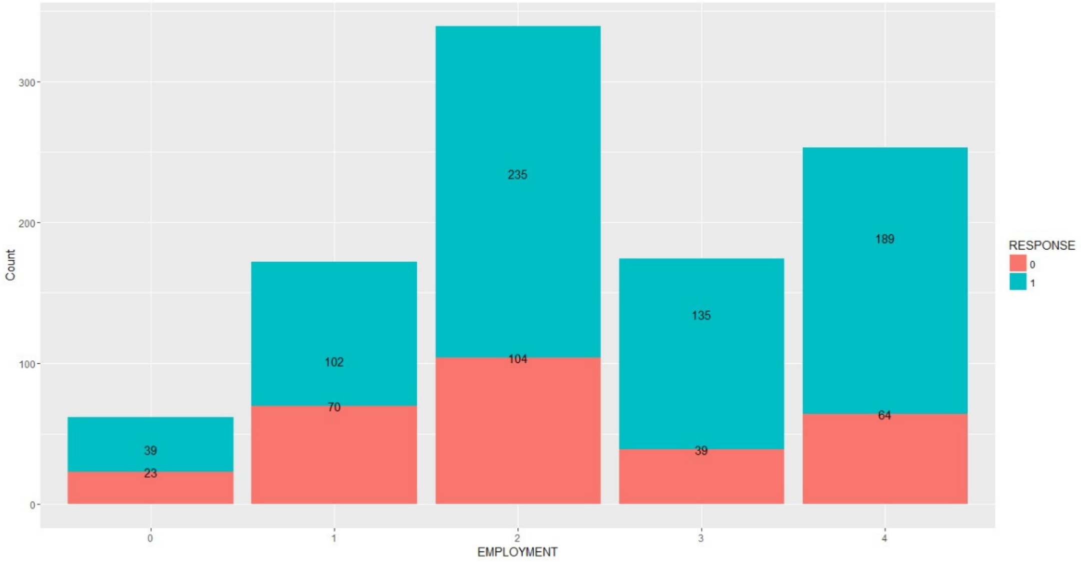
Foreign:

Local people are likely to default more compared to foreigners. But then, the data sample for foreigners is very low, so we cannot confirm this on such a small data set.

History:

Almost 82.9% of applicants having critical accounts are good creditors. This also comes as a surprise.





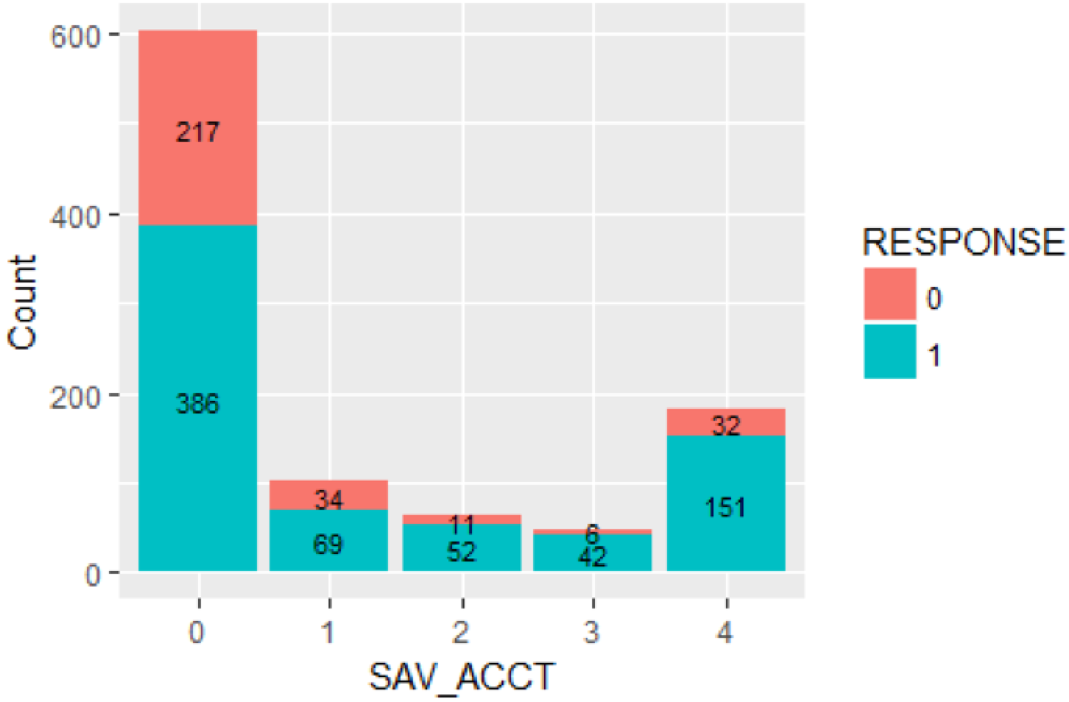
Employment :

Applicants that have less than 4 years of work exp have 76.1% of good creditors as compared to that of 7 years work exp

Savings A/C

Below is the plot showing the frequency of the response variable with SAV_ACCT variable. For the category 0 (average balance less than 100 DM), there is good proportion of good as well as bad cases, which can help us in getting a good model. Most applicants having average balance in their savings account between 500 - 1000 DM and more than 1000 DM are good creditors.

Also, 82.5% of applicants having no savings account are good creditors, and this comes as a surprise



INTERESTING VARIABLES

From the data exploration, We find the below variables will be most relevant for the outcome of interest.
CHK_ACCT and SAV_ACCT – Liquidity has to be shown in these accounts in order to get the loan approved, since it shows that a person can afford the payments. Customers who do not have a savings account have a good credit response

Employment – Employment in an organization enhances the chances of getting a loan approved since it shows that the applicant has stable income to support his payments.

C**redit History** – This is the response or reaction of an account holder in a bank according to the previous credits. Critical account holders history have a good credit response

Guarantor - Interesting because the customers who do not have a Guarantor have a good credit response

Foreign - Foreign is interesting because of its decent number of 33 customers who are foreigners having good credit response

Employment - People who have an employment of less than 4 yrs have a better credit response than the ones having 7 years experience

