CS 7641: Assignment 1

Shaikh Shamid

February 2, 2015

Introduction

In this supervised machine learning project I choose two different datasets, one with binary classification and another with multi class classification. Both data sets are from the UCI machine learning repository. To get started the first set of data I choose is the Abalone data, the classification task here is to predict the age of abalone from physical measurements. We know that the Abalone is an excellent source of iron and pantothenic acid, which is a nutritious food resource and farming in Australia, America and East Asia. 100 grams of abalone yields more than 20% recommended daily intake of these nutrients. The economic value of abalone is positively correlated with its age. Therefore, to detect the age of abalone accurately is important for both farmers and customers to determine its price. However, the current technology to decide the age is quite costly and inefficient. Farmers usually cut the shells and count the rings through microscopes to estimate the age of the abalone. This complex method increases the cost and limits its popularity. Our target is to find out the best indicators to forecast the age of abalones.

The second data set is the Adult data set, the classification task here is to predict whether a person makes over $ 50K/yr based on census data taken in 1994 from many diverse demographics. The different methods I will be using to analyze these problems are K Nearest Neighbors, Support Vector Machines, a Neural Network, Boosting, and a pruned decision tree.

	income	Freq
1	high	5271
2	low	16005

Table 1: Binary class distribution of the income variable of Adult data set

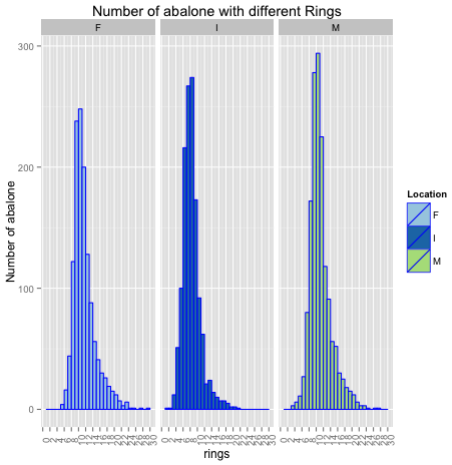


Figure 1: Abalone data distribution

Data acquisition

Both data sets come from the UCI Machine Learning repository. I did some preprocessing of both data sets before applying classifiers.

Abalone Data Set

The abalone data set has 4177 instances and 8 different attributes. If we look at the ring distribution for male, female and infant abalones given in Fig. 1, For infant we see that the maximum number of abalone occurs with smaller rings. So, we define three age groups: 0-8 rings as infant, 9-11 as adult and greater than 11 rings as old.

Adult Data Set

The adult data set has over 22000 instances and 14 features of mostly categorical variables. I did some preprocessing of the original data by discarding couple of repeated attributes and clean the categorical values by giving good names.

We see from the table 1 that the binary classes are imbalanced. So, in addition to Accuracy metric, ROC metric is also needed for this data set when we compare and contrast different algorithms.

Tools Used and Methodology

I use R and R-Studio to complete this project because it has some useful packages, especially the unified caret package that simplifies building classification problems. For each algorithm, I run single repeated 5-folds cross validation. The purpose of running cross validation first is to build models that would have sufficient predictive power when applied to unseen data.

K-Nearest Neighbors (KNN)

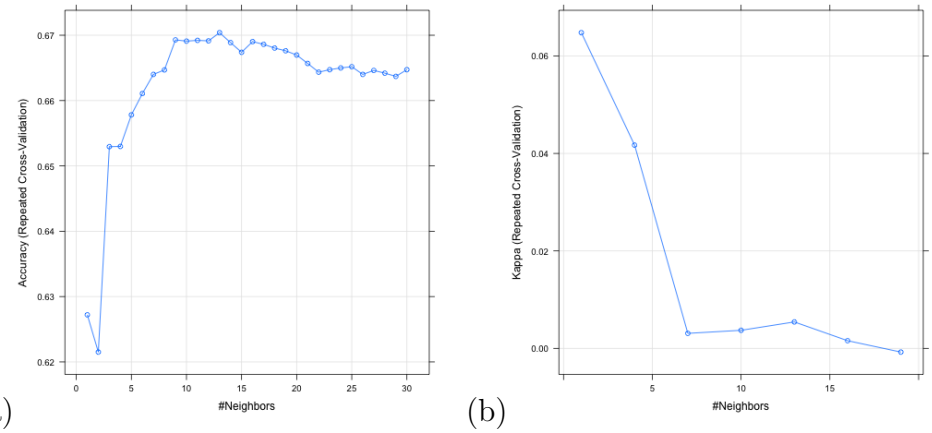
Given the nearest K points to a query point q from a given data set, KNN votes on what is the most common classification. The domain knowledge of K-Nearest Neighbor is what distance function you use, and what K. I would like to tune the K values using a 5-fold cross validation for numbers from 1 to 30 for abalone data set, and for numbers 1 to 20 in steps of 3 for adult data set. What we see is that KNN does really bad on adult data set because kappa gets negative after 5 neighbors, negative values indicate agreement less than chance. This is also reflected on the error plots for abalone data. However, there was a reasonable classification on abalone data, where I did three class classification with accuracy 67 %.

