Homework 4 : Statistical Data Mining (EAS 506)

Shubham Sharma, Person No.: 50290293, Class No.: 44

November 29, 2018

Question 1: For the prostate data of Chapter 3, carry out a best- subset linear regression analysis, as in Table 3.3 (third column from the left). Compute the AIC, BIC, five- and tenfold cross-validation, and bootstrap .632 estimates of prediction error.

The Prostate Cancer Data Set has been taken from the ElemStatLearn package. The dataset has 97 observations and 10 variables where the last column ”train” is used to depict the split between train and test dataset. We split the dataset into train and test set using this column. The training set has 67 observations and the test set has 30 observations.

The model containing all of the predictors will always have the smallest RSS and the largest R^2. Thus, we adjust our training error to be able to select among a set of models with different number of variables. To implement this we adopt the following approaches: C_p and *Bayesian information criterion (BIC)*. We know, for least squares models, C_p and AIC are proportional to each other, since AIC is defined by maximum likelihood and if our errors, ϵ for the model are Guassian, maximum likelihood and least squares are the same thing. Figure 1 shows the plot for C_p and BIC for the best subset selection method and Table 1 summarizes the results.

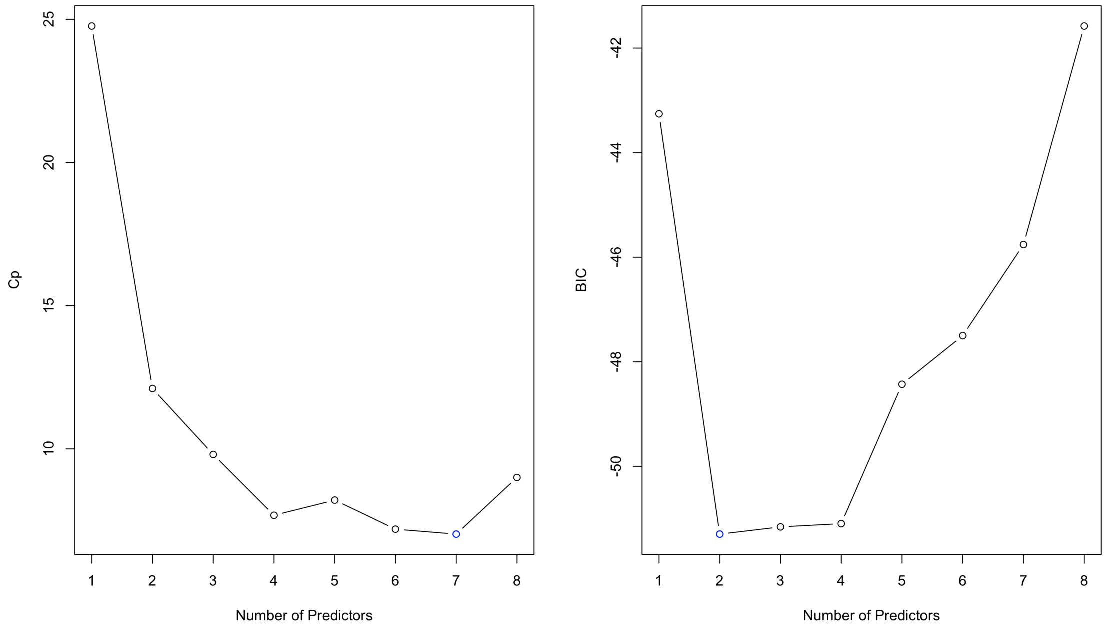


Figure 1: Plot for C_p and BIC for the best subset selection method

The prediction error(test set MSE) using the model having the predictors ”lcavol” and ”lweight” as suggested by the BIC statistic comes out to be **0.4925** while using the model suggested by C_p statistic(7

	Best Model (No. of Predictors)	Value
BIC	2	-51.29578
C_p	7	7.021515

Table 1: C_p and BIC values and the corresponding best models

predictors), comes out to be **0.5165**.

Figure 2 shows the plots of Adjusted Cross-validation error for different predictors for K=5 and K=10 and Table 2 summarizes the results.