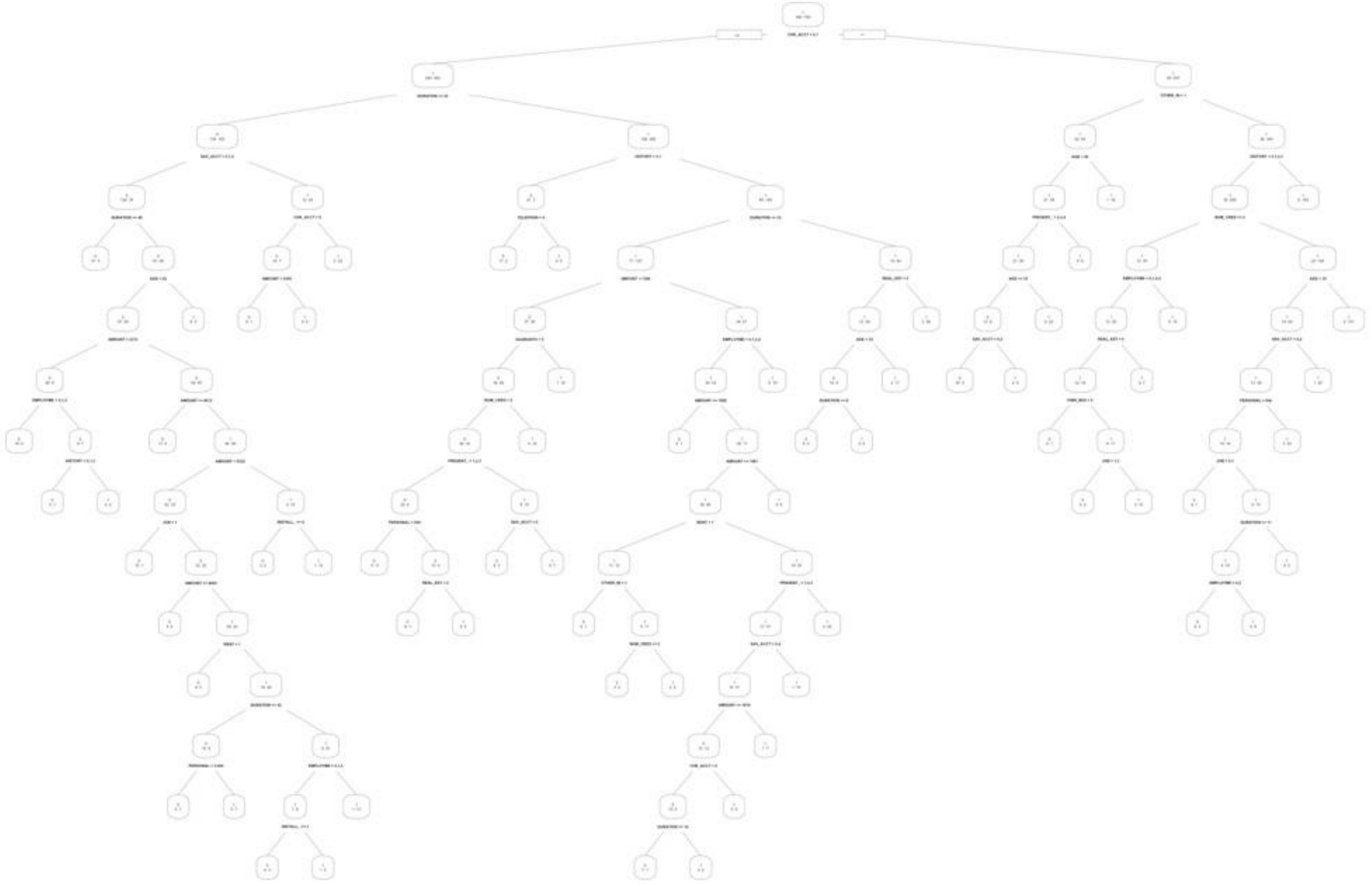
QUESTION 2

a) We will first focus on a descriptive model – i.e. assume we are not interested in prediction. (a) Develop a decision tree on the full data (using the rpart package).
What decision tree node parameters do you use to get a good model. Explain the parameters you use (see rpart.control – https://www.rdocumentation.org/packages/rpart/versions/4.1-13/topics/rpart.control)
(b) Which variables are important to differentiate "good" from "bad" cases – and how do you determine these? Does this match your expectations (from your response in Question 1)?
(c) What levels of accuracy/error are obtained? What is the accuracy on the "good" and "bad" cases? Obtain and interpret the lift chart. Do you think this is a reliable (robust?) description, and why.

Our initial decision tree was a very simple model based on the Information Gain criterion and a minimum split of 15. Below is the decision tree used for the same.

Amongst the 4 models we created this tree had the highest accuracy of 86.6%

But, with this model we obtained a decision tree which is very big and it is difficult to read the decision tree efficiently. This makes the tree unreliable. We know that for a model to be reliable, it should be possible to validate it by performing statistical test on it. Also, it should be resistant to error i.e. it should perform well if its assumptions are violated by the true model from which the data were generated. In order to get a good model, below are the node parameters that result in a more optimized tree which has a better performance. To make our decision tree more robust and readable, we split the data into training and testing data

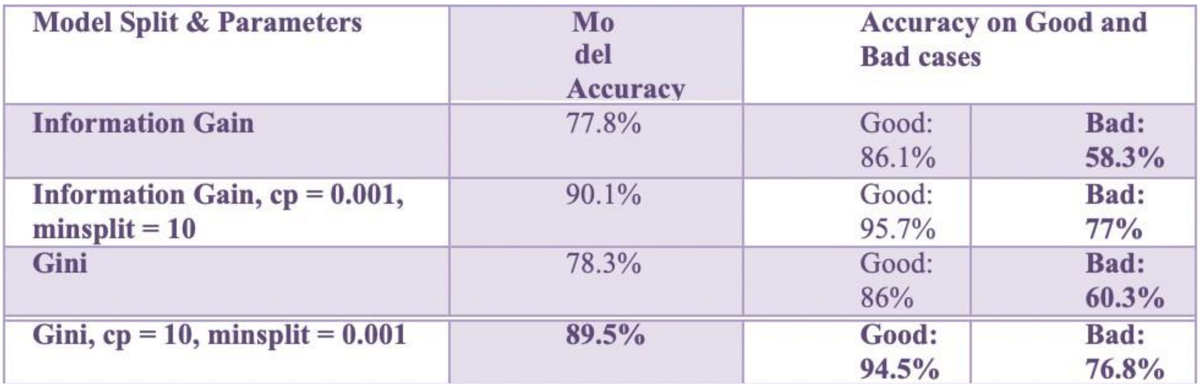


(b) Which variables are important to differentiate "good" from "bad" cases – and how do you determine these? Does this match your expectations (from your response in Question 1)?

The important variables can be determined by analyzing the summary of our models. We see that with different models, the important variables coming on the top part of the tree are nearly the same. The 4 most important variables are CHK_ACCT, AMOUNT, DURATION and HISTORY.
From question 1 we see that the variables which seem most relevant are - age, guarantor, employment, checking account, savings account, history. We can see that Checking account and history are present in both variable importance.

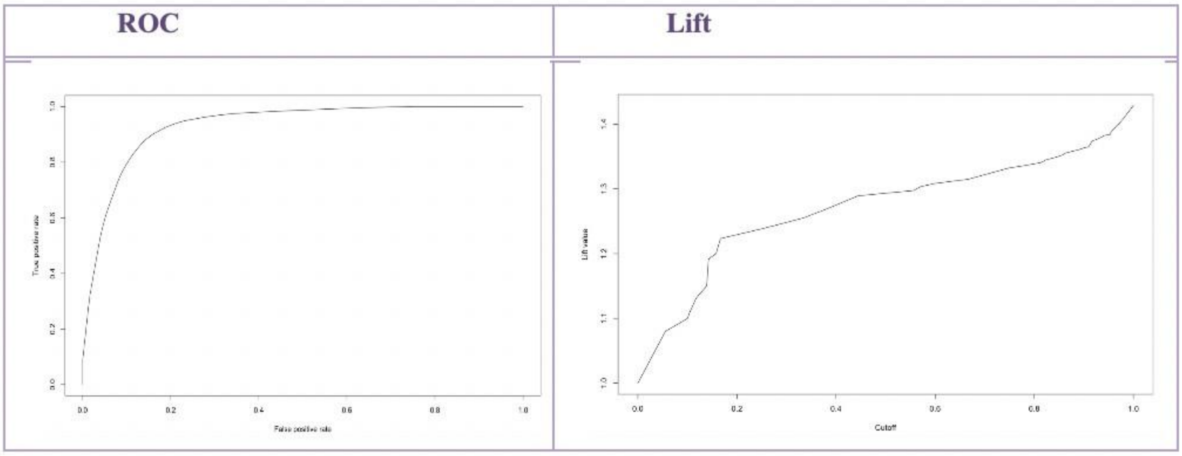
(c) What levels of accuracy/error are obtained? What is the accuracy on the "good" and "bad" cases? The levels of accuracy on the good and bad cases differ with models.

The best accuracy is obtained with the model having Information gain as the split criteria, cp as 0.001 and minsplit as 10 (94.5% of good creditors and 76.8% of bad creditors were correctly predicted by this model).
The model had a overall accuracy of 90.1%



Model Split & Parameters	Model Accuracy	Accuracy on Good and Bad cases	
Information Gain	77.8%	Good: 86.1%	**Bad: 58.3%**
Information Gain, cp = 0.001, minsplit = 10	90.1%	Good: 95.7%	**Bad: 77%**
Gini	78.3%	Good: 86%	**Bad: 60.3%**
Gini, cp = 10, minsplit = 0.001	**89.5%**	**Good: 94.5%**	**Bad: 76.8%**

Obtain and interpret the lift chart.



Interpretation: From the plot we can see that since the percentage area under the curve and the value for AUC is high, it is a good model.

