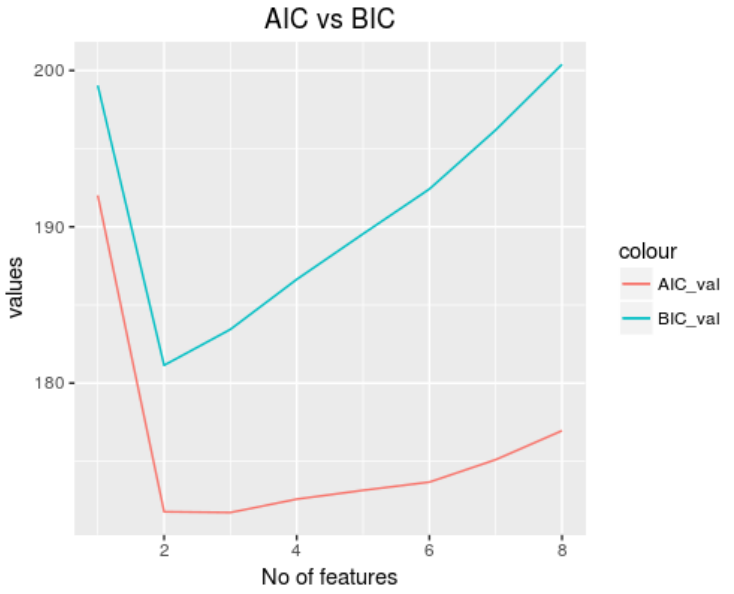
Homework – 4

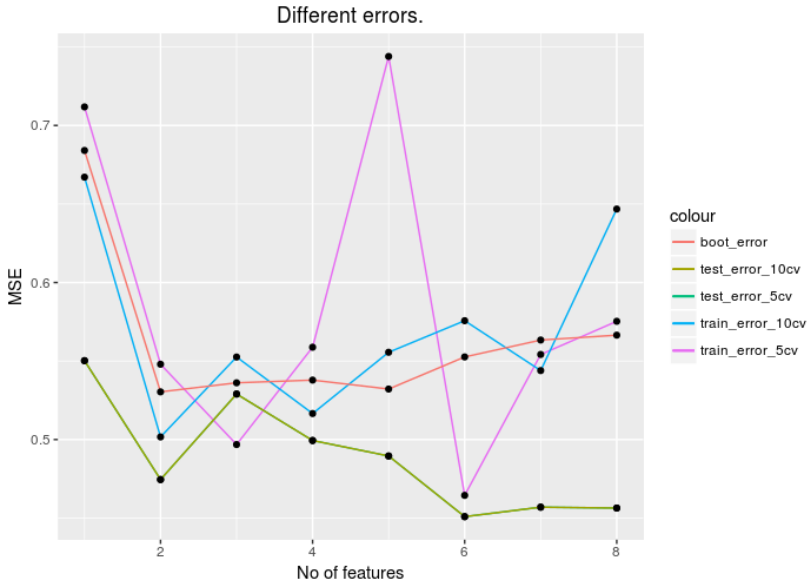
Q1) Following are the steps that we followed.

- Doing a sanity check to see if there are any NA values in the data. We find that there are none.
- Dividing the datasets into training set and testing set.
- Doing an exhaustive subset selection using the method, regsubsets in the package leaps.
- According to the Cp value, 2 features subset– lcp and lpsa are the best features.
- We get the following graph when we check for AIC, BIC values across different subset selections.



Inference: AIC and BIC values too agree with the Cp value that is 2 feature subset gives the best results.

