# MLT - Assignment 2

Ayush Sekhari

March 1, 2015

## 1 Observations

1. We do not have missing attributes

## 2 Part-a

I performed a binary search between 2 and 450 and chose the point at which the data started to level off. That is, the error is in the difference range of 4.5%-5% of the minimum error.

The number of trees where the error curve started to level off: 198 and error at this point is : 3.515%

The plot is shown in the figure below and the levelling off point has been marked in red.

## 3 Part-b

The tree size for the above case = 198, The OOB error for this forest over the entire training data is = 3.195%

## 4 Part-c

The data table shows that too much randomness or strictness is not good. I get comparatively higher error for small values of m(i.e. m = 1 , more randomness) and for large values of m(i.e. m = 8 , less randomness).

| **m Value** | **CV error** |
|---|---|
| **m = 1** | 0.05145 |
| **m = 2** | 0.03450 |
| **m = 3** | 0.03465 |
| **m = 4** | 0.03510 |
| **m = 5** | 0.03625 |
| **m = 6** | 0.03685 |
| **m = 7** | 0.03800 |
| **m = 8** | 0.04105 |

Table 1: CV errors for different values of splitting attribute number for trees in the forest

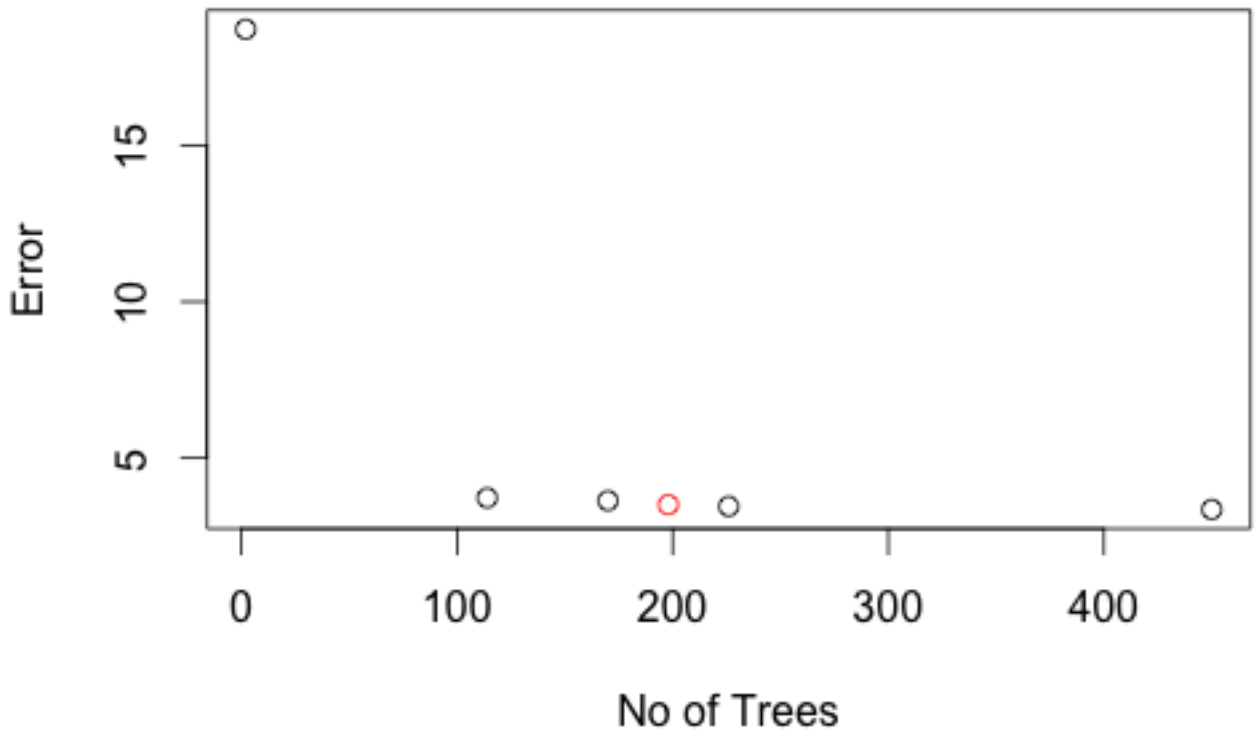


Figure 1: Plot of error rates vs number of trees while performing Binary search

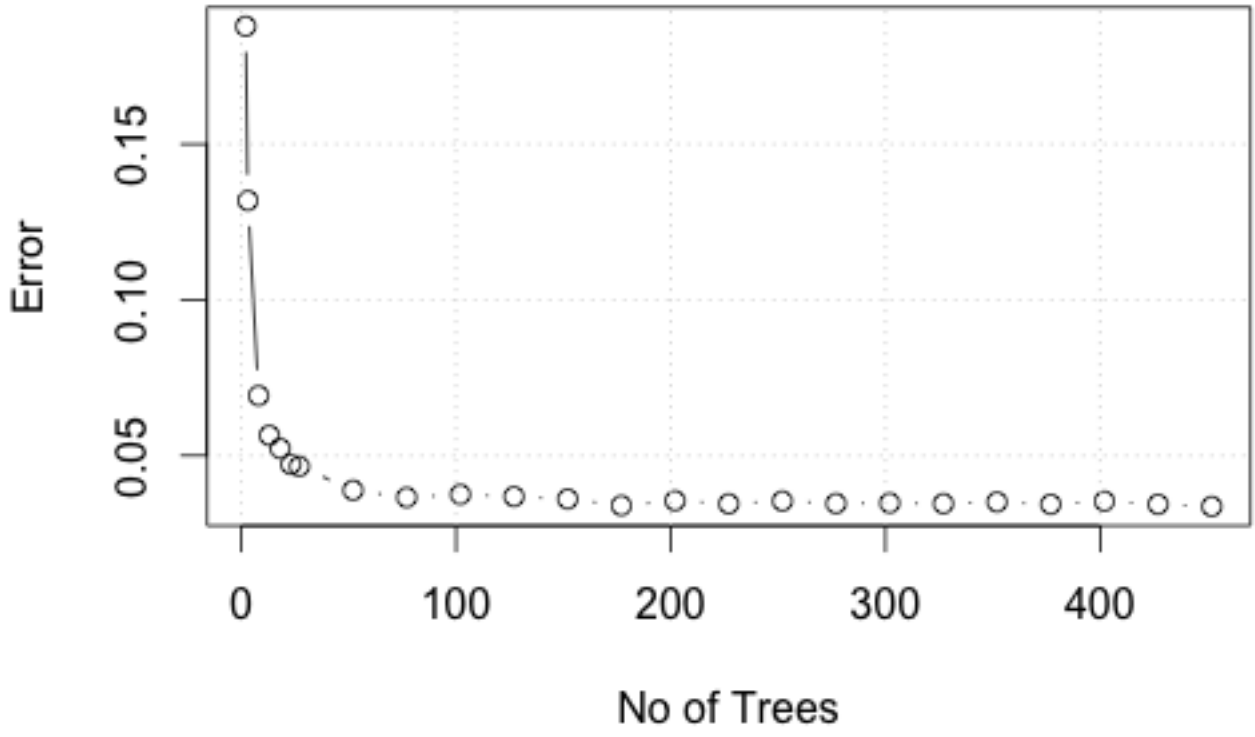


Figure 2: General Trend of Error vs Number of Trees for Random Forest

# 5 Part-d

In this part, I used the same algorithm as in Part - (a) to find the point where the error curve levels off.