Do you think this is a reliable (robust?) description, and why.
This method doesn't give a reliable model as we cannot estimate how this model will work on unseen data. Hence, for being absolutely certain, we need to run this model on test data.

QUESTION 3

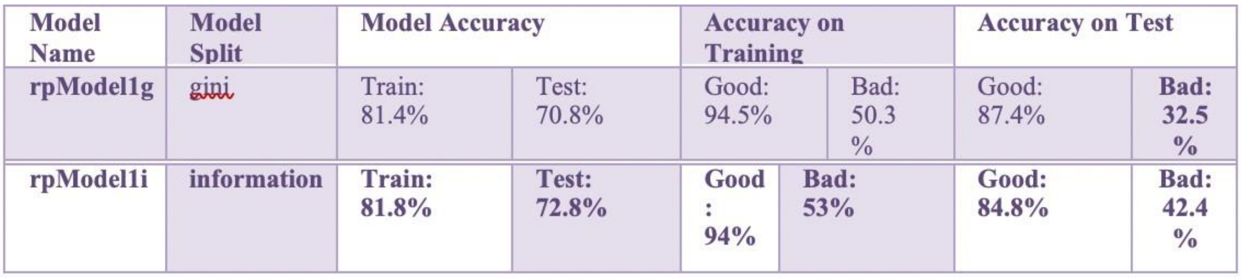
We next consider developing a model for prediction. For this, we should divide the data into Training and Validation sets. Consider a partition of the data into 50% for Training and 50% for Test

a) Develop decision trees using the rpart package. What model performance do you obtain? Consider performance based on overall accuracy/error and on the 'good' and 'bad' credit cases – explain which performance measures, like recall, precision, sensitivity, etc. you use and why. Also consider lift, ROC and AUC.
In developing the models above, change decision tree options as you find reasonable (for example, complexity parameter (cp), the minimum number of cases for split and at a leaf node, the split criteria, etc.) - explain which parameters you experiment with and why. Report on if and how different parameters affect performance. Which decision tree parameter values do you find to be useful for developing a good model.
Describe the pruning method used here. How do you examine the effect of different values of cp, and how do you select the best pruned tree?
Explain how you use different performance measures to determine your best model.
(b) Consider another type of decision tree – C5.0 – experiment with the parameters till you get a 'good' model. Summarize the parameters and performance you obtain.
Also develop a set of rules from the decision tree, and compare performance. Does performance differ across different types of decision tree learners? Compare models using accuracy, sensitivity, precision, recall, etc (as you find reasonable – you answer to questions (a) above should clarify which performance measures you use and why). Also compare performance on lift, ROC curves and AUC.
How do the models obtained from these decision tree learners differ?
(c) Decision tree models are referred to as 'unstable' – in the sense that small differences in training data can give very different models. Examine the models and performance for different samples of the training/test data (by changing the random seed). Do you find your models to be unstable – explain.
(d) Which variables are important for separating 'Good' from 'Bad' credit? Determine variable importance from the different 'best' trees. Are there similarities, differences?
Explain how variable importance is determined (for rpart and C5.0 models)
(d) Consider partitions of the data into 70% for Training and 30% for Test, and 80% for Training and 20% for Test and report on model and performance comparisons (for the decision tree learners considered above).
In the earlier question, you had determined a set of decision tree parameters to work well. Do the same parameters give 'best' models across the 50-50, 70-30, 80-20 training-test splits? Are there similarities among the different models ....in, say, the upper part of the tree, and in variable importance – and what does this indicate?
Is there any specific model you would prefer for implementation?

We next consider developing a model for prediction. For this, we should divide the data into Training and Validation sets. Consider a partition of the data into 50% for Training and 50% for Test.

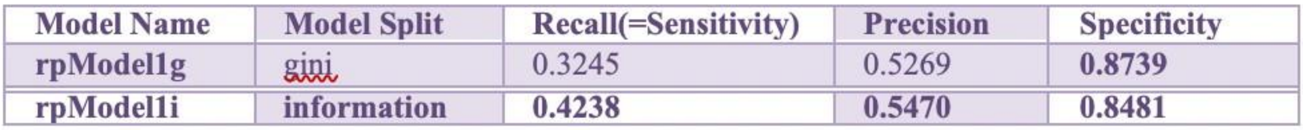
(a) Develop decision trees using the rpart package. What model performance do you obtain? Consider performance based on overall accuracy/error and on the 'good' and 'bad' credit cases

The best accuracy is of Model rpModel1i, which is made on information gain as the split. The accuracy of the model on test data is 72.8%.



Model Name	Model Split	Model Accuracy		Accuracy on Training		Accuracy on Test	
rpModel1g	gini	Train: 81.4%	Test: 70.8%	Good: 94.5%	Bad: 50.3 %	Good: 87.4%	**Bad: 32.5 %**
rpModel1i	**information**	**Train: 81.8%**	**Test: 72.8%**	**Good : 94%**	**Bad: 53%**	**Good: 84.8%**	**Bad: 42.4 %**

Explain which performance measures, like recall, precision, sensitivity, etc. you use and why.
The various performance measures for the above models are:

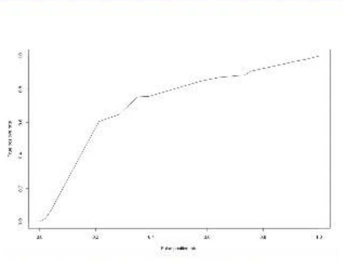
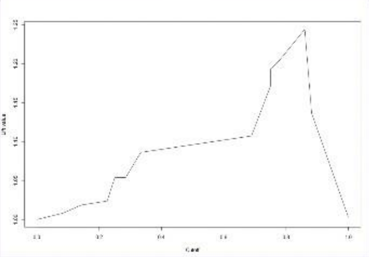
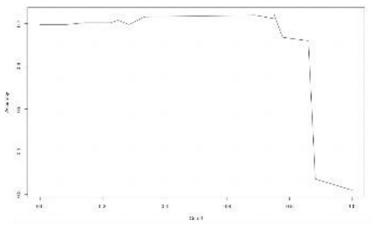


Model Name	Model Split	Recall(=Sensitivity)	Precision	Specificity
rpModel1g	gini	0.3245	0.5269	**0.8739**
rpModel1i	**information**	**0.4238**	**0.5470**	**0.8481**

Recall gives the proportion of bad cases correctly predicted by our model, with respect to the total number of actual bad cases. The model using information gain as the split can correctly predict 42% of the bad cases.
Precision measures the percentage of accurately predicted bad cases out of the total number of predictions made by our model. Hence, this model can accurately predict 55% of bad cases out of the total predictions it makes.
Specificity measures the proportion of good cases correctly predicted. For the model on split as gini, it shows that it can correctly predict 87% of the total good cases.

Also consider lift, ROC and AUC.

Since our best model is the one built on information gain as the split, the lift, ROC and AUC is as below:

ROC	Lift	AUC
 Cost ROC = 0.2857143		 AUC value: 0.7193116

Is the model reliable (why or why not)?

Yes, the model is reliable. From the plots and the high AUC value above, we can say that the model is reliableand robust.

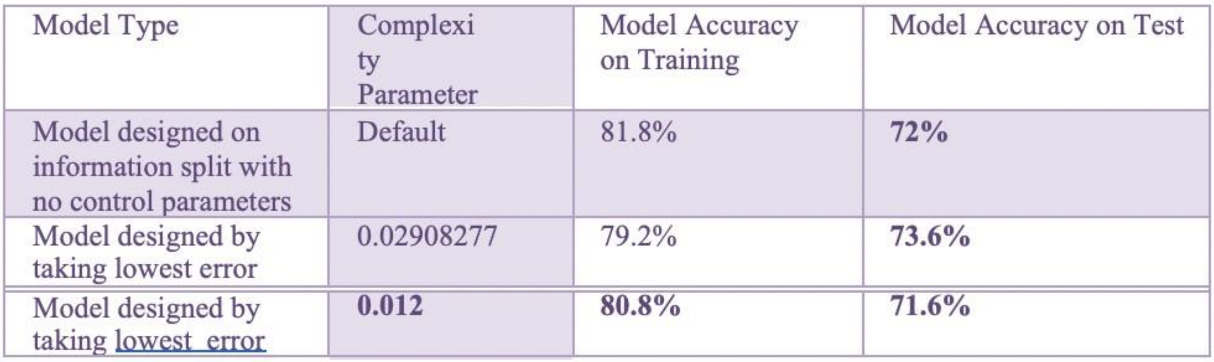
In developing the models above, change decision tree options as you find reasonable (for example, complexity parameter (cp), the minimum number of cases for split and at a leaf node, the split criteria, etc.) - explain which parameters you experiment with and why. Report on if and how different parameters affect performance.