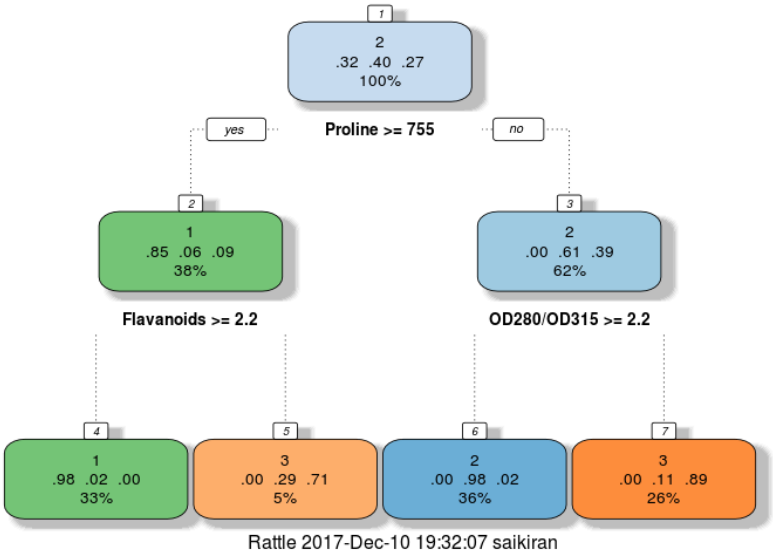
- Following graph is obtained when we use 5CV, 10CV, Bootstrap .632 strategies.



Inference: In almost all the strategies we get that using 2 features is best for this dataset. The least MSE estimate for this test dataset is obtained when we use Cross Validation with 10 folds.

Q2) Following are the steps that were followed.

- Doing a sanity check to see if there are any NA values present in the dataset. We find that there are no NA values.
- The Wine feature that is the target variable is a categorical variable. But it is a numeric feature in the original dataset. Hence converting the feature into categorical.
- Dividing the dataset into training and testing dataset.
- Fitting a decision tree on the training set and plotting the decisions gives us the following graph.



Inference: We can see that data is very separable since the the purity of leaf nodes is high despite fewer decisions (Smaller tree).
And also despite having 13 predictors, we can see that the decision is made using only 3 features – Proline, Flavanoids and OD280/OD315.

- Following are the distributions of training set in the leaf nodes.
  node4 (class –1) = 47
  node5 (class – 3) = 7
  node6 (class – 2) = 51
  node7 (class – 3) = 37
  We can observe that predicting class 1, 2 is relatively easy since the purity of those nodes is high. But predicting class 3 is tough since their leaf nodes' purity is less (nodes 5, 7)

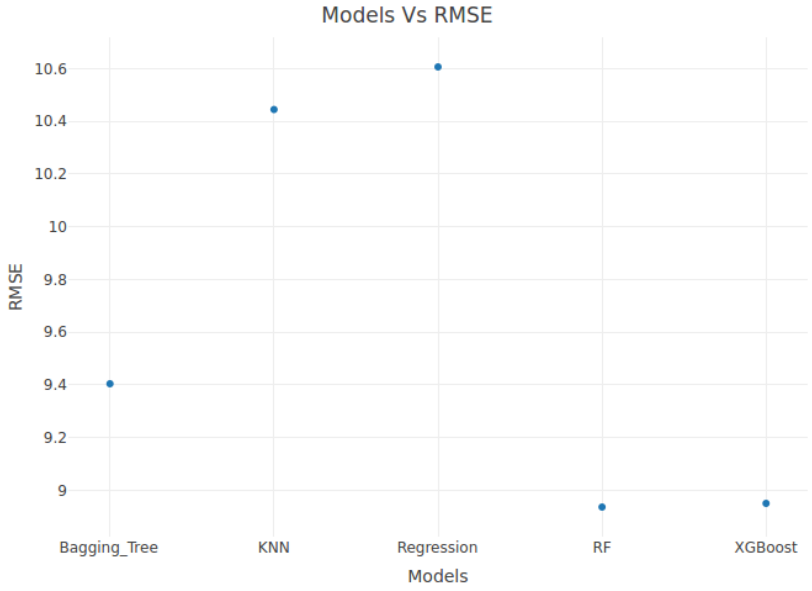
- Following is the strategy that was used to check which data points go into which leaf node.
  We can see from the above picture that class – 1 and 2 go to nodes 4, 6 respectively. Hence when the model predicts a datapoint as class 1 / 2 then we can assert that the datapoint goes to node 4 / 6 respectively. For class 3 we can check if the datapoint has the first condition – Proline >= 755 since both the nodes are from different branches all together.

