**IDS 572**

**DATA MINING**

**ASSIGNMENT 1 :**

GERMAN CREDIT RISK ANALYSIS

TEAM MEMBERS:

Satya Julakanti (667258378)

Raja Amlan (651045470)

Prashant Chaudhary (678226947)

**INTRODUCTION:**

The German Credit dataset has data on 1000 past credit applicants, described by 30 variables where each applicant is rated as “Good” or “Bad” credit (encoded as 1 and 0 respectively in the Response variable). New applicants for credit can be classified as “Good” or “Bad” credit on basis of these 30 variables. We will develop a credit scoring rule and obtain a decision tree based model to determine if new applicants present a good or bad credit risk. The original data has been transformed to ease our analysis.

**ASSIGNMENT QUESTIONS:**

***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 (why ‘interesting’)? From the data exploration, which variables do you think will be most relevant for the outcome of interest, and why?

The Proportion of “Good” to “Bad” cases is 7:3. Among the 1000 observations, 70% is good credit cases and remaining 30% depicts bad credit cases.

A graphical representation of good and bad credit cases is shown below.

Missing Values:

7 out of 31 columns had missing values. 6 columns (NEW\_CAR, USED\_CAR, RETRAINING, EDUCATION, FURNITURE, RADIO/TV) contains binomial values 0&1. The missing data in these columns are replaced with 0. The missing values in the 7th Column “Age” is replaced by the median Age by class, ie the missing value for age in Good credit class is replaced by the median age corresponding to good credit and same was done for missing values in age column corresponding to Bad credit

Description of Predictor Variables:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Variable** | **Mean** | **Median** | **Std.Deviation** | **Max** | **Min** | **Range** |
| DURATION | 20.90 | 18.0 | 12.06 | 72 | 4 | 68 |
| AMOUNT | $3,271.16 | $2,319.50 | $2,822.63 | $18,424 | $250 | $18,174 |
| INSTALL\_RATE | 2.97 | 3.0 | 1.12 | 4 | 1 | 3 |
| AGE | 35.45 | 33.0 | 11.33 | 75 | 19 | 56 |
| NUM\_CREDITS | 1.41 | 1.0 | 0.58 | 4 | 1 | 3 |
| NUM\_DEPENDENTS | 1.16 | 1.0 | 0.36 | 2 | 1 | 1 |

Categorical Variables:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Checking Account** | **# of bad credit** | **% of bad credit** | **# of good credit** | **% of good credit** | **Ratio of bad to good** |
| < 0 DM | 135 | 45% | 139 | 20% | 0.97 |
| 0 < ...< 200 DM | 105 | 35% | 164 | 23% | 0.64 |
| => 200 DM | 14 | 5% | 49 | 7% | 0.29 |
| no checking account | 46 | 15% | 348 | 50% | 0.13 |

* Checking account status of Type 0 and 1 have more people with bad credit score.
* Checking account status of Type 3 have highest percentage of people with good credit score.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Savings Account** | **# of bad credit** | **% of bad credit** | **# of good credit** | **% of good credit** | **Ratio of bad to good** |
| < 100 DM | 217 | 72% | 386 | 55% | 0.56 |
| 100<= ... < 500 DM | 34 | 11% | 69 | 10% | 0.49 |
| 500<= ... < 1000 DM | 11 | 4% | 52 | 7% | 0.21 |
| =>1000 DM | 6 | 2% | 42 | 6% | 0.14 |
| unknown/ no savings account | 32 | 11% | 151 | 22% | 0.21 |

* Savings account status in “Category 1” have highest ratio of bad to good credit score than any other category.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Employment** | **# of bad credit** | **% of bad credit** | **# of good credit** | **% of good credit** | **Ratio of bad to good** |
| unemployed | 23 | 8% | 39 | 6% | 0.59 |
| < 1 year | 70 | 23% | 102 | 15% | 0.69 |
| 1 <= ... < 4 years | 104 | 35% | 235 | 34% | 0.44 |
| 4 <=... < 7 years | 39 | 13% | 135 | 19% | 0.29 |
| >= 7 years | 64 | 21% | 189 | 27% | 0.34 |

* Here is an interesting fact, the number of good credits for unemployed people is higher than the number of bad credits which was unexpected. Moreover the ratio of number of bad credits to good credits for applicants with an experience of 7 years or more is higher for applicants with an experience between 4-7 years

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Present Resident** | **# of bad credit** | **% of bad credit** | **# of good credit** | **% of good credit** | **Ratio of bad to good** |
| <= 1 year | 36 | 12% | 94 | 13% | 0.38 |
| 1<…<=2 years | 97 | 32% | 211 | 30% | 0.46 |
| 2<…<=3 years | 43 | 14% | 106 | 15% | 0.41 |
| >4years | 124 | 41% | 289 | 41% | 0.43 |