Model Name	Complexity Parameter	minsplit	Model Accuracy on Training	Model Accuracy on Test
rpMode_1	0.001	10	89.6	**64.4%**
rpMode_2	0.1	10	70.2%	**69.8%**
rpMode_3	**0.001**	**5**	**94.4%**	**64.6%**

After changing cp value drastically to 0.1 from 0.001, we notice in the confusion matrix that the bad creditors are all predicted as good creditors. The accuracy for the model predicted on training data decreases by around 19% and the accuracy for the model predicted on test data increases by 4%.

After keeping cp value as 0.001 and changing minsplit to 5, we observe that the accuracy for the model predicted on training data increased to 94.4% from 89%, and the accuracy for the model predicted on testing data changed to 64.6% from 64.4%.

Describe the pruning method used here. How do you examine the effect of different values of cp, and how do you select the best pruned tree?



Model Type	Complexi ty Parameter	Model Accuracy on Training	Model Accuracy on Test
Model designed on information split with no control parameters	Default	81.8%	**72%**
Model designed by taking lowest error	0.02908277	79.2%	**73.6%**
Model designed by taking lowest error	**0.012**	**80.8%**	**71.6%**

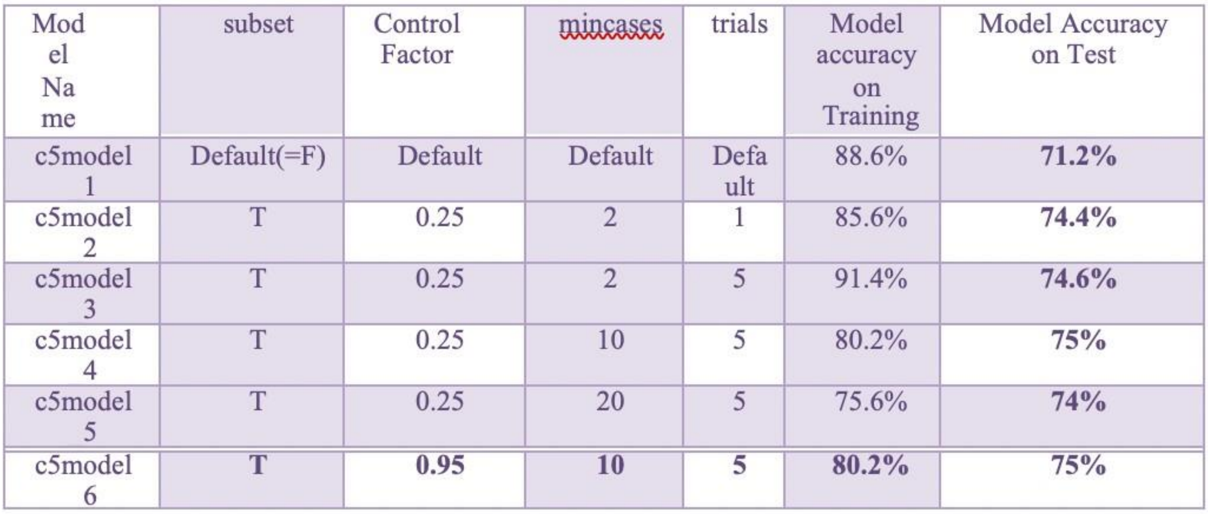
We first find the base model accuracy for models predicted on training data and test data. After that we find the optimal cp value where the cross-validation error (xerror) and where relative error (rel error) have minimum value. From that we find two optimal cp values : 0. 02908277and 0.012. By keeping these optimal values of cp, the accuracy for the training data decreased but for the testing increased.

We find optimal cp value to be 0.02908277, with minsplit as 10 and split criteria as information gain. These values decide the best pruned tree.

Which decision tree parameter values do you find to be useful for developing a good model?

We find that optimal values of cp and minsplit are most useful for developing a good model.

(b) Consider another type of decision tree – C5.0 – experiment with the parameters till you get a 'good' model. Summarize the parameters and performance you obtain.

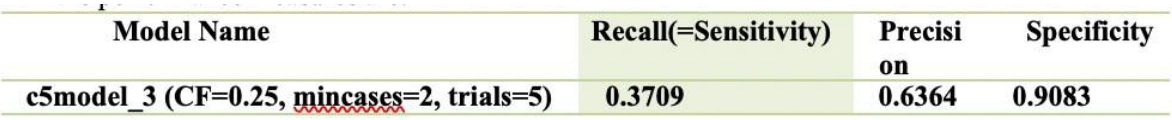


Model Name	subset	Control Factor	mincases	trials	Model accuracy on Training	Model Accuracy on Test
c5model 1	Default(=F)	Default	Default	Default	88.6%	**71.2%**
c5model 2	T	0.25	2	1	85.6%	**74.4%**
c5model 3	T	0.25	2	5	91.4%	**74.6%**
c5model 4	T	0.25	10	5	80.2%	**75%**
c5model 5	T	0.25	20	5	75.6%	**74%**
c5model 6	**T**	**0.95**	**10**	**5**	**80.2%**	**75%**

Using discrete predictors for splits (subsets) doesn't seem to have much effect on the accuracy of the model on both training and test data. Increasing the Confidence Factor (which gives the measure of the threshold of the allowed error in data while pruning) from 0.25 to 0.95 increases the accuracy of the model on the training data by 2.5% and decreases on the test data by 1%. Increase in the value of mincases results in decrease of the accuracy, both on training and test data. Also, increasing the trials (which controls the boosting) results in better accuracy of the model on both training and test sets.
Therefore, the model built with CF = 0.25, mincases = 2, and trials = 5 gives the best accuracy of prediction.

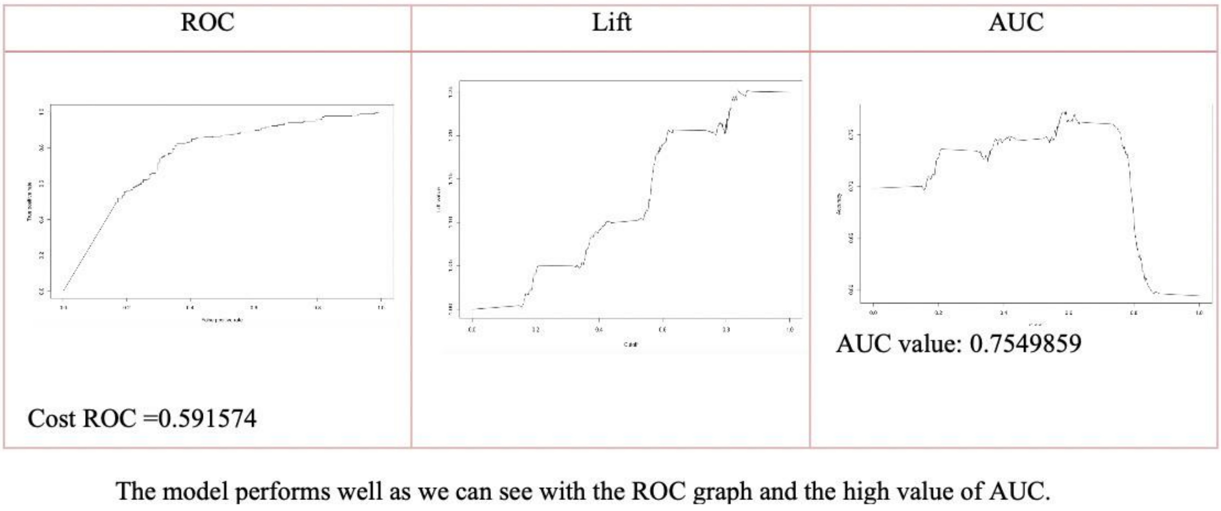
Does performance differ across different types of decision tree learners?
As seen from the above 6 models, performance does differ across different decision tree learners.
Compare models using accuracy, sensitivity, precision, recall, etc (as you find reasonable – you answer to Questions (a) above should clarify which performance measures you use and why).