While KNN is a simplistic algorithm it did well with abalone classification. This is due to the fact that abalone's that are young are similar in nature. The adult data set did close to chance because income classification isn't as similarity based, also it has a lot of categorical variables.

	adult	old	young
adult	516	141	122
old	32	76	3
young	74	7	283

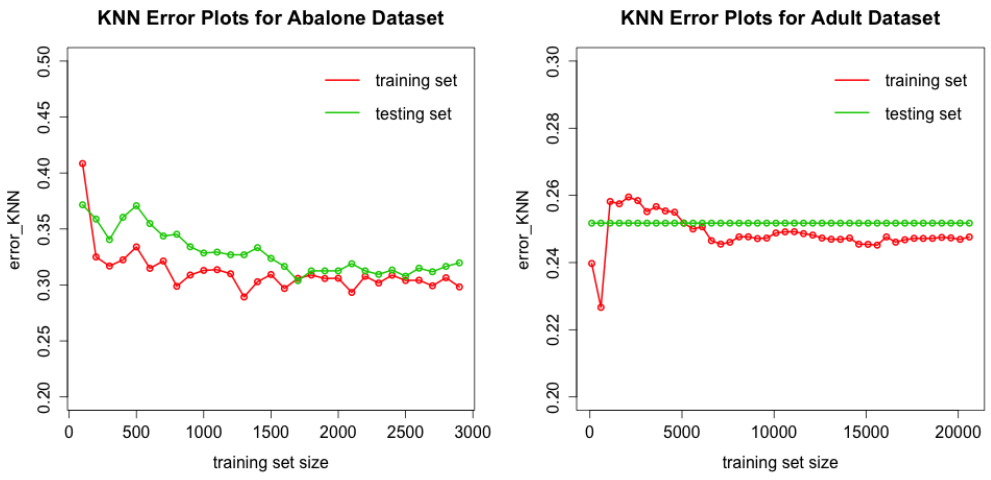
	high	low
high	404	1019
low	1833	5630

Table 2: Confusion Matrix for the KNN model



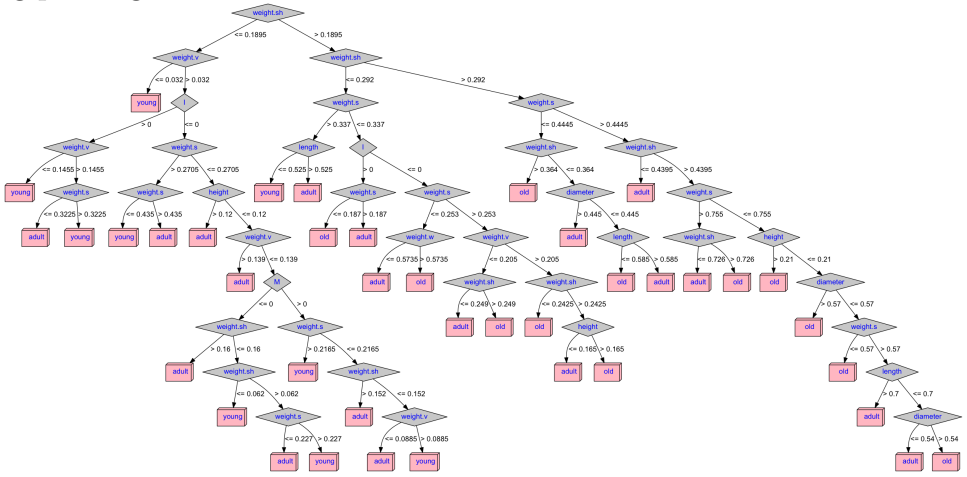
(a) (b)

Figure 2: Plot of (a) accuracy vs k for abalone data and (b) kappa vs k for adult data.