- Figures 3 and 4 show the plot of 5 fold cross-validated error for a forest with T trees against the size of the sampled data set expressed as a percentage of L when 4 attributes are randomly used for splitting. It has a local minima at 60% with error value = 3.745%.
- Figures 5 and 5 show the plot of 5 fold cross-validated error for a forest with T trees against the size of the sampled data set expressed as a percentage of L when all the attributes are used for splitting. It has a local minima at 55% with error value = 5.64%.
- This shows that the maximum accuracy comes when training data is $\sim$ 55-60% of the total data available for training.
- High values of the error for the cases of smaller sample sizes is because we do not have much data to train on, therefore the random forest formed has lesser strength and correspondingly lesser accuracy.
- We observe high error values when sample size is large because there would be a huge sharing of training data between the two trees. This increases the correlation in the forest and thus the accuracy decreases.
- In case of bagging, the expected size of the bag is $\sim$ 63% of the training data available.
- Since, most accuracy comes at around $\sim$55-60%, this shows that bagging is a good approximation for fixing training data for trees in the forest.
- I would prefer bagging as it introduces randomness in the model. There may be some trees trained on bigger data sets and some trained on smaller data sets. This introduces variation in the forest which is good to some amount.
- With bagging we get an error of 5.665% with 52 trees which is comparable to the minimum error observed with no bagging.