* People who have been resident from past 4 years have bad credit history. But we cannot any relationship of credit rate from the present resident variable as even the people who are resident for 1- 2 years also have a bad credit history.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Cateogy** | **Present Resident** | **# of bad credit** | **% of bad credit** | **# of good credit** | **% of good credit** | **Ratio of bad to good** |
| 0 | unemployed/ unskilled - non-resident | 7 | 2% | 15 | 2% | 0.47 |
| 1 | unskilled - resident | 56 | 19% | 144 | 21% | 0.39 |
| 2 | skilled employee / official | 186 | 62% | 444 | 63% | 0.42 |
| 3 | management/ self-employed/highly qualified employee/ officer | 51 | 17% | 97 | 14% | 0.53 |

* We again found an interesting fact. Category 2 which consists skilled employee/officials are the once with maximum % of bad credit history, however its Bad to good credit ratio is not greater as the % of good credit score is similar to the % of bad credit score.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Cateogru** | **Present Resident** | **# of bad credit** | **% of bad credit** | **# of good credit** | **% of good credit** | **Ratio of bad to good** |
| 0 | no credits taken | 25 | 8% | 15 | 2% | 1.67 |
| 1 | all credits at this bank paid back duly | 28 | 9% | 21 | 3% | 1.33 |
| 2 | existing credits paid back duly till now | 169 | 56% | 361 | 52% | 0.47 |
| 3 | delay in paying off in the past | 28 | 9% | 60 | 9% | 0.47 |
| 4 | critical account | 50 | 17% | 243 | 35% | 0.21 |

* Category 1 has the largest ratio of Bad to good credit which is a surprise as it corresponds to the people who's all credits at this bank are paid back duly. Similarly, we can see that % of bad credit is maximum in Category 2 even it consists the people who have paid dues which are till date.

Binomial Variables:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Binary variables** | **# of bad credit** | **% of bad credit** | **# of good credit** | **% of good credit** | **Ratio of bad to good** |
| NEW\_CAR | 89 | 30% | 145 | 21% | 0.61 |
| USED\_CAR | 17 | 6% | 86 | 12% | 0.20 |
| FURNITURE | 58 | 19% | 123 | 18% | 0.47 |
| RADIO/TV | 62 | 21% | 218 | 31% | 0.28 |
| EDUCATION | 22 | 7% | 28 | 4% | 0.79 |
| RETRAINING | 34 | 11% | 63 | 9% | 0.54 |

Integer Variables:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Variable** | **Mean** | **Median** | **Std.Deviation** | **Max** | **Min** | **Range** |
| DURATION | 20.90 | 18.0 | 12.06 | 72 | 4 | 68 |
| AMOUNT | $3,271.16 | $2,319.50 | $2,822.63 | $18,424 | $250 | $18,174 |
| INSTALL\_RATE | 2.97 | 3.0 | 1.12 | 4 | 1 | 3 |
| AGE | 35.45 | 33.0 | 11.33 | 75 | 19 | 56 |
| NUM\_CREDITS | 1.41 | 1.0 | 0.58 | 4 | 1 | 3 |
| NUM\_DEPENDENTS | 1.16 | 1.0 | 0.36 | 2 | 1 | 1 |

* We found that the number of bad cases in certain value-ranges. Bad credit seems to be prevalent when the duration of credit is 12 to 36 months, credit amount is between 1500 and 5000 with many outliers beyond credit amount of 12000, age is between 25 and 40 with few outliers in 62-76 age group.

Variables, Most relevant to the outcome of interest**:**

**CHK\_ACCT** – Tells us the capability of the customer in repaying the credit taken from the bank.

**SAV\_ACCT** - Gives us the information about the savings of a customer.

**AGE** - Indicates the remaining customer’s employement period.

**HISTORY** – Gives us information about the punctuality of the customer in ***repaying credits taken.***

**AMOUNT** - Indicates how much credit a customer requested for.

**EMPLOYMENT** – Provides us the about the Employment status of the customer

***QUESTION 2***

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).

1. What decision tree node parameters do you use to get a good model. Explain the parameters you use

The below node Parameters have been used because they gave better accuracy (83.1%) with less overfit.

minsplit = 30 -> so that there are atleast 30 observations that must exist in a node in order for a split to be attempted.

minbucket = 10 -> so that there are atleast 10 observations in the terminal nodes.

cp = 0 -> if the cost of adding another variable to the decision tree from the current node is above 0, then tree building does not continue.

xval = 10 -> so that it gets cross validated 10 times

1. Which variables are important to differentiate “good” from “bad” cases - and how do you determine these? Does this match your expectations (from the your response in Question 1)?