- Following the above strategy following is the distribution of test set in the leaf nodes.
  node4 = 12
  node5 = 1
  node6 = 11
  node7 = 12

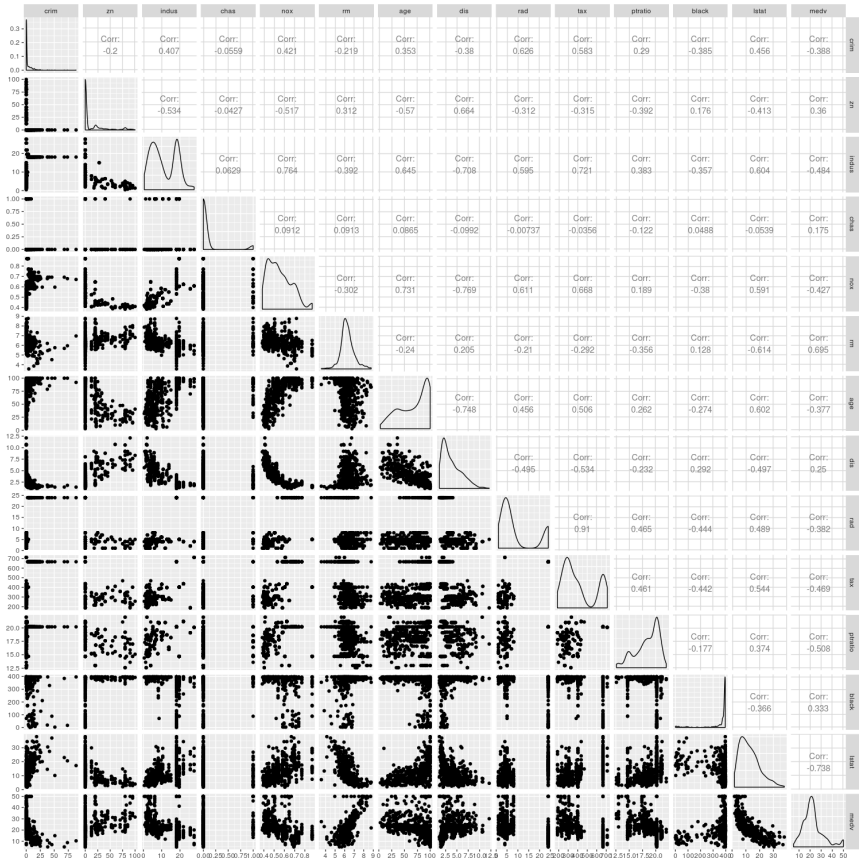
- The train and test accuracy are 94% and 81% respectively.

Q3) Following are the steps that were followed.

- Doing a sanity check to see if there are any NA values in the data. We find that there are no NA values in the data.
- Splitting the data into training set and testing set.
- Running different models on the dataset and plotting gives following figure. Decision Trees are used in bagging.