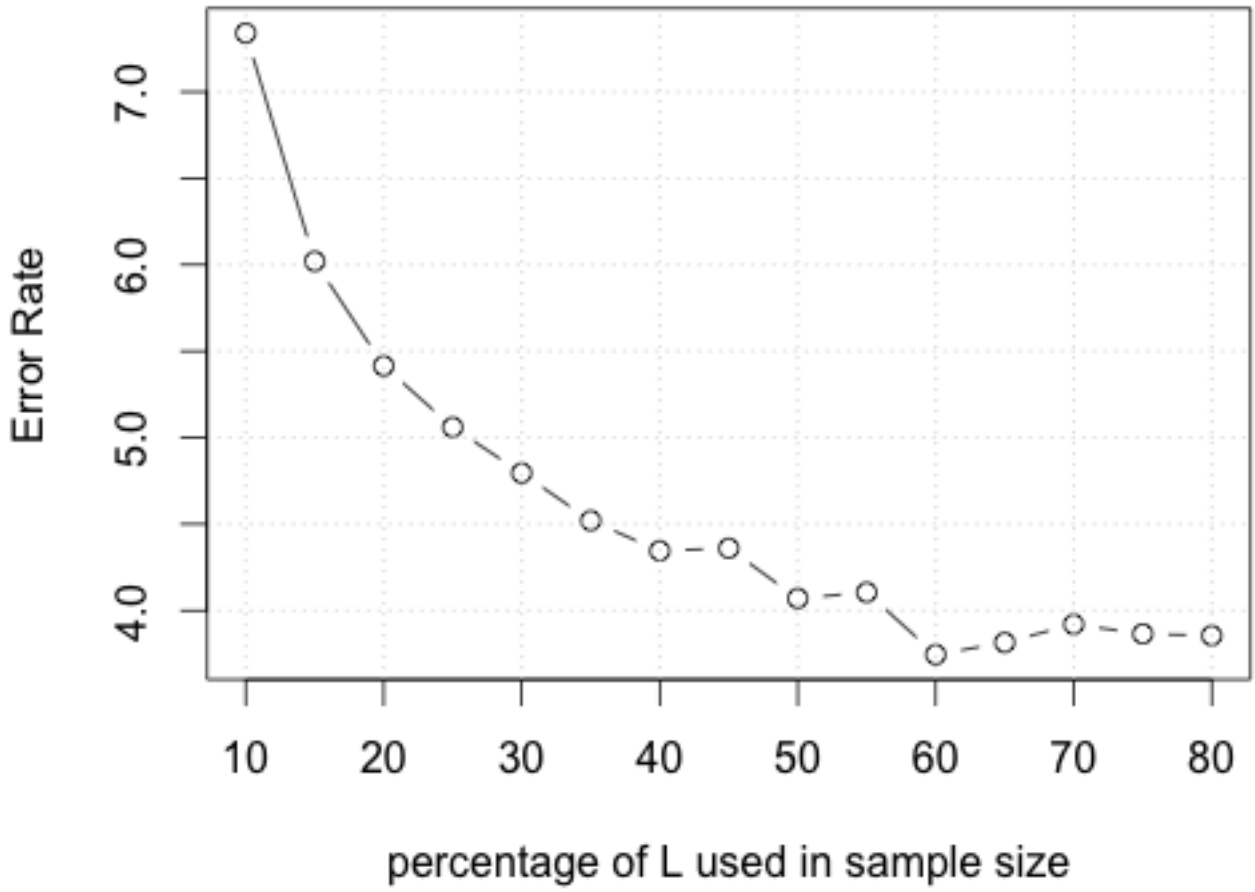


Figure 3: 5 fold cross-validated error for a forest with T trees against the size of the sampled data set expressed as a percentage of L with mtry = 4

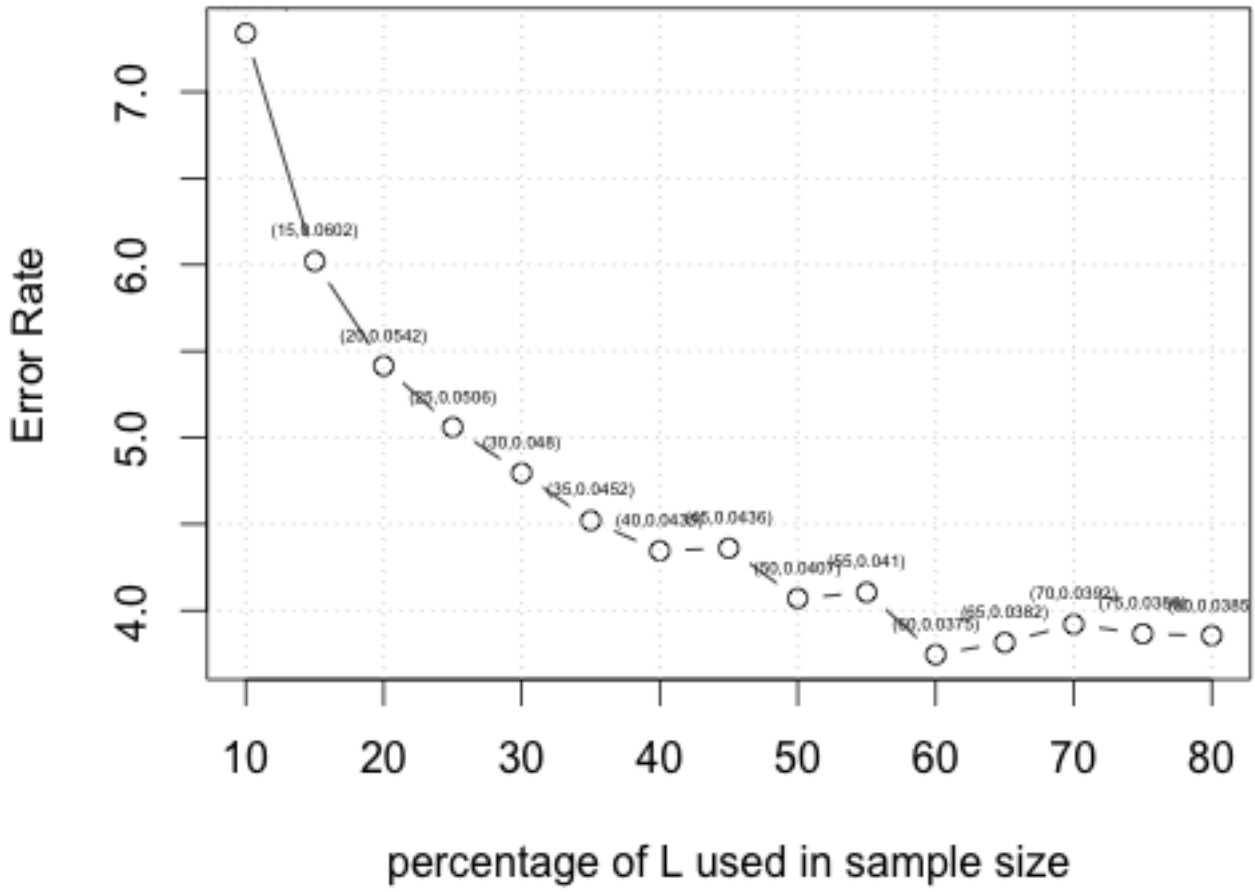


Figure 4: 5 fold cross-validated error for a forest with T trees against the size of the sampled data set expressed as a percentage of L with mtry = 4