One way of determining important variables is - the variables that give pure leaf nodes towards the top of the tree. So, CHK\_ACCT, HISTORY, SAV\_ACCT, EMPLOYMENT, AGE, AMOUNT are some of the variables are what I expected in Question 1. However, variable importance can be found out properly using Random Forest. We’ll do it in the steps where data is split between training and test set. So, let’s see what variables are important using Random Forest!!!

1. What levels of accuracy/error are obtained? What is the accuracy on the “good” and “bad” cases?

We achieved an Accuracy of 82.1% using the decision tree classification on full data.

Accuracy of good cases - 87.7%

Accuracy of bad cases - 69%

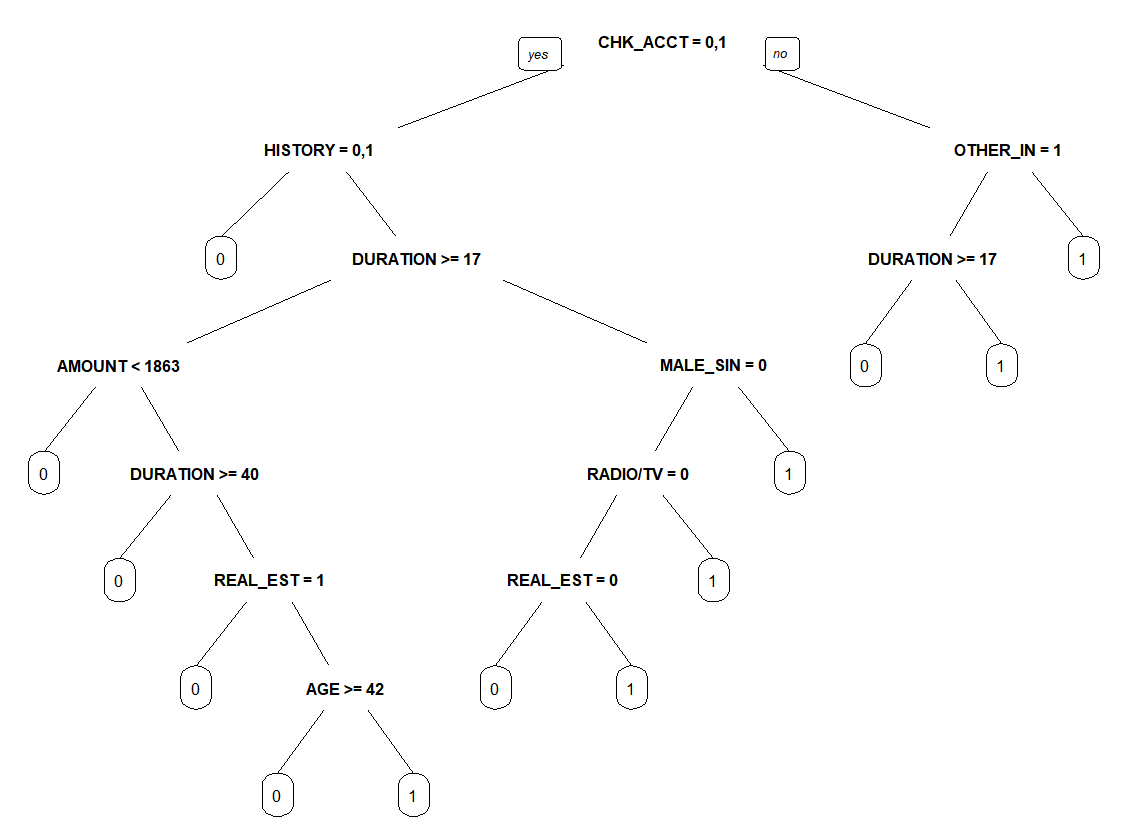
1. Do you think this is a reliable (robust?) description, and why.

This is not a reliable or robust description because this has been done using the entire data and a small change in data causes a large change in the final estimated tree; we should split it into training and test set in order to get a robust description.