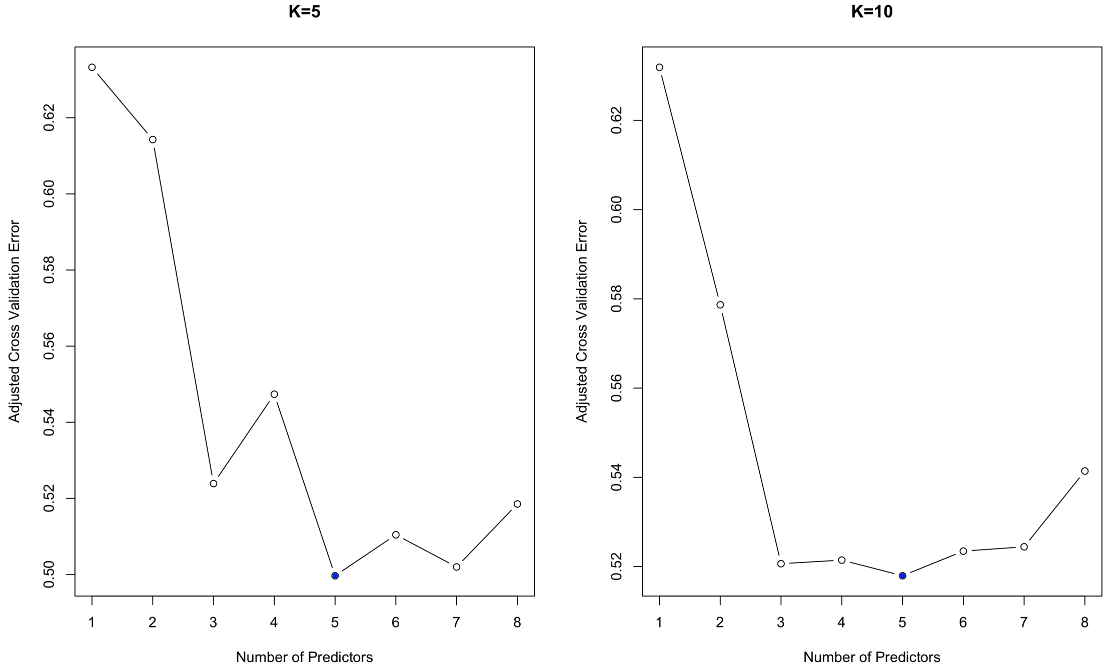


Figure 2: Plot for Adjusted Cross-validation error for different predictors for K=5 and K=10

K	No. of Predictors for minimum error	Adjusted CV error
5	5	0.4997
10	5	0.5179

Table 2: K-fold cross validation results

In, this case the number of predictors for K=5 and K=10 agree, which is 5.

For bootstrap 0.632, we get the minimum Bootstrap estimate of prediction error for 6 variables and the error value is **0.5163**. Figure 3 shows the Bootstrap estimate of Prediction error for different predictors.

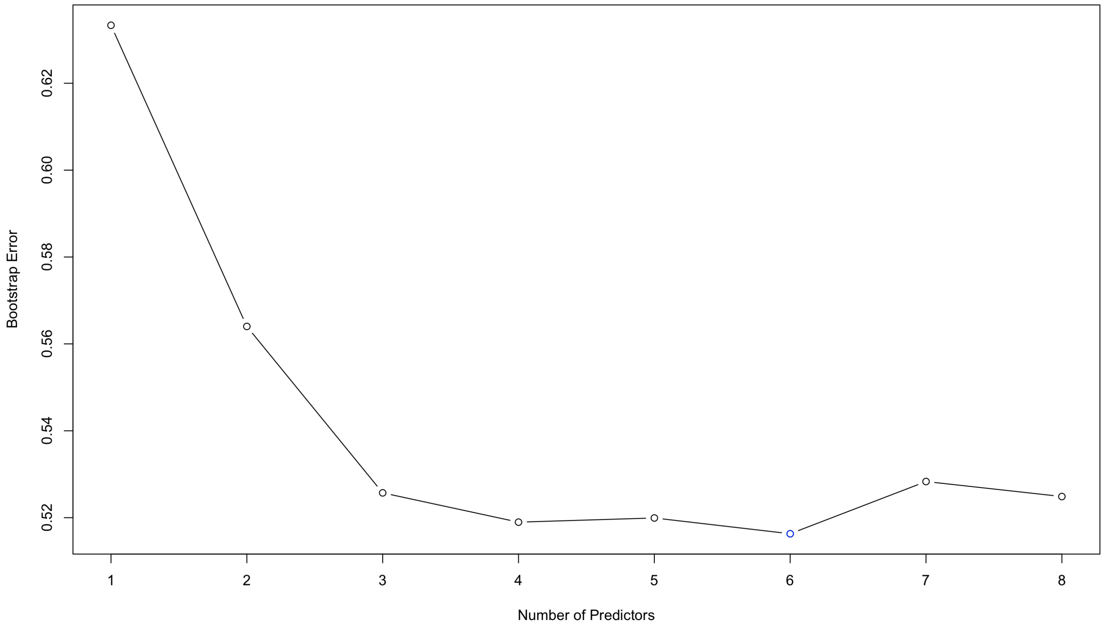


Figure 3: Plot for Bootstrap estimate of Prediction error for different predictors