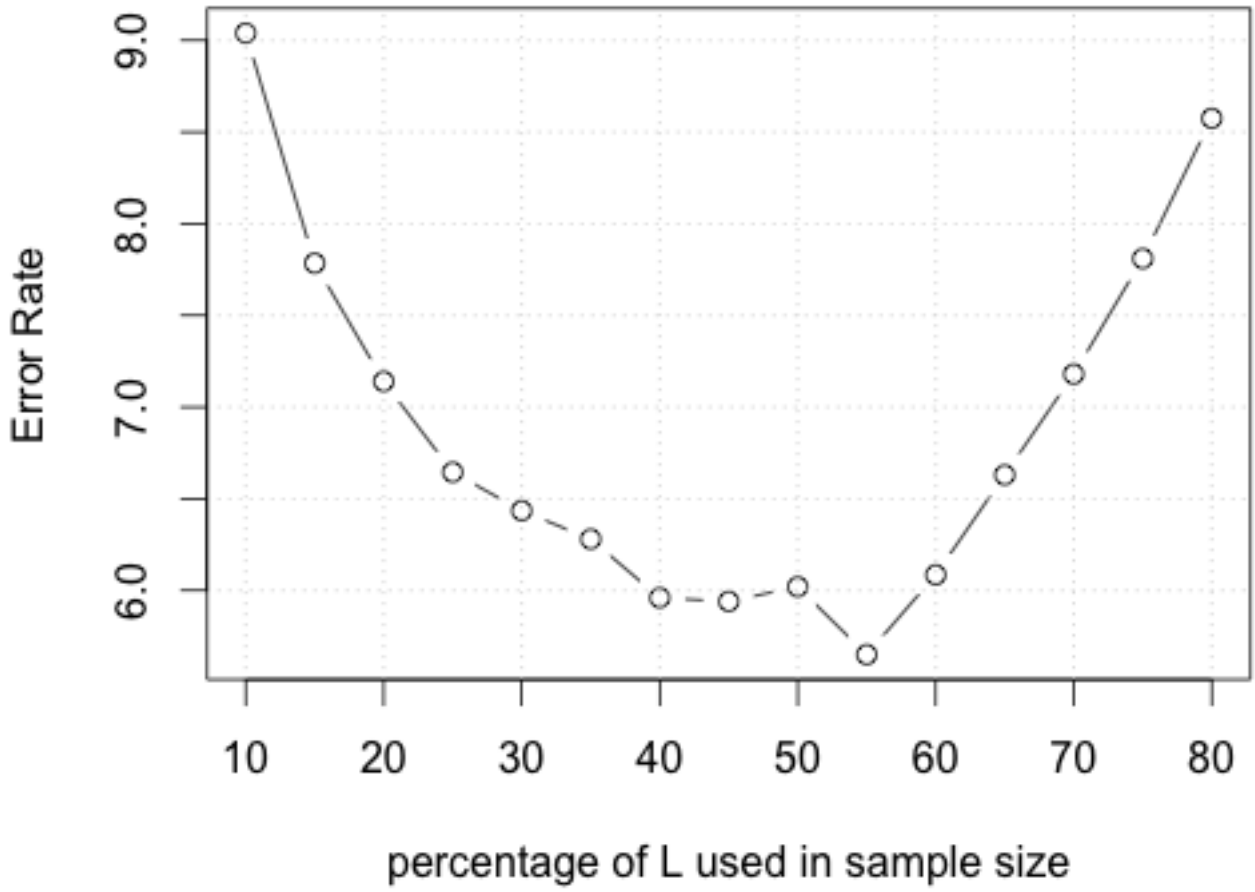


Figure 5: 5 fold cross-validated error for a forest with T trees against the size of the sampled data set expressed as a percentage of L with mtry = 16