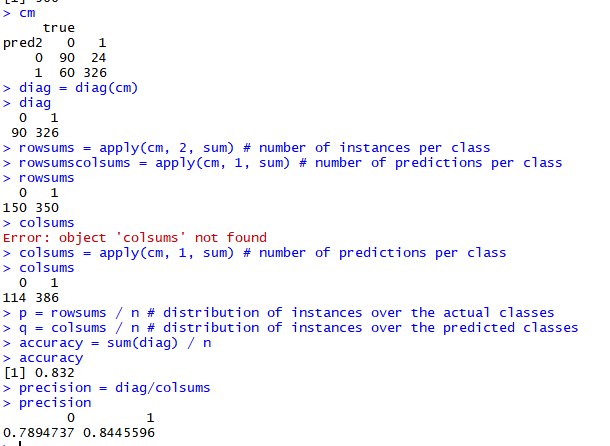
***QUESTION 3***

*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*

1. *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.*



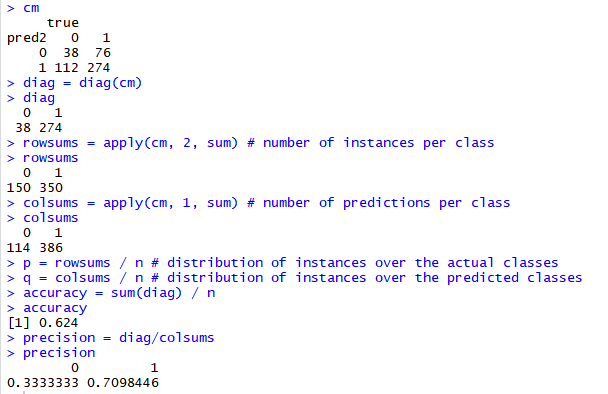
Confusion Matrix for Training Dataset :



For training data set the overall accuracy came out to be 83.2 %. The accuracy of Good Cases is 84.45% and the accuracy of bad cases is 78.94%.

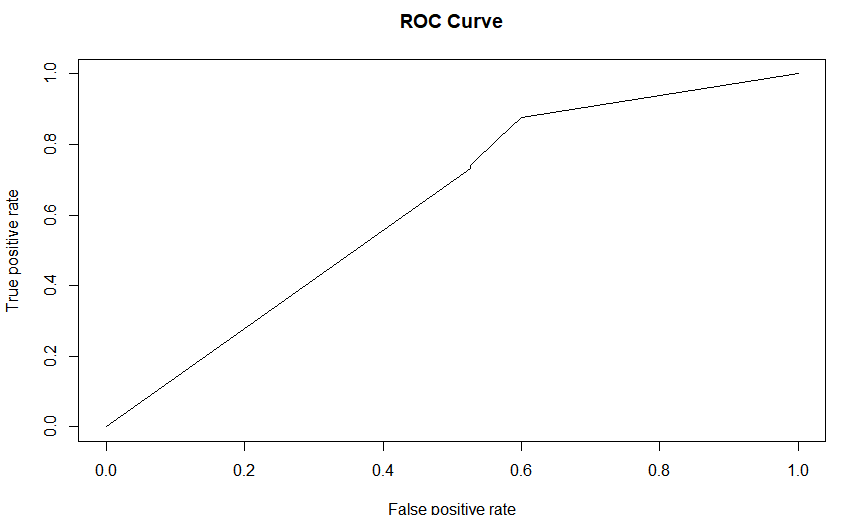
Sensitivity = 93.14% and specificity is 60%

The Confusion Matrix for test dataset :