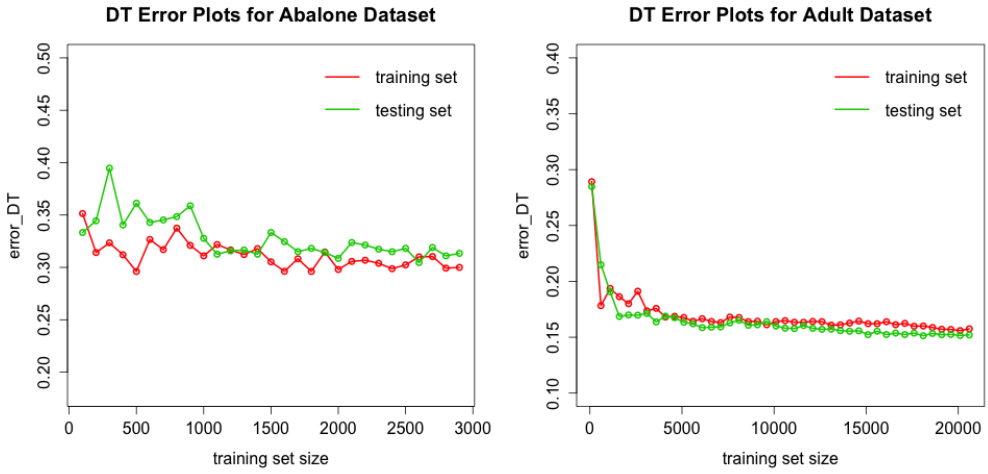


1 Decision trees (C5.0) with pruning

In the decision tree algorithm we can determine the most important variable for the classification problem from the data set by looking at the first split in the tree. Decision trees sometimes overfits the data that's why I will be using pruning method.



In order to apply decision tree algorithm I use C50 package which is a C5.0 implementation in C. Here C5.0 constructs decision trees in two phases. A large tree is first grown to fit the data closely and is then pruned by removing parts that are predicted to have a relatively high error rate. This pruning process is first applied to every subtree to decide whether it should be replaced by a leaf or sub-branch, and then at global stage looks at the performance of the tree as a whole. For abalone data without global pruning the training errors was approximately 30 % and the tree size was 48. with global pruning tree size reduces to 40 (as shown in the tree plot) but the training error increases a little bit. However, the test accuracy (0.67) and Kappa value (0.41) stays the same. So I decided to go with global pruning. The hypothesis set are the minimum number of objects in each leaf and the confidence factor. For these I have tuned them to get the confidence factor 0.05 and min_num_obj 2. I was surprised to see that the variable weight.sh is by far the most important attribute and so on. This tree is useful in a sense that you can screen out attributes by looking at the most important one. Adult dataset performed ok with about 85 % accuracy, but the error plot shows that both the training and testing errors decrease slowly and never gets the plateau. It seems to me that there is a room for improvement for



this data set.

Decision trees seem to be very well suited for some problems and not for others. In the case of abalone data there is a lot of continuous variables that seem to perform poorly. The benefit here is that it is relatively fast and easy to see what determines what.

	adult	old	young
adult	486	135	122
old	54	80	3
young	82	9	283

	high	low
high	1354	477
low	883	6172

Table 3: Confusion Matrix for the DT model

Boosting (GBM)

The generalized boosted method determines the best fit through iterations. The downside to AdaBoost is using too many iterations and overfitting the data, also it doesn't require a lot of tuning like other algorithms, and most of the work can be done by setting how many iterations we want to go through. The gbm package implements extensions to Freund and Schapire's AdaBoost algorithm and Friedman's gradient boosting machine.

I decided to tune the model for iteration from 1 to 100 and shrinkage parameter [0.05,0.1,0.15] and keeping the interaction depth at 3 for abalone