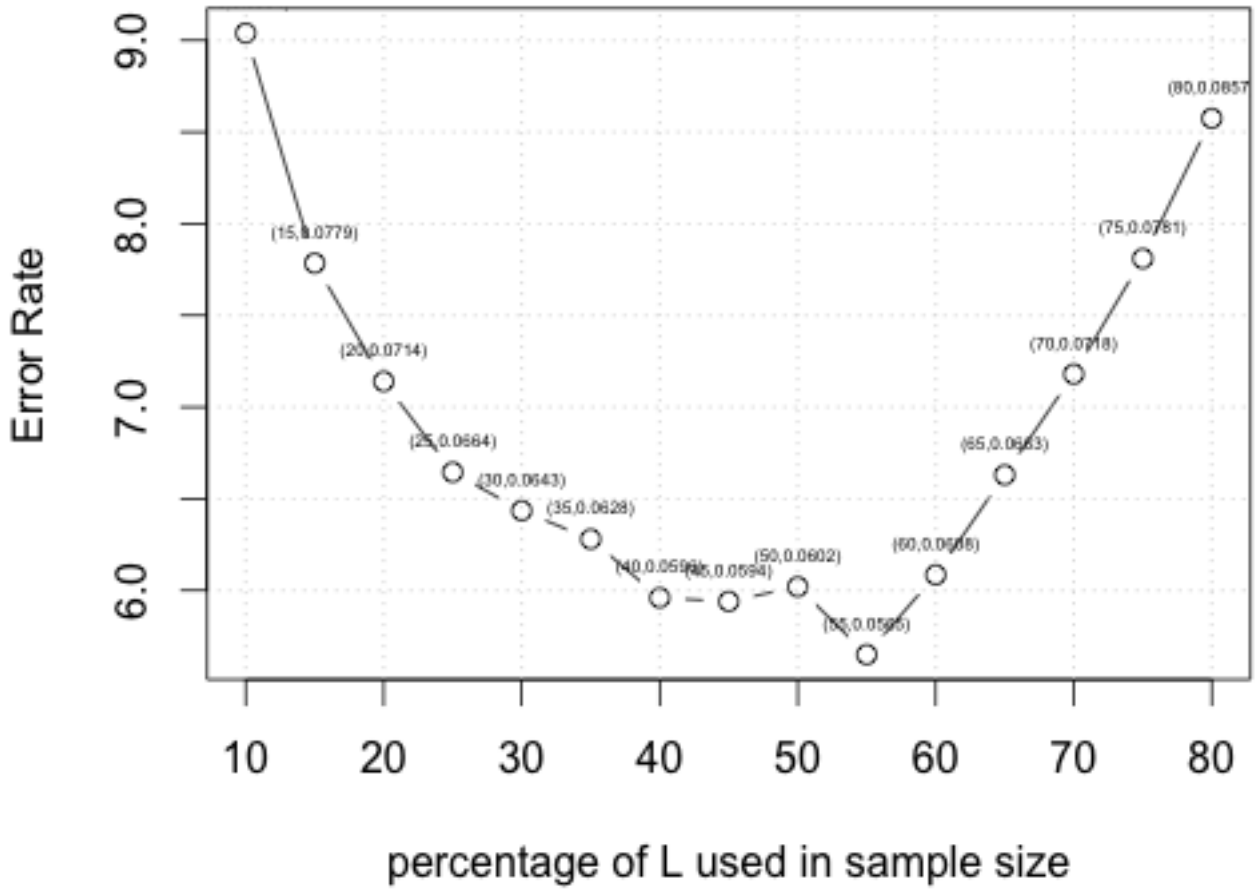


Figure 6: 5 fold cross-validated error for a forest with T trees against the size of the sampled data set expressed as a percentage of L with mtry = 16