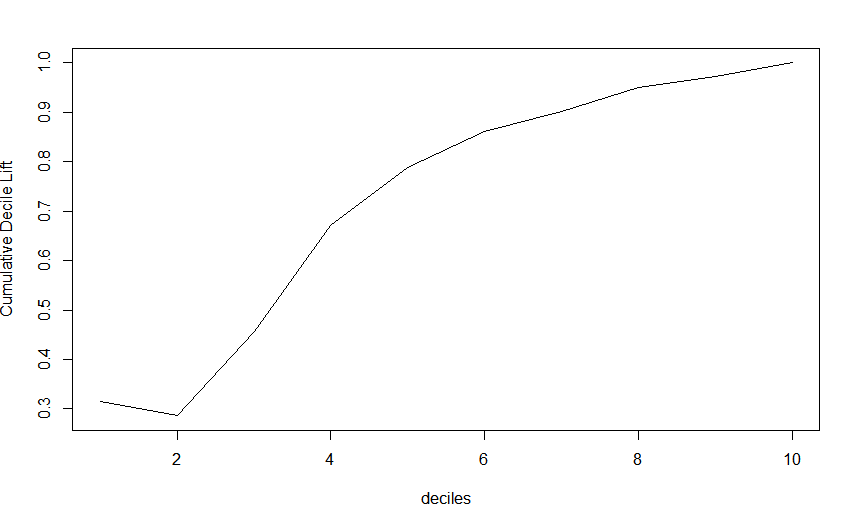


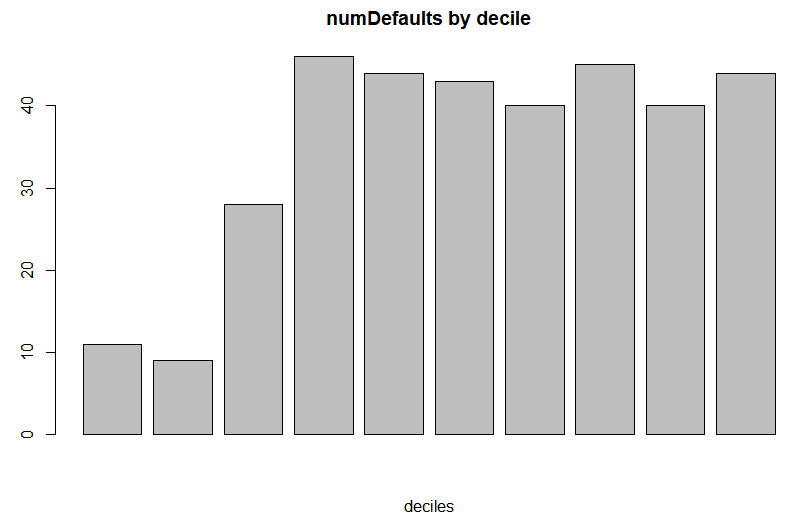
The overall accuracy of the test dataset is around 62.4%. The accuracy of good cases is 70.98% and the accuracy of bad cases is 33.33%. The sensitivity for test data is 78.2% and specificity is 25.3%

Measures such as accuracy, recall/sensitivity and precision have been used. Calculating these variables was essential as German credit union might look at the predictive model and its performance from various aspects based on their operational constraints.



Plotted ROC curve. Area under ROC curve (AUC) =0.627

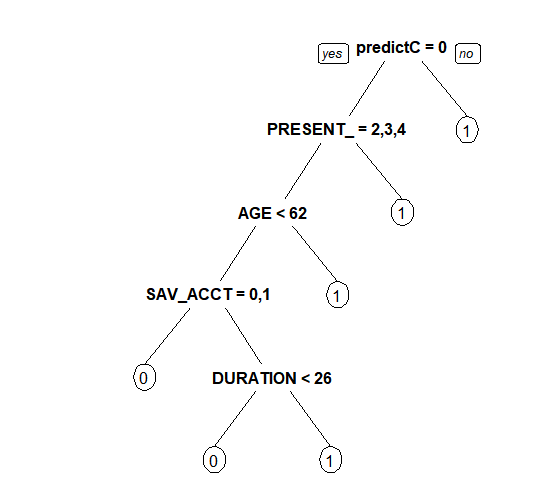




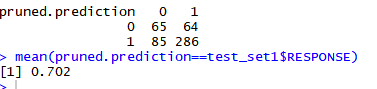
In developing the models above, change decision tree options as you find reasonable (for example, complexity parameter (cp), the minimum number of cases for split and at a leaf node, the split criteria, etc.) - explain which parameters you experiment with and why. Report on if and how different parameters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

We have used minsplit, minbucket, cp and xval where minsplit – least number of obs that have to be present in a node for further splitting to be done. Increase in minsplit value leads to decrease in performance. Least number of obs that have to present in any terminal node is minbucket. Increase in minbucket also leads to decrease in performance. Because with more count of values in the root its is likely to have high proportion of defaulters. Increase in cp leads to decrease in performance however number of cross validations xval doesnot affect the performance much.

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?



We have used cost-complexity pruning. The effect of changes in values of CP appears in metrics like accuracy, sensitivity, specificity and precision



The best tree has been selected based on accuracy which is 70.2% which occurs at CP value of 0.009. for any value greater accuracy tends to decrease. The main parameter values are minsplit, minbucket, cp, xval and split = information

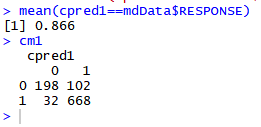
*b) Consider another type of decision tree – C5.0 – experiment with the parameters till you get a ‘good’ model. Summarize the parameters and performance you obtain.*

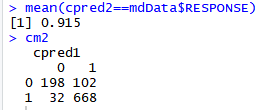
*Also develop a set of rules from the decision tree, and compare performance.*

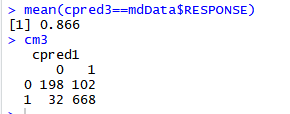
1. With Parameters - Formula, data, and method, accuracy - 86.6%
2. Added trials = 3 which increased the accuracy to 91.5%. Trials are the number of boosting iterations
3. Added costs (a matrix of costs associated witht the possible errors) as NULL and weights(optional numeric parameter of case weights) as NULL, hence, doesn’t change the accuracy, the accuracy is 86.6%
4. Adding winnowing = FALSE doesn’t have any effect on accuracy

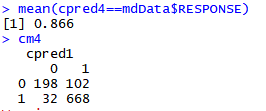
RULES = TRUE has been added in all models

Also develop a set of rules from the decision tree and compare performance. The rules have been specified using the ‘summary’ function in the chunk below and the code is mentioned in the chunk below, the performance for cModel2 is the highest with an accuracy of 91.5%.