Since c5model_3 (CF=0.25, mincases=2, trials=5) gives the best accuracy, this is our best model and the performance measures are:



Model Name	Recall(=Sensitivity)	Precision	Specificity
c5model_3 (CF=0.25, mincases=2, trials=5)	**0.3709**	**0.6364**	**0.9083**

(Also compare performance on lift, ROC curves and AUC. For Model – c5model_3 (CF=0.25, mincases=2, trials=5):



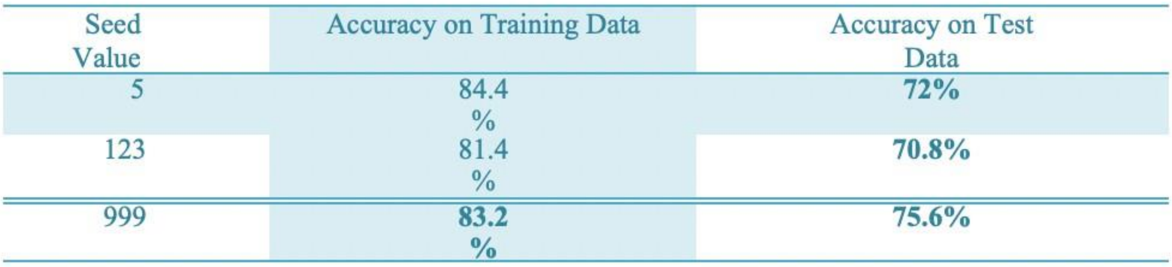
ROC	Lift	AUC
Cost ROC =0.591574		AUC value: 0.7549859

The model performs well as we can see with the ROC graph and the high value of AUC.

The model performs well as we can see with the ROC graph and the high value of AUC.

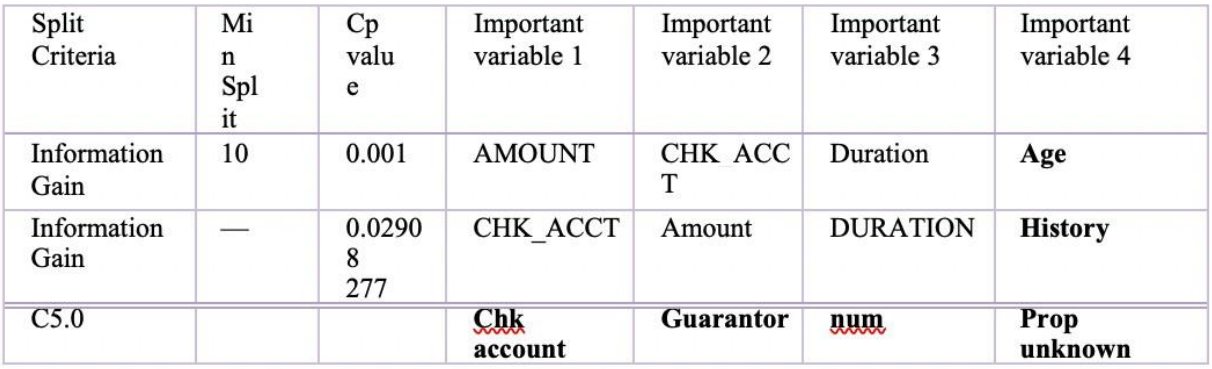
3 (c) Decision tree models are referred to as 'unstable' – in the sense that small differences in training data can give very different models. Examine the models and performance for different samples of the training/test data (by changing the random seed). Do you find your models to be unstable -- explain?

We check our base model on 3 different seed values – 5, 123 and 999
We notice that on different seed values, the accuracy on Training data differ as much as 3% change, as compared to the accuracy on Test data with a change of approximate 3%



Seed Value	Accuracy on Training Data	Accuracy on Test Data
5	84.4 %	**72%**
123	81.4 %	**70.8%**
999	**83.2 %**	**75.6%**

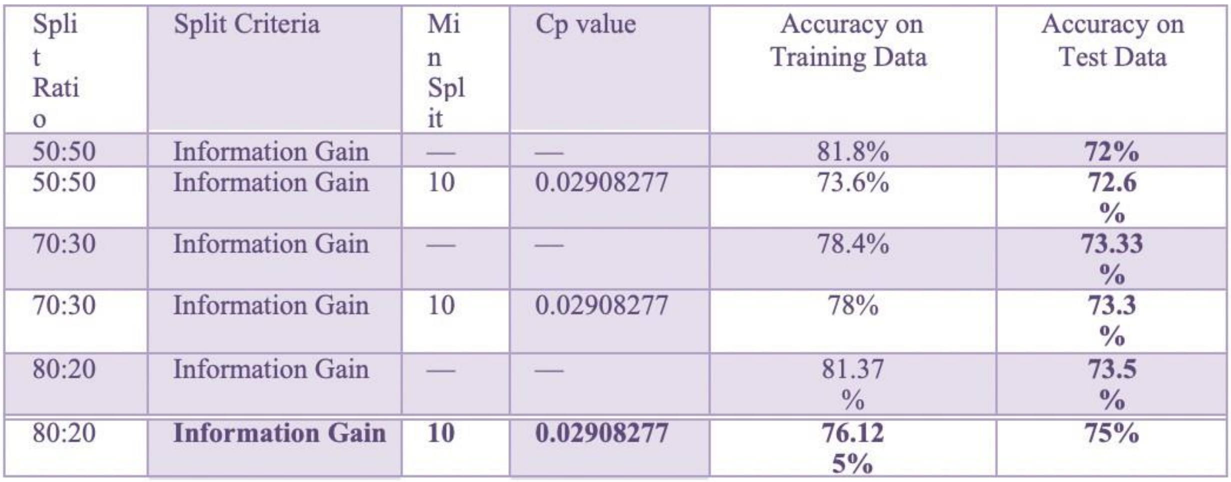
3 (d) Which variables are important for separating 'Good' from 'Bad' credit? Determine variable importance from the different 'best' trees. Are there similarities, differences? From different 'best' trees we determine the following variable importance –



Split Criteria	Min Split	Cp value	Important variable 1	Important variable 2	Important variable 3	Important variable 4
Information Gain	10	0.001	AMOUNT	CHK_ACCT	Duration	**Age**
Information Gain	—	0.02908277	CHK_ACCT	Amount	DURATION	**History**
C5.0			**Chk account**	**Guarantor**	**num**	**Prop unknown**