	adult	old	young
adult	503	125	111
old	42	94	2
young	77	5	295

	high	low
high	1300	391
low	937	6258

Table 4: Confusion Matrix for the boosting model

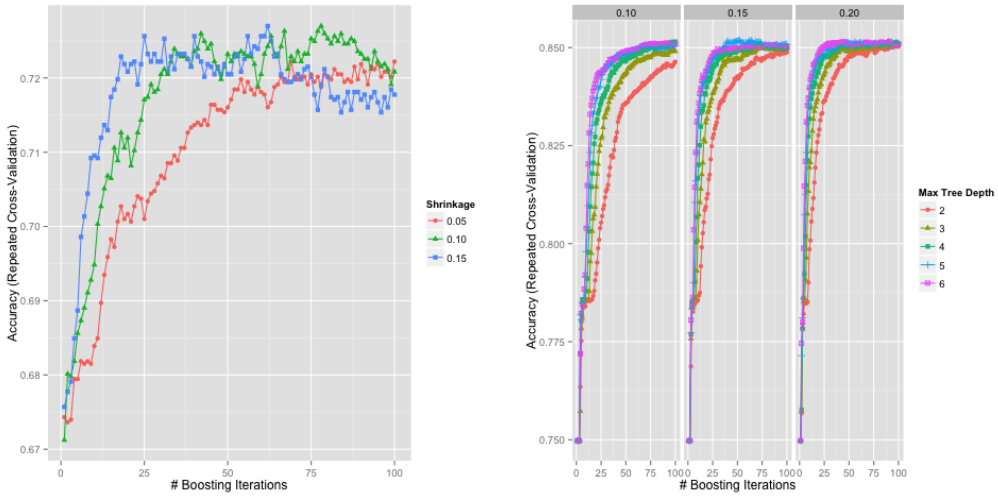
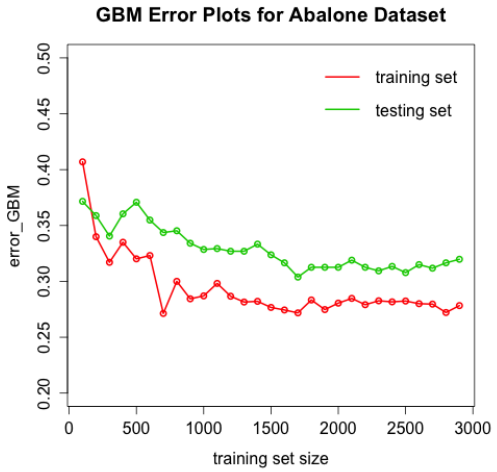


Figure 3: Parameter tuning in boosted tree (a) for abalone data (b) for adult data.

data, which gives a decent hypothesis set for GBM with iteration number = 78, interaction.depth = 3 and shrinkage = 0.1. Running this against yields good answers in a quick period of time. For this small data set, the testing error starts as high error but goes down as we add more data, at the end we have a big gap between the training-set error and the testing-set error, and the curves do not level out, so it seems that if we add more data the testing error would keep going down. So I identify it suffers High Variance (overfits) because the testing error is much larger than the training error, adding more data is likely to help.

However, the adult data set worked pretty well against this algorithm. The final values used for the model were iteration number = 51, interaction.depth = 5 and shrinkage = 0.15. The error curves level out quickly. Also from fig 7 we see that the area under the ROC curve is the greatest for this algorithm.



Neural Networks (NN)

The package I will be using is R nnet package, which lets you construct standard ANNs with one hidden layer of sigmoid function neurons with back propagation algorithm. Even with those choices fixed, there are still couple of parameters to tweak. I decided to tune the model for hidden units [4, 5, 6] and weight decay from 0.001 to 0.01 in steps of 0.001 for abalone data. With highest ROC (0.9) the final values used for the model were hidden layer = 4 and weight decay = 0.006. The error plots look ok, the testing error starts as high error but goes down as we add more data. For adult data

	adult	old	young
adult	500	119	103
old	44	102	3
young	78	3	302

	high	low
high	1117	355
low	1120	6294

Table 5: Confusion Matrix for the NN model

I decided to tune the model for hidden units [2,3,4] and weight decay from 0.005 to 0.02 in steps of 0.0025. With highest ROC (0.9) the final values used for the model were hidden layer = 4 and weight decay = 0.01. The error curves get plateau quickly. The neural network did perform well (2nd best after boosting) for this data set. Also from fig 7 we see that the area under the ROC curve is the 2nd best for this algorithm. NN performed best