*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?*

Here, till now, we are going to compare 50:50 Split, pruned tree and C5.0 tree.We have compared models on accuracy because it determines how often is the classifier correct. As discussed in the previous cases, Accuracy for 50:50 model is 83.2% Accuracy for pruned 50:50 model is 70.2% Accuracy for C5.0 tree is 86.6% and with trials set to 3, it’s 91.5%

Comparing the performance on lift, ROC and AUC - this part of 50:50 split model has been covered in the first part. Here, AUC came out to be 62.7%(Please note that other factors - Recall/Sensitivity, Specificity and Precision have been computed earlier while doing the 50:50 Split)

Based on all these factors, C5.0 seems like a good model. However, more insights into importance of variables and classification can be acquired after the splitting of dataset into 70-30 and 80-20 split.

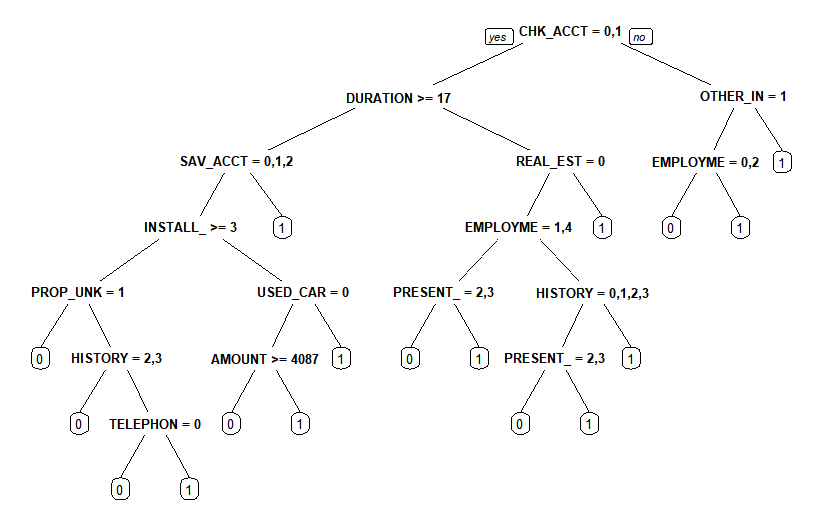
50:50 split has been performed on the training set (50% of the entire dataset) and test set(remaining 50% of the dataset), pruned tree uses the cost complexity pruning, and C5.0 model extends the C4.5 classification algorithms. The model can take the form of a full decision tree or a collection of rules.

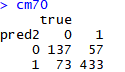
*d) Consider partitions of the data into 70% for Training and 30% for Test, and 80% for Training and 20% for Test and report on model and performance comparisons (for the decision tree learners considered above).*

*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 what does this indicate?*

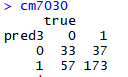
*Is there any specific model you would prefer for implementation?*

**70:30 data split**



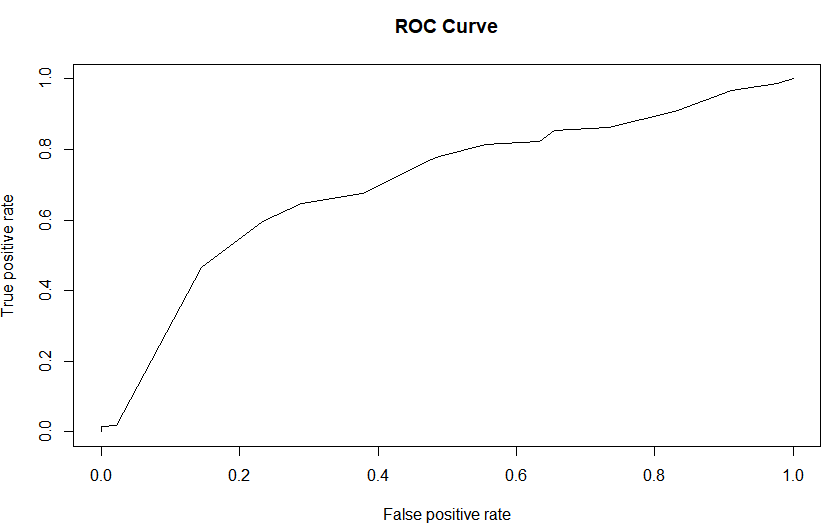


For training data set: The accuracy of the model 81.42% . sensitivity or recall is 65.42% and specificity is 88.37% and Precision(Positive Predictive Value) - 70.62%.