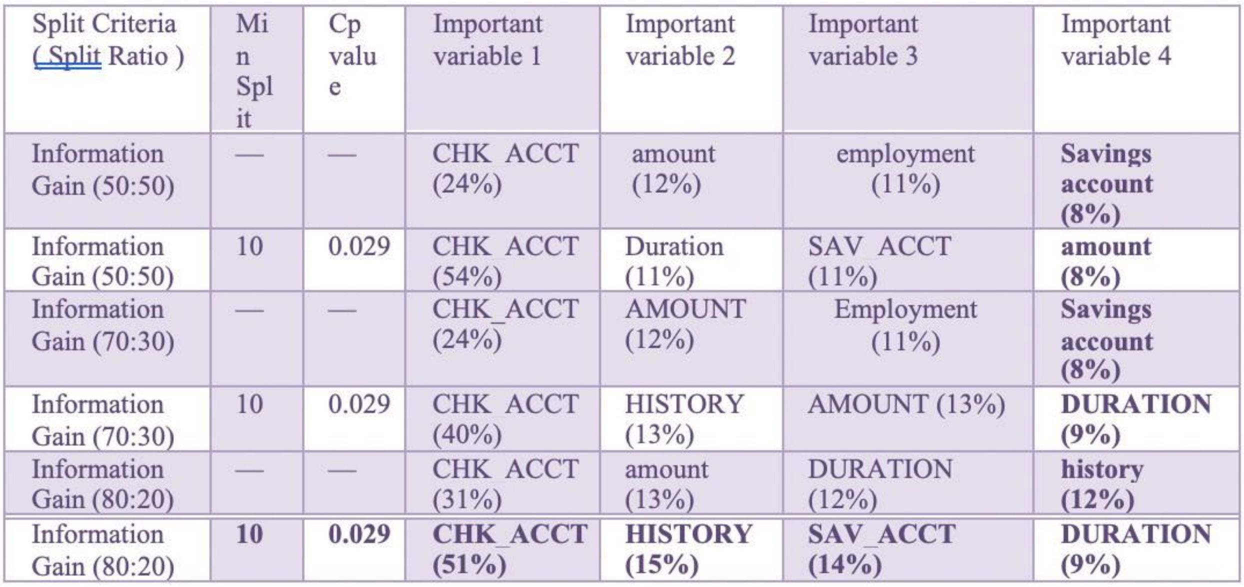
We see that with different 'Best' models, the important variables coming on the top part of the tree are nearly the same. With each model the percentage of observations change. Seeing one model we can match our expectations with other models. The top 3 most important variables are CHK_ACCT, AMOUNT and HISTORY, followed with AGE, DURATION and SAV_ACCT.

3 (e) Consider partitions of the data into 70% for Training and 30% for Test, and 80% for Training and 20% for Test and report on model and performance comparisons (for the decision tree learners considered above).
In the earlier question, you had determined a set of decision tree parameters to work well. Do the same parameters give 'best' models across the 50-50, 70-30, 80-20 training-test splits? Are there similarities among the different models ….in, say, the upper part of the tree – and what does this indicate?
Is there any specific model you would prefer for implementation?



Split Ratio	Split Criteria	Min Split	Cp value	Accuracy on Training Data	Accuracy on Test Data
50:50	Information Gain	—	—	81.8%	**72%**
50:50	Information Gain	10	0.02908277	73.6%	**72.6%**
70:30	Information Gain	—	—	78.4%	**73.33%**
70:30	Information Gain	10	0.02908277	78%	**73.3%**
80:20	Information Gain	—	—	81.37%	**73.5%**
80:20	**Information Gain**	**10**	**0.02908277**	**76.125%**	**75%**

The performance of the trees on the test & train data increases the accuracy when we increase the split ratio from 50:50 to 80:20



Split Criteria (Split Ratio)	Min Split	Cp value	Important variable 1	Important variable 2	Important variable 3	Important variable 4
Information Gain (50:50)	—	—	CHK_ACCT (24%)	amount (12%)	employment (11%)	**Savings account (8%)**
Information Gain (50:50)	10	0.029	CHK_ACCT (54%)	Duration (11%)	SAV_ACCT (11%)	**amount (8%)**
Information Gain (70:30)	—	—	CHK_ACCT (24%)	AMOUNT (12%)	Employment (11%)	**Savings account (8%)**
Information Gain (70:30)	10	0.029	CHK_ACCT (40%)	HISTORY (13%)	AMOUNT (13%)	**DURATION (9%)**
Information Gain (80:20)	—	—	CHK_ACCT (31%)	amount (13%)	DURATION (12%)	**history (12%)**
Information Gain (80:20)	**10**	**0.029**	**CHK_ACCT (51%)**	**HISTORY (15%)**	**SAV_ACCT (14%)**	**DURATION (9%)**

As we increase the partitioning parameter from 0.5 to 0.7 and finally 0.8 we see that some variables have gained in observation percentage. The variables seen in all the models are CHK_ACCT, HISTORY, DURATION,SAV_ACCT and AMOUNT. We also observe that, this increment in training data observations leads to an increase in the number of observations percentage in the variable on 1st position of the model.

QUESTION 4

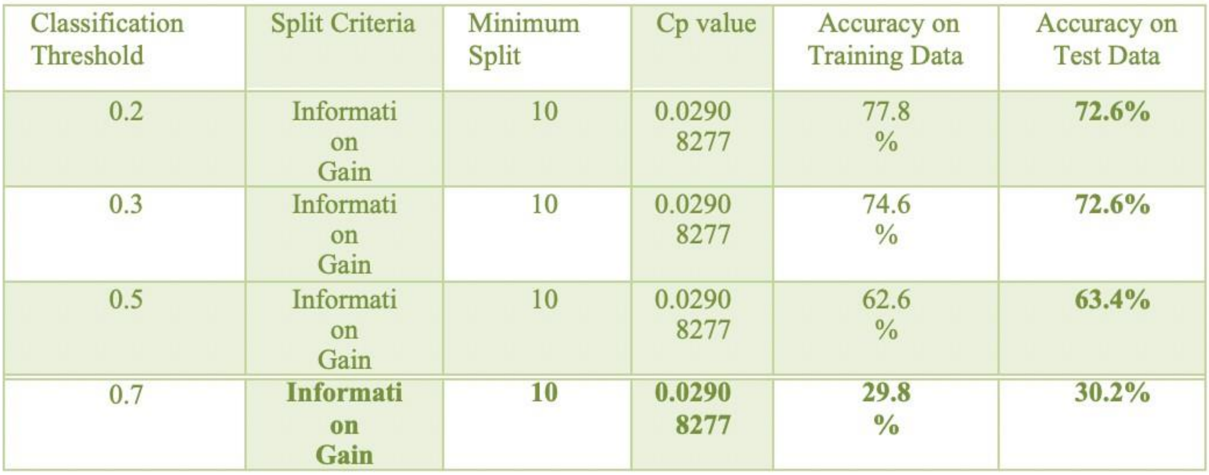
Consider the net profit (on average) of credit decisions as:
Accept applicant decision for an Actual "Good" case: 100DM, and
Accept applicant decision for an Actual "Bad" case: -500DM
This information can be used to determine the following costs for misclassification: (table)

a)Use the misclassification costs to assess performance of a chosen model from Q 3 above. Compare model performance. Examine how different cutoff values for classification threshold make a difference. Use the ROC curve to choose a classification threshold which you think will be better than the default 0.5. What is the best performance you find?

(b) Calculate and apply the 'theoretical' threshold and assess performance – what do you notice, and how does this relate to the answer from (a) above.

(c) Use misclassification costs in building the tree models (rpart and C5.0) – are the trees here different than ones obtained earlier? Compare performance of these two new models with those obtained earlier (in part 3a, b above).

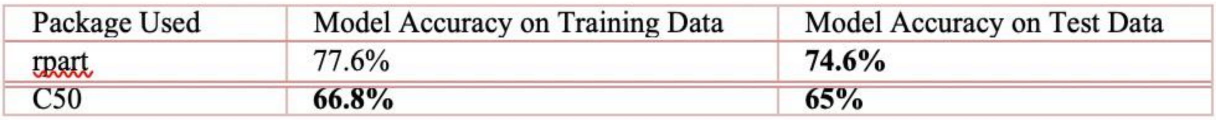
We see that accuracy is maximum when the cthresh is set to 0.2.
As the cthresh value increases from 0.2 to 0.5 the accuracy drops slightly from 77.8% to 62.6% in Training data and from 72.6% to 63.4% in testing data and as the cthresh value increases from 0.5 to 0.7, we see a sharp drop in accuracy from 64.8% to 29.8% in Training data and from 62.6% to 30.2% in testing data.



Classification Threshold	Split Criteria	Minimum Split	Cp value	Accuracy on Training Data	Accuracy on Test Data
0.2	Information Gain	10	0.02908277	77.8%	**72.6%**
0.3	Information Gain	10	0.02908277	74.6%	**72.6%**
0.5	Information Gain	10	0.02908277	62.6%	**63.4%**
0.7	**Information Gain**	**10**	**0.02908277**	**29.8%**	**30.2%**

(b) Calculate and apply the 'theoretical' threshold and assess performance – what do you notice, and how does this relate to the answer from (a) above.
The theoretical threshold for good cases was found out to be 0.1666667.
We can also see that the classification threshold 0.2 gives us the best accuracy.