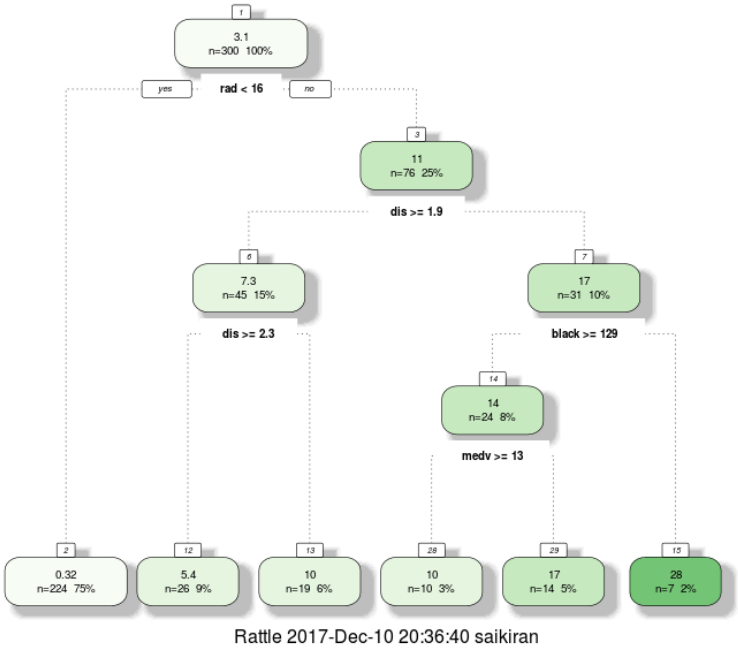


Inference: We can observe that the test RMSE is lowest for XGBoost and Random Forest.
The reason why tree based models are doing good for this data could be because of a proper blob like seperation along X / Y axis. Plotting the pairplots to confirm this, we get the following figure.



Inference: We can see that there is good seperation structure by which taking mean (since numeric target) is easier.

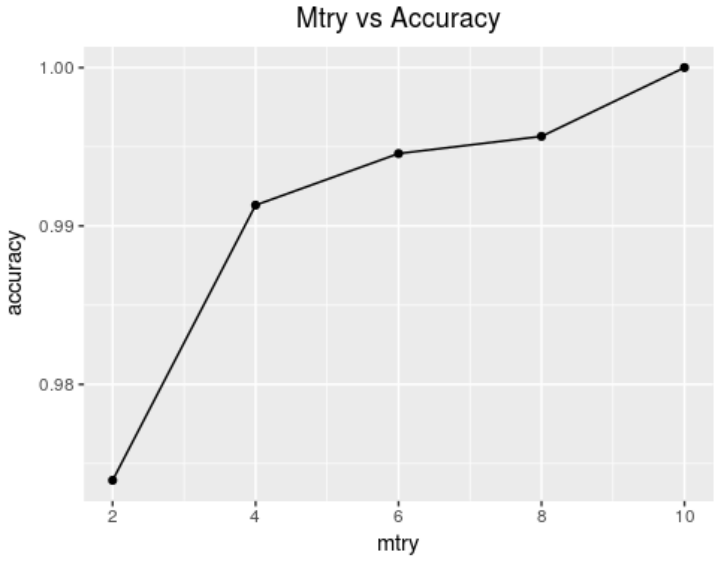
- To confirm further, plotting the decision tree gives us the following figure.



Q4)

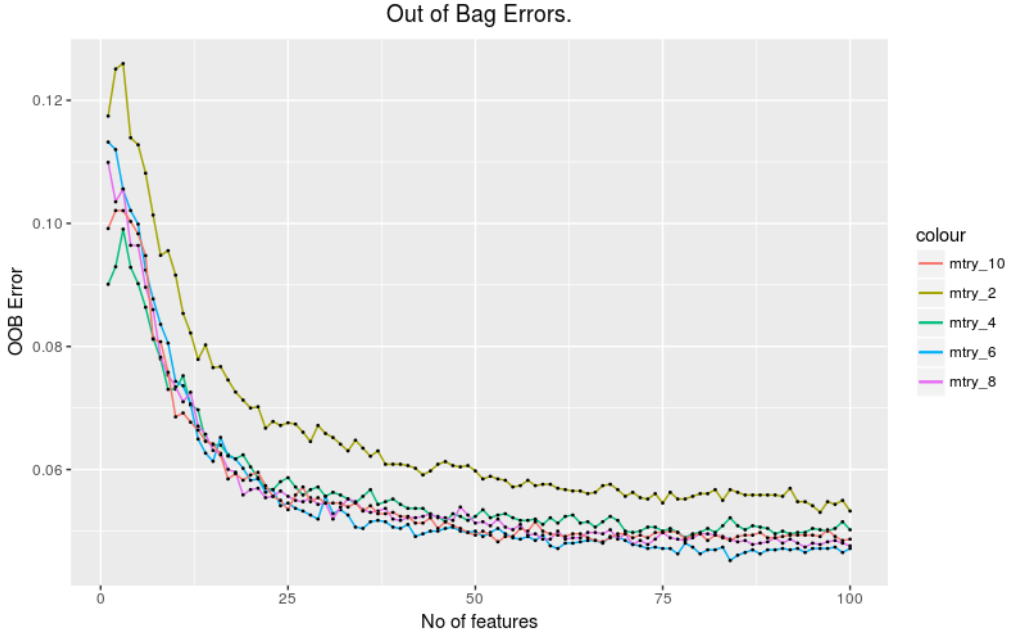
Following are the steps that were followed.