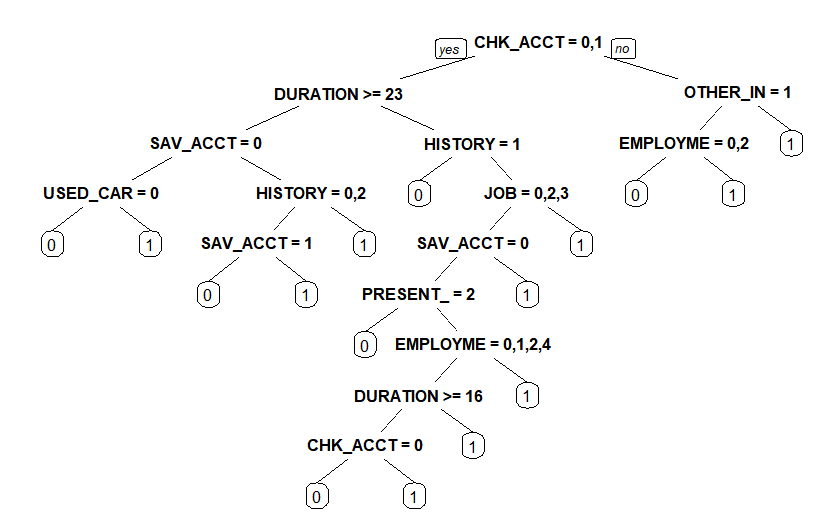


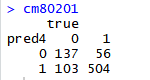
For test dataset: The accuracy of the model 68.67% . sensitivity or recall is 36.67% and specificity is 82.38% and Precision(Positive Predictive Value) – 47.14%.

The area under curve for 70:30 curve is 0.7024

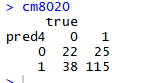


**80:20 data split**



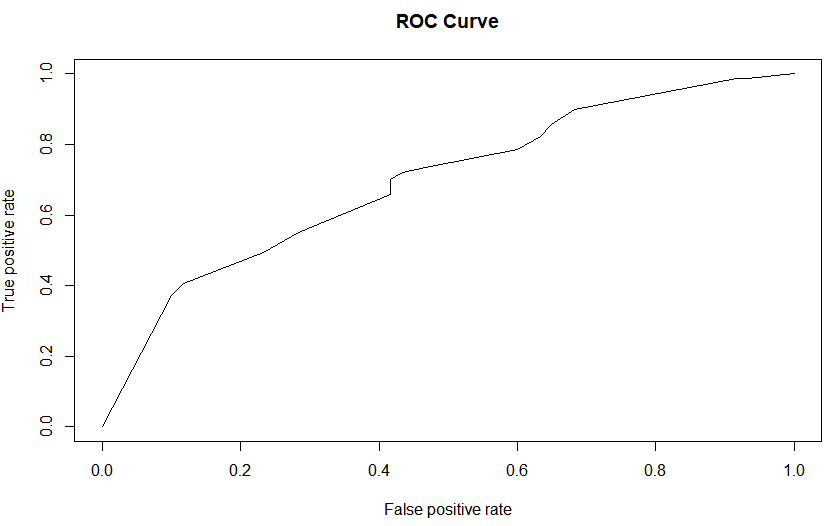


For training data set: The accuracy of the model 80.25% . sensitivity or recall is 60.83% and specificity is 88.57% and Precision(Positive Predictive Value) – 69.52%.



For test data set: The accuracy of the model 67.00% . sensitivity or recall is 40% and specificity is 78.57% and Precision(Positive Predictive Value) – 44.44%.

The are under the ROC curve for 80:20 split is 0.6943



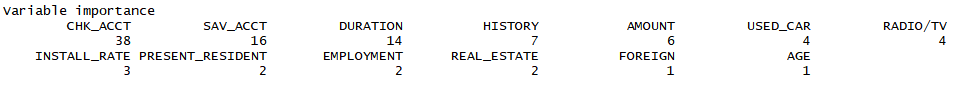
Though the 80:20 split has a better accuracy for test data set of 67%, 70:30 split looks better as the area under the curve for this graph is 0.7024 which is higher than the other two splits and it also has a training dataset with an accuracy of 81.42%. 70:30 split is preferred

*(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?*

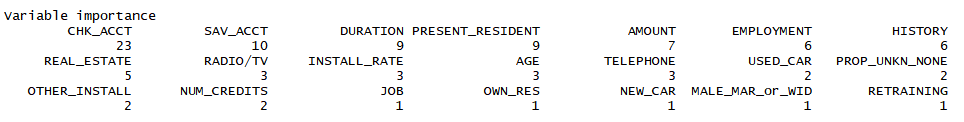
As we have concluded that 70:30 split is better. For seed 123 the accuracy of the training dataset was around 81.42% . set.seed (90) accuracy is 78.17% and with set.seed (16666) the accuracy is 76.64%. Based on above observations we can conclude that the decision tree model is unstable as it is giving different accuracy values for different values of the seed.