(c) Use misclassification costs to develop the tree models (rpart and C5.0) – are the trees here different than ones obtained earlier? Compare performance of these two new models with those obtained earlier (in part 3a, b above).

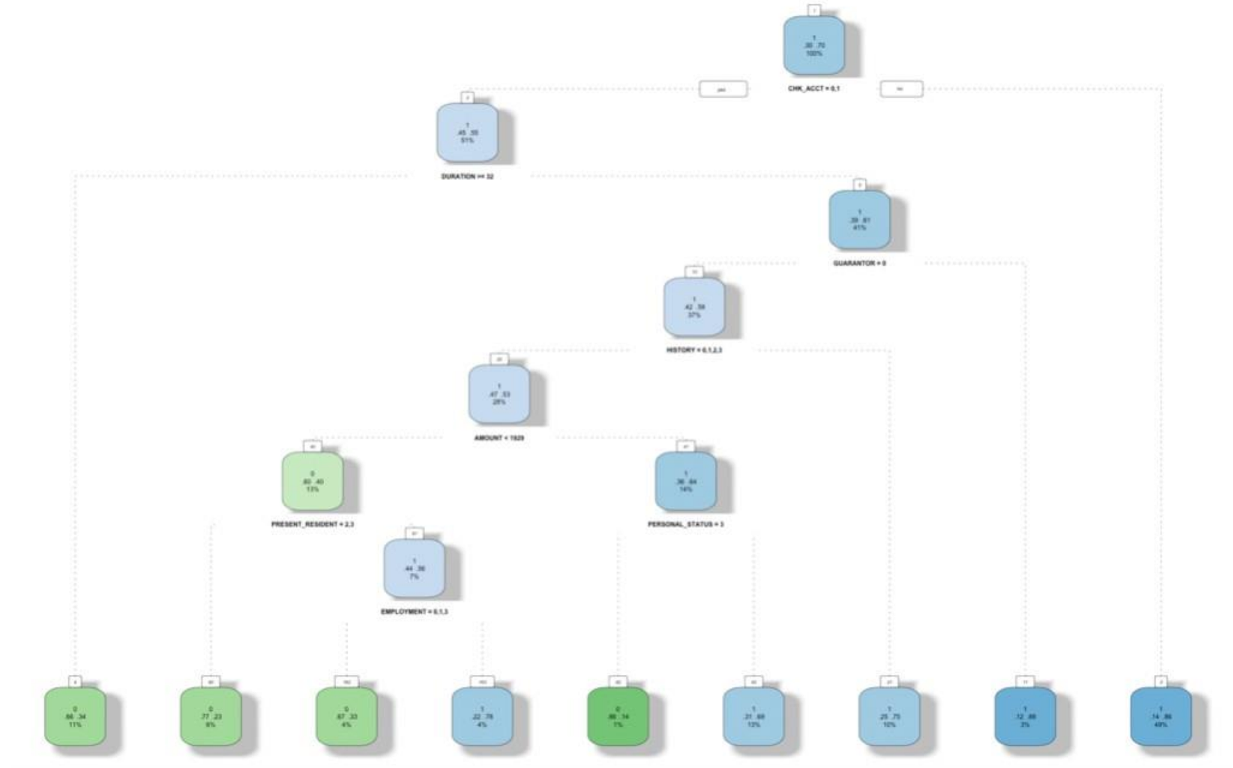


Package Used	Model Accuracy on Training Data	Model Accuracy on Test Data
rpart	77.6%	**74.6%**
C50	**66.8%**	**65%**

QUESTION 5

Let’s examine your ‘best’ decision tree model obtained. What is the tree depth? And how many nodes does it have? What are the important variables for classifying “Good’ vs ‘Bad’ credit? Identify two relatively pure leaf nodes. What are the ‘probabilities for ‘Good’ and ‘Bad’ in these nodes? Calculate the smoothed values for these ‘probabilities’ for “Good” and “Bad” cases in these nodes – calculate the Laplace smoothing and m-estimate smoothing values.

Best decision tree with split criteria as Information Gain, minsplit as 10 and cp value as 0.0290



Rattle 2019-Feb-16 16:26:46 talishbarmare

Nodes: 17
Depth: 6

Split Criteria (Split Ratio)	Min Split	Cp value	Important variable 1	Important variable 2	Important variable 3	Important variable 4
Information Gain (50:50)	**10**	**0.0290**	**CHK_ACCT (54%)**	**Duration (11%)**	**SAV_ACCT (11%)**	**Amount (8%)**

Two relatively pure nodes:
Node 3: Predicted as 1. Probability ratio of good vs bad cases = 0.86:0.14
Node 4: Predicted as 0. Probability ratio of good vs bad cases = 0.34:0.66

QUESTION 6

The predicted probabilities can be used to determine how the model may be implemented. We can sort the data from high to low on predicted probability of "good" credit risk. Then, going down the cases from high to low probabilities, one may be able to determine an appropriate cutoff probability – values above this can be considered acceptable credit risk. The use of cost figures given above can help in thisanalysis.

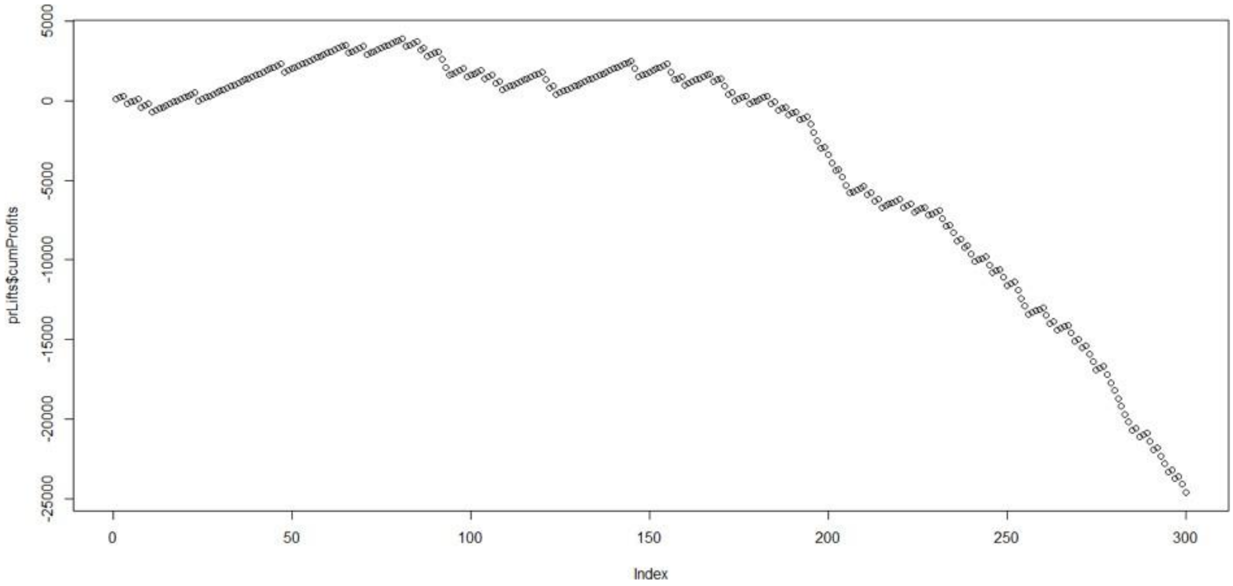
For this, first sort the validation data on predicted probability. Then, for each validation case, calculate the actual cost/benefit of extending credit. Add a separate column for the cumulative net cost/benefit.

How far into the validation data would you go to get maximum net benefit? In using this model to score future credit applicants, what cutoff value for predicted probability would you recommend? Provide appropriate performance values to back up your recommendation.

We used probability method to determine how the model may be implemented. We predicted the data based on the probability on " good" credit risk. We then sorted in descending order i.e,from high to low probabilities.