Question 2: A access the wine data from the UCI machine learning repository. These data are the results of a chemical analysis of 178 wines grown over the decade 1970-1979 in the same region of Italy, but derived from three different cultivars (Barolo, Grignolino, Barbera). The Babera wines were predominately from a period that was much later than that of the Barolo and Grignolino wines. The analysis determined the quantities MalicAcid, Ash, AlcAsh, Mg, Phenols, Proa, Color, Hue, OD, and Proline. There are 50 Barolo wines, 71 Grignolino wines, and 48 Barbera wines. Construct the appropriate-size classification tree for this dataset. How many training and testing samples fall into each node? Describe the resulting tree and your approach.

The following columns are considered for analysis: MalicAcid, Ash, AlcAsh, Mg, Phenols, Proa, Color, Hue, OD, and Proline. The train and the test data are split in the ratio 7:3. A full classification tree is grown and the tree was pruned thereafter as per the minimum C_p value. Figure 4 shows the Classification tree on the training data before pruning.

Figure 5 shows the C_p values. The minimum value of C_p is at index = 4 which comes to be equal to 0.1667.

Figure 6 shows the Classification tree on the training data after pruning.

Clearly, the pruned tree has 6 terminal (leaf) nodes. The misclassification error rate for the training set comes out to be 0.024 while for the test set it comes out to be 0.0189. The number of training and testing samples that fall into each node are given in figure 7 and figure 8.

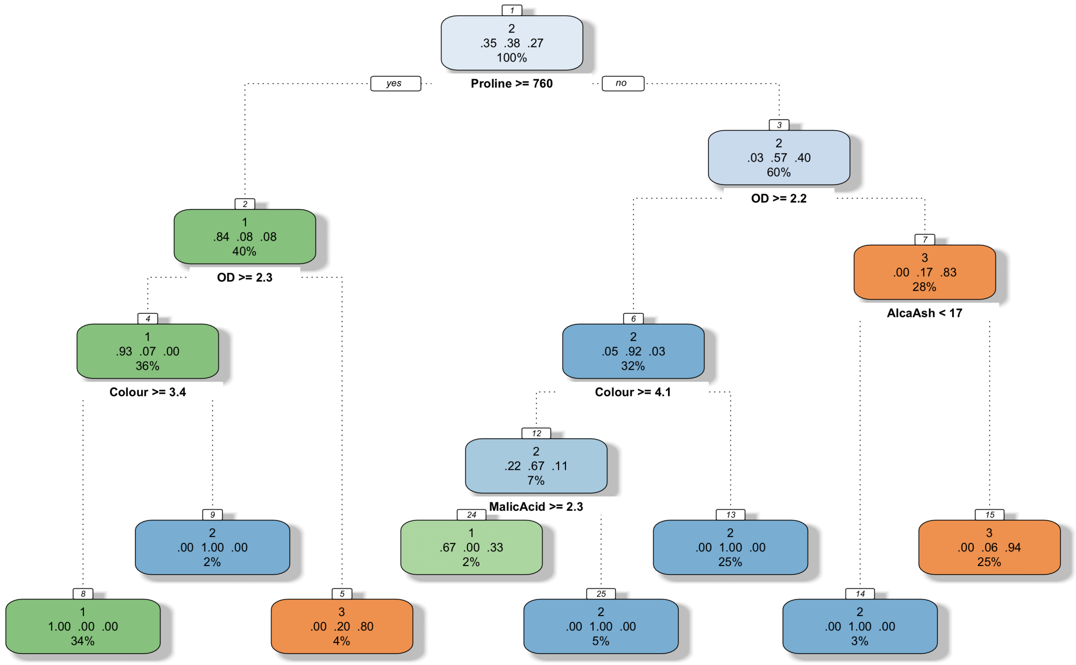


Figure 4: Classification tree on the training data before pruning

The resulting tree was split at 5 nodes and have 6 different terminal (leaf) nodes. The first split is done on 'Proline', second on 'OD' and so forth. The percentage in each box gives the percent of data which that each node has and splits. The split at each node is decided by the one which gives maximum reduction in the misclassification error. Also we see that the number of nodes for test and training set are different as it might be the case that in the test set we do not have observations corresponding to a particular split (split at node 9 in this case). Upon summing the number of elements in the terminal nodes for the train as well as test set we get 125 and 53 respectively which are the total number of observations the train and the test set have respectively.