- Doing a sanity check for the data to see if there are any NA values. We observe that there are no NA values.
- Splitting the data into training set and testing set.
- Iterating through different values from 2 to 10 as the mtry hyper parameter and plotting it gives us the following figure.



Inference: As it is expected, as the number of features increase in a tree the accuracy increases.

- Random Forest models have been built using 100 trees. Plotting Out of Bag errors for different values of mtry gives us the following figure.



Inference: As it is expected, like above, with the increase in number of features the error decreases. Hence the error of mtry_2 is consistently more out of all the curves in the above figure.

Q6)

I have used “Gender Classification by Voice” dataset from Kaggle. Following is the link: https://www.kaggle.com/primaryobjects/voicegender
The dataset has 3100 rows and 21 columns.

Following are the steps that were followed.

- Doing a sanity check for the data to see if there are any NA values in the dataset. We observe that there are no NA values.
- Dividing the dataset into training and test set.
- Introducing an outlier in the training data. The boxplot with and without outlier that we introduced are below.

Inference: We can clearly see that the value that we changed is an outlier.

- Running a Neural Network with the original data (without the introduced outlier) gives an accuracy profile like follows when the number of neurons in the hidden layer is varied from 1 to 20.

```
> acc
 [1] 0.9763407 0.9763407 0.9763407 0.9700315 0.9779180 0.9716088 0.9779180 0.9763407 0.9794953 0.9794953
[11] 0.9794953 0.9826498 0.9842271 0.9794953 0.9842271 0.9826498 0.9826498 0.9810726 0.9810726 0.9826498
```

Inference: There is no much difference in the accuracy profile.

- Running the Neural Network with the introduced outlier gives an accuracy profile like follows when the number of neurons in the hidden later is varied from 1 to 20.

```
> acc
 [1] 0.9779180 0.9763407 0.9826498 0.9763407 0.9763407 0.9794953 0.9779180 0.9731861 0.9779180 0.9810726
[11] 0.9747634 0.9763407 0.9842271 0.9826498 0.9810726 0.9858044 0.9826498 0.9826498 0.9842271 0.9826498
```

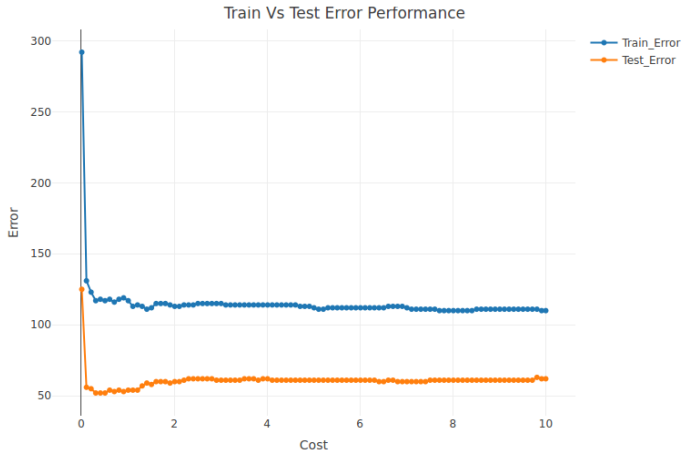
Inference: There is no much variation in the accuracy profile.

- **Inference:** The probable reason why there is no change in the accuracy profile in the second case when compared with first case is because of the number of data points that we have. A single outlier might not cause too much variation to the weights that a neural network learns.