Please find the plot below

As observed from the cumulative Cost /Benefit chart, the maximum net profit is $3900 and the cut off value is 0.867 at the 88th observation.
Based on the graph using the guidelines given in the question, assuming a 'profit' value for correctly predicting a 'good' case is 100, and a 'cost' for mistakes is -500, with this we calculated the profits and the cumulative profits.we obtained that the maximum net benefit is achieved at the 88th record of the validation data after which the cumulative cost starts decreasing.In order to score future credit applicants, we would recommend a cut-off value of 159th record of the validation data for predicted probability.



QUESTION 7

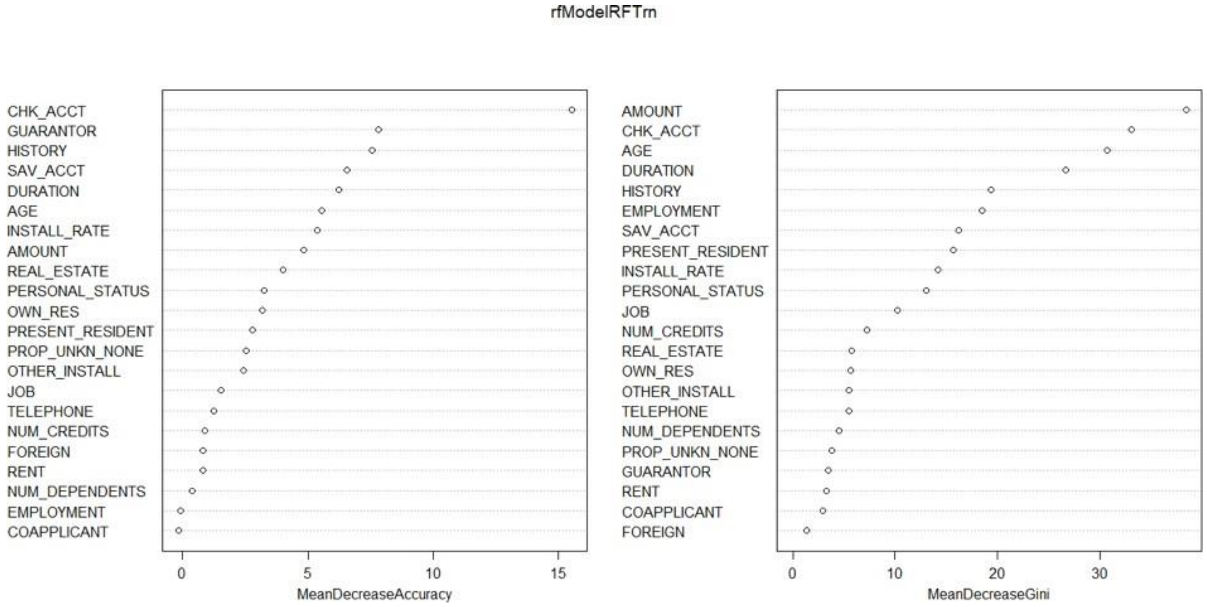
Develop a random forest model (using a 70:30 training: test data split). What random forest parameters do you try out, and what performance do you obtain? Compare the performance of the best random forest and best decision tree models – show a ROC plot to help compare models, and also the maximum net benefit (as in question 6).

Below is a plot of all the variables used to build the Random Forest and their importance in the model.

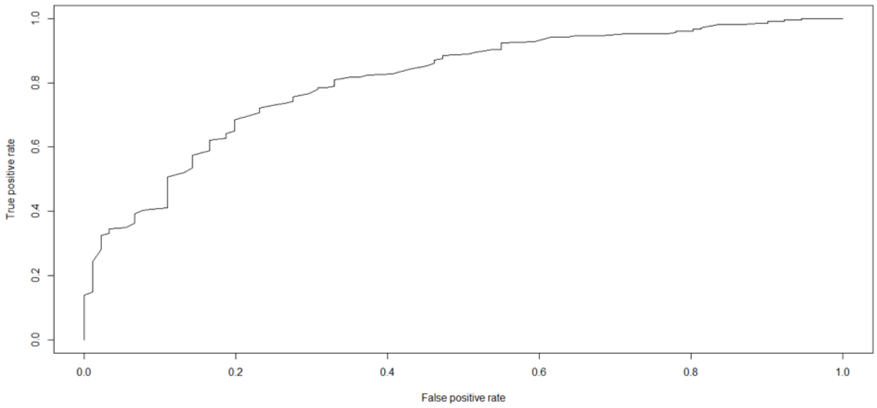
Random Forest on 70-30 data split Variable Importance:
The accuracy test is to see how worse the model performs without each variable, so a high decrease in accuracy would be expected for very predictive variables.
Again, it tests to see the result if each variable is taken out and a high score means the variable was important.

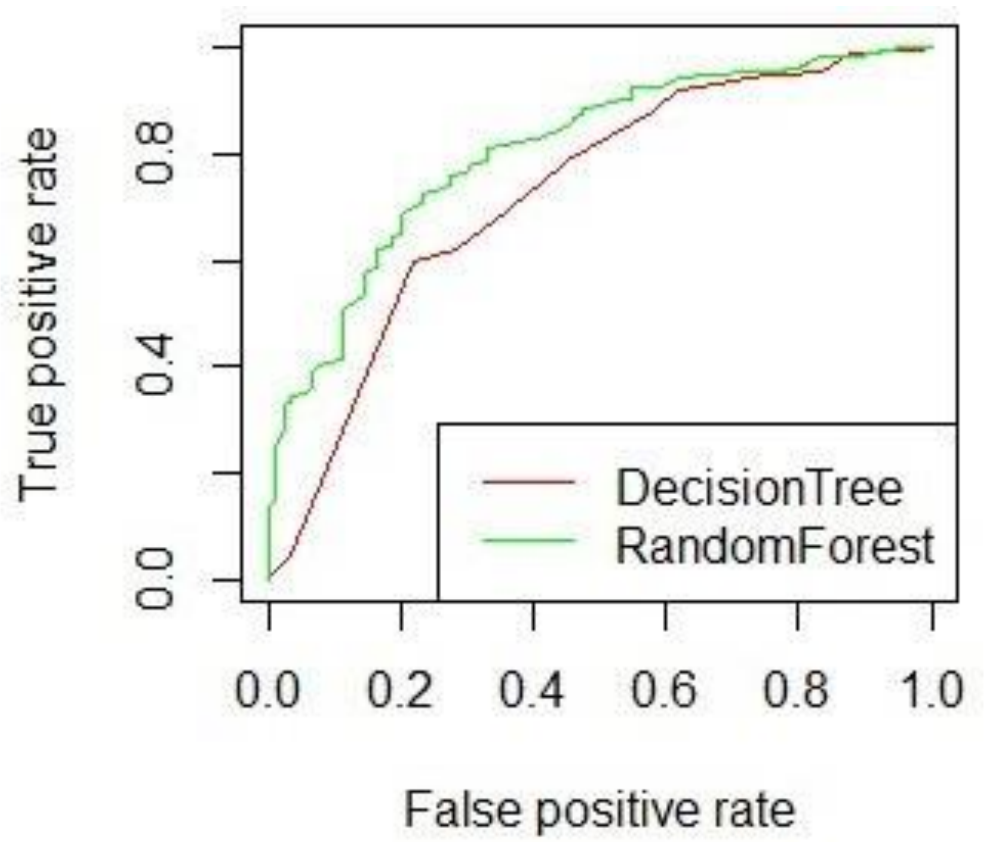
The second measure in the table below gives the total decrease in node impurities from splitting on the variable, averaged over all trees.

The rfModel graph below gives us a single score per variable aggregated across the whole forest. The MeanDecreaseGini from the table below, shows how much heterogeneity in each node is decreased after every split. So, CHK_ACCT and Guarantor seem to have contributed the most to obtaining such splits, showing that they can be important.



ROC curve for Random Forest:





Comparing the performance of the best Random Forest vs Best Decision tree using ROC Plot
As seen above clearly, the Random Forest Model is better than the Decision Tree Model.