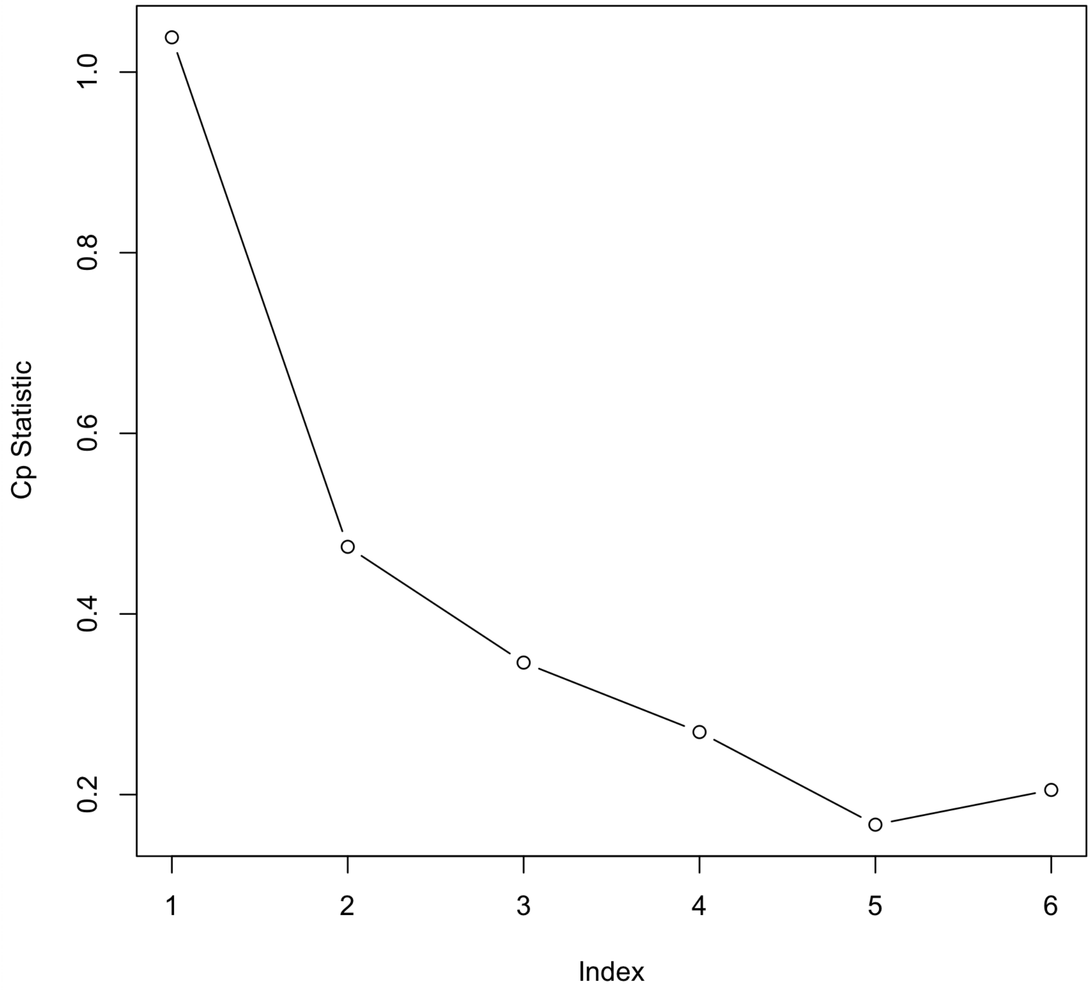


Figure 5: C_p values

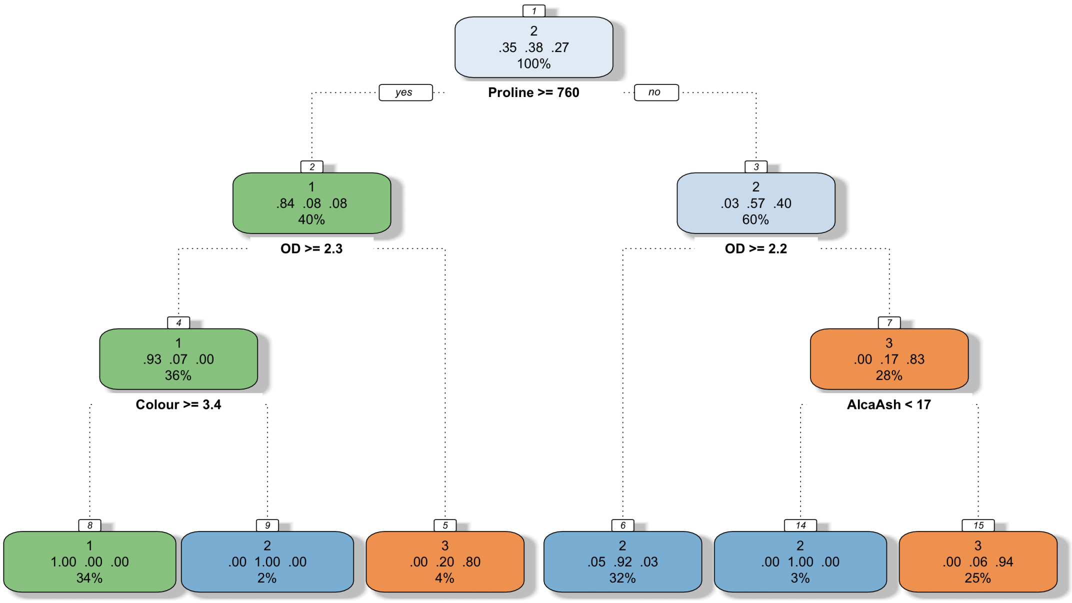
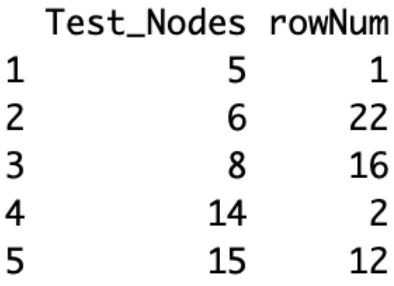
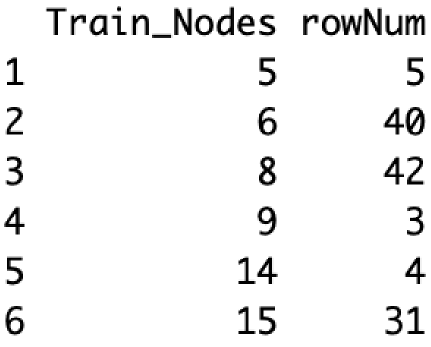


Figure 6: Classification tree on the training data after pruning



```
  Test_Nodes rowNum
1          5      1
2          6     22
3          8     16
4         14      2
5         15     12
```

Figure 7: Number of testing samples that fall into each node



```
  Train_Nodes rowNum
1           5      5
2           6     40
3           8     42
4           9      3
5          14      4
6          15     31
```

Figure 8: Number of training samples that fall into each node

Question 3: Apply bagging, boosting, and random forests to a data set of your choice (not one used in the committee machines labs). Fit the models on a training set, and evaluate them on a test set. How accurate are these results compared to more simplistic (non-ensemble) methods (e.g., logistic regression, kNN, etc)? What are some advantages (and disadvantages) do committee machines have related to the data set that you selected?

The dataset used in this question is Boston dataset in the MASS package. The train and the test data are split in the ratio 7:3. The variable crim has been converted to a qualitative variable with values 1 if the value is greater than the median value and 0 elsewhere.

The Boston data set contains the following predictors:

crim: per capita crime rate by town.

zn: proportion of residential land zoned for lots over 25,000 sq.ft.

indus: proportion of non-retail business acres per town.

chas: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).

nox: nitrogen oxides concentration (parts per 10 million).

rm: average number of rooms per dwelling.

age: proportion of owner-occupied units built prior to 1940.

dis: weighted mean of distances to five Boston employment centres.

rad: index of accessibility to radial highways.

tax: full-value property-tax rate per $ 10,000.

ptratio: pupil-teacher ratio by town.

black: $1000(B_k - 0.63)^2$ where B_k is the proportion of blacks by town.

lstat: lower status of the population (percent).

medv: median value of owner-occupied homes in $ 1000s.