Q7)

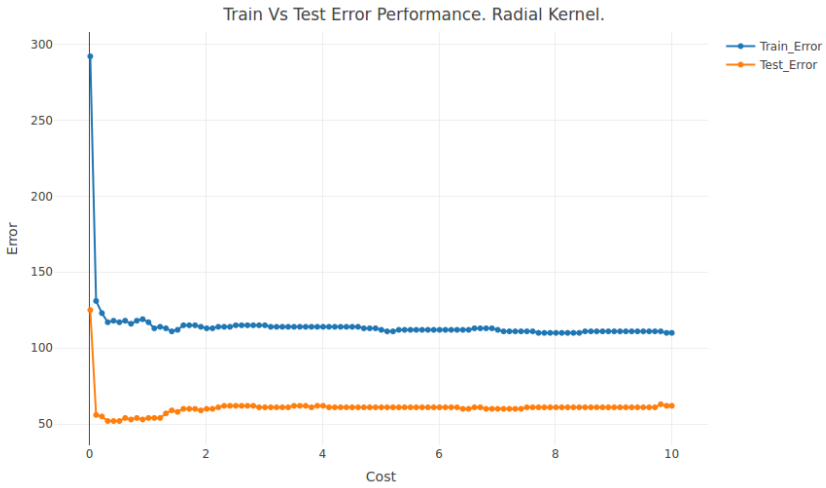
Following are the steps that were followed.

- Checking the sanity of the data if there are any NA values in the data. We observe that there are no NA values in the dataset.
- There are features like StoreID, SpecialCH, SpecialMM, Store that are categorical variables but are actually numeric in the dataset. Hence changing these variables into categorical variables.
- Dividing the dataset into training and test datasets.
- Iterating the hyper parameter, cost from 0.01 to 10, calculating the errors and plotting gives us the following figure.



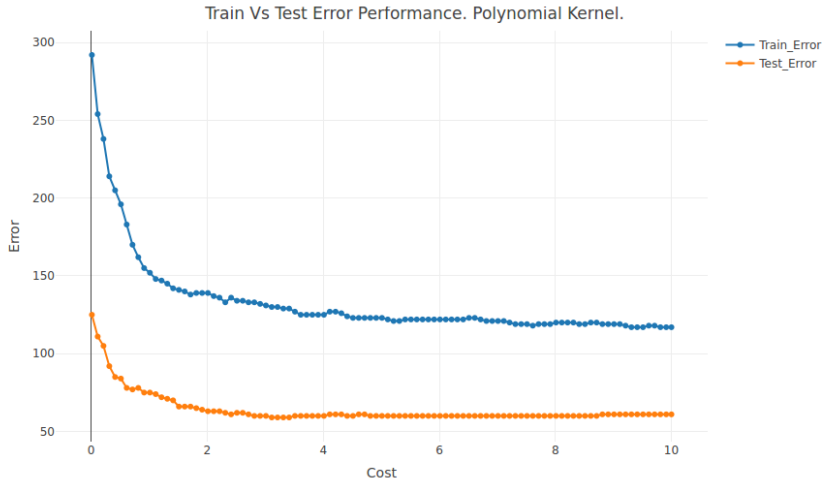
Inference: We can see that the error values for cost greater than around 2 are the worst. Choosing a cost parameter around 0.1 to 1 seem to be doing good.

- Iterating the hyper parameter, cost from 0.01 to 10, using a radial kernel, calculating the errors and plotting gives us the following figure.



Inference: We can observe a similar error profile like in the above case. For the cost values that are greater than 2, models doesn't seem to be performing well.
Choosing a cost parameter between 0.1 to 1 seem to give best results.

- Iterating the hyper parameter, cost from 0.01 to 10, using a polynomial kernel with degree 2, calculating the errors and plotting gives us the following figure.



Inference: Unlike the above figures we can see that with the increase in cost hyper parameter the model seem to be doing better. But it platues after a while, from the figure it seems like it platues after 3.