*(d) Which variables are important for separating ‘Good’ from ‘Bad’ credit? Determine variable importance from the different ‘best’ trees. Are there similarities, differences?*

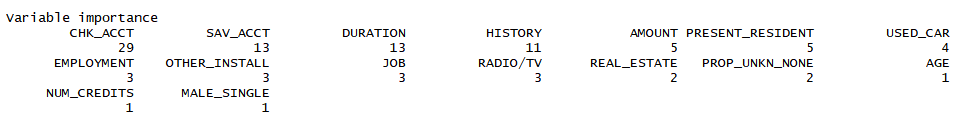
**50:50 split**



**70:30 split**



**80:20 split**



The important variables are CHK\_ACCT, SAV\_ACCT, DURATION, HISTORY for separating good and bad credit. These variables are similar across all three decision trees. The variable importance has been obtained using the summary function in R.

***Question 4***

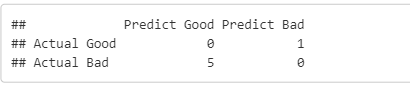
*Consider the net profit (on average) of credit decisions as:*

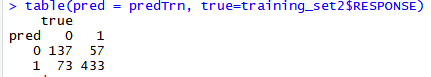
*Accept applicant decision for an Actual “Good” case: 100DM, and*

*Accept applicant decision for an Actual “Bad” case: -500DM*

1. *Use the misclassification costs to assess performance of a chosen model from Q 2 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?*

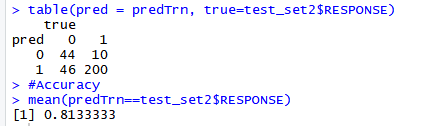
The performance (accuracy) of 70:30 split model is increased to 81.2% when misclassification cost is used for training dataset.





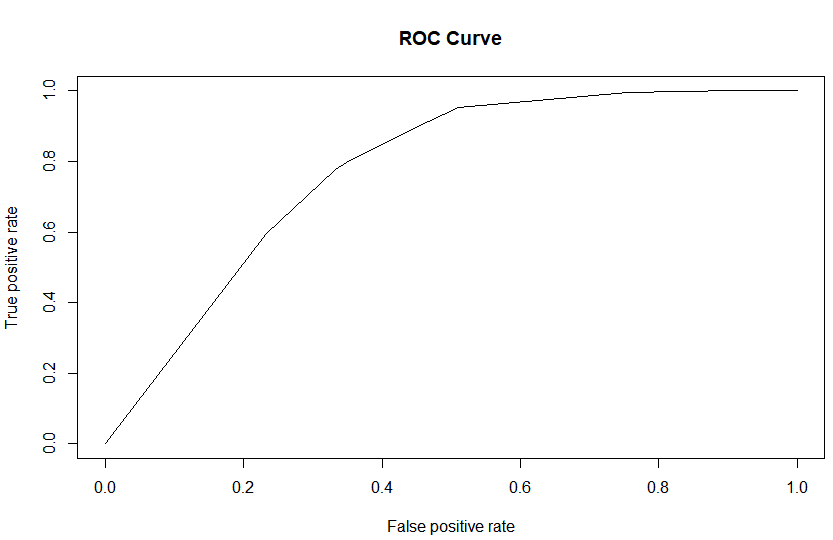


The performance(accuracy) increases by a significant amount for the test data set as well, when misclassification cost is used.



Now moving forward and examining how different cutoff values for classification threshold make a difference, we observe that there is an inverse relationship between CTHRESH and Accuracy.

Accuracy for CTHRESH of 0.5 is 76.80%.When the CTHRESH was lowered to 0.23 from the default 0.5 accuracy further increased to 81.1 % . The ROC curve used for obtaining the CTHRESH is depicted below .The area under the curve came to 0.7747



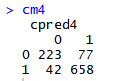
1. *Calculate and apply the ‘theoretical’ threshold and assess performance – what do you notice, and how does this relate to the answer from (a) above.*

The accuracy after applying theoretical threshold is 83.33



1. *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).*

Yes, the trees are different from the ones obtained earlier. Accuracy for rpart tree is 83.3% and for C5.0 is 88.1%.



1. 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?

‘Best’ decision tree model is the one with 70-30 split with the decision node parameter as minSplit = 10, minbucket = 30, cp = 0, and split based on information. Tree Depth - 6 Total nodes - 16 Most important variables - CHK\_ACCT, DURATION, HISTORY, OTHER\_INSTALL, SAV\_ACCT and EMPLOYMENT Relatively Pure Leaf nodes - CHK\_ACCT and DURATION. This can be observed from the summary function output in the previous question

1. 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 this analysis.

b) 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.

c) 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.

The cost came out to be 0.2063636. We obtained the max profit of 61.90

As indicated by the curve and the calculation, the cutoff value has come out

to be 0.6296