Figure 9 shows the plot for Bagging: showing the significant predictors. Figure 10 shows the plot for Boosting: showing the significant predictors.

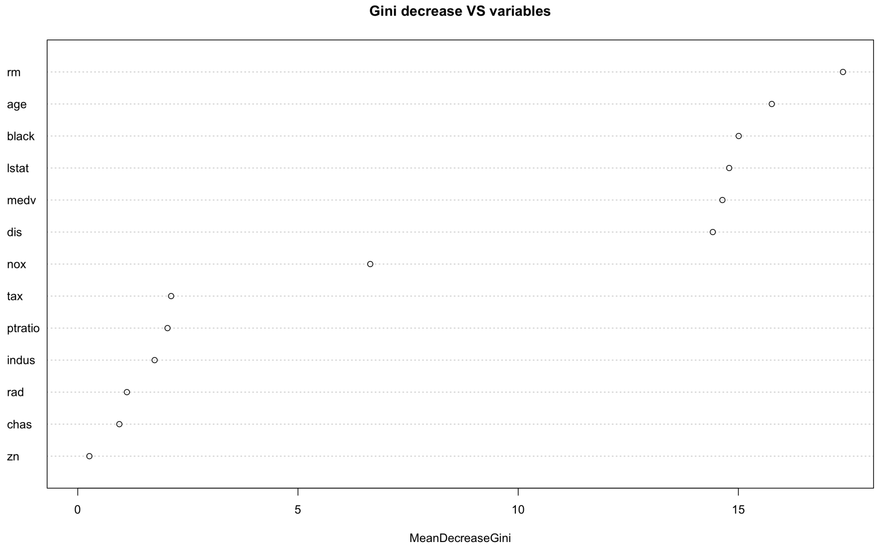


Figure 9: Plot for Bagging: showing the significant predictors

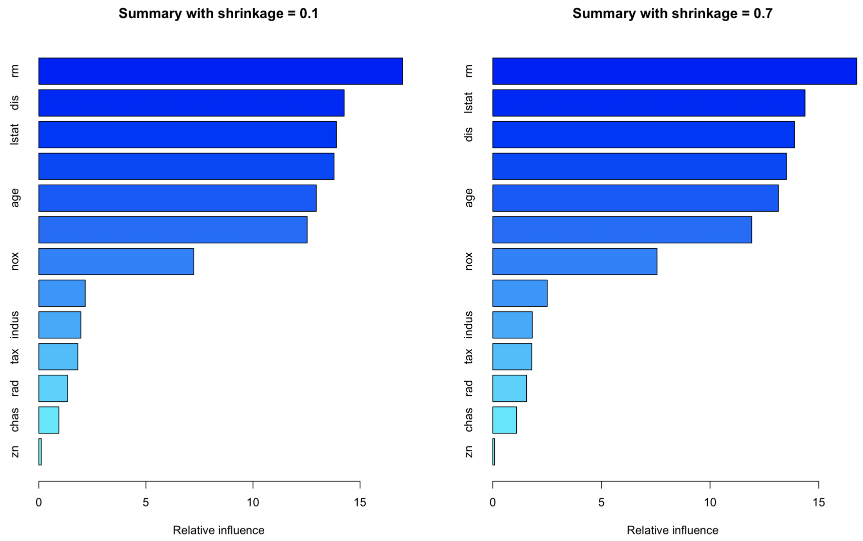


Figure 10: Plot for Boosting: showing the significant predictors

The test error obtained using Random Forest is 1.8816, using boosting is 0.789 for shrinkage = 0.1 and 0.828 for shrinkage = 0.7, using bagging: it is 0.855. The error for logistic regression and kNN comes out to be 1.5 and 0.5 respectively. We see that the ensemble methods have errors less than the more simplistic models in general.

The major advantage of using committee machines is that the error is reduced here due to averaging. There are also less chances of overfitting as compared to logistic regression and kNN. The framework for methods like Random forests are intuitive in terms of correlation and strength. Random Forests are also good for dealing with high dimensional space. Committee methods are effective for large number of predictors whereas kNN is not that effective. But all of these advantages comes at some cost, these are computationally expensive and less interpretable.