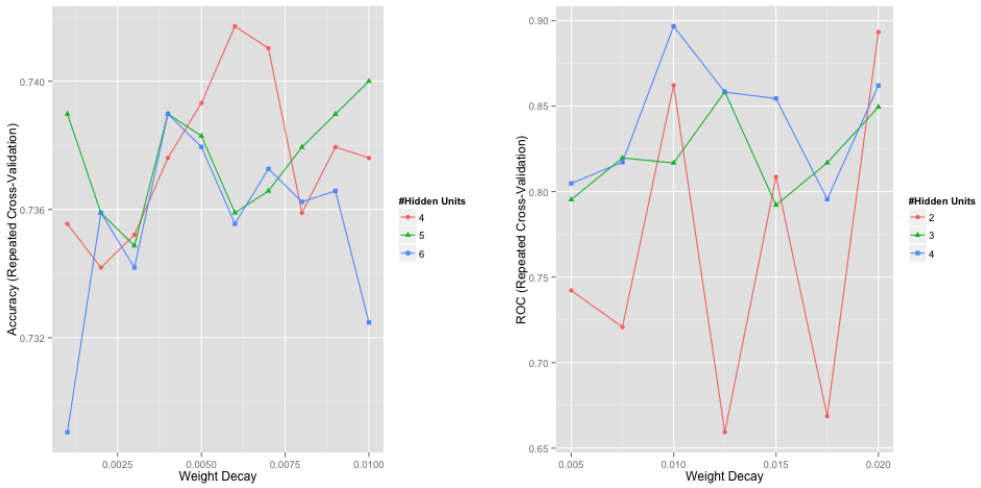
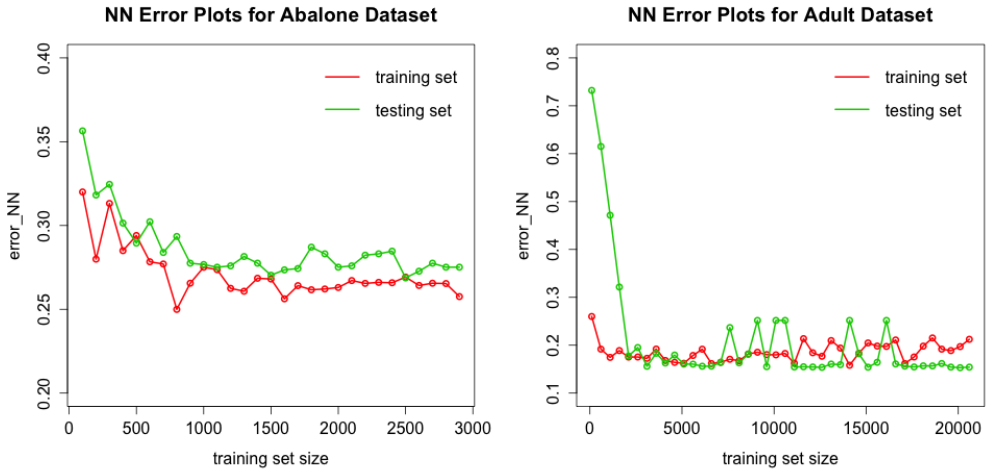


Figure 4: NN parameter tuning (a) for abalone data (b) for adult data.

for abalone data compared to all other algorithms. This is because NN are good at outputting lots of nodes, and for a problem like this one classifying across many different categories, the complexity of the model outperforms all the other algorithms applied on the same data.

Support Vector Machines (SVM)

SVM represents a powerful technique for general (nonlinear) classification with an intuitive model representation. The R package e1071 offers an interface to the award-winning C++ implementation by Chih-Chung Chang and Chih-Jen Lin. It uses a simple quadratic programming calculation and introduces different useful kernels option and different cost parameters. The cost parameter introduces slack into the original optimization problem and Kernels maps the original data set into a new mathematical space in which the decision boundary is easy to describe. In order to find the best SVM classifiers that separates hyperplane by maximizing the margin for our datasets I tested three kernel functions (linear, polynomial, and radial). Fig 5 shows that the linear kernel is the best for both data sets. Then I introduce grid search to find the best cost parameter, due to space limitation I could not attached that plot, but what I found is C=2 for abalone and C=4 for adult data are the best cost parameters. The error curves look good. The testing



error starts as high error but goes down as train size increases, at the end the curves do level out for both the data set. SVM did well because it avoids over-fitting via regularization parameters.

	adult	old	young
adult	513	136	111
old	29	81	0
young	80	7	297

	high	low
high	1115	371
low	1122	6278

Table 6: Confusion martrix for SVM model

Conclusion

From Fig 6 and 7 we see that, for abalone data NN algorithm does the best with highest accuracy compared to all other algorithms and for adult data Boosting algorithm yields the best performance in terms of area under the ROC curve. Things did not work as well for the abalone data set which seems to have a lot of noise in it. We learn that the supervised learning methods work well for data sets that have an underlining distribution, but when we compare and contrast models, the differences in their performance can be partly attributable not to their differing structure, but to the different levels of tuning effort we invest in them. Also the types of algorithms that work best are problem-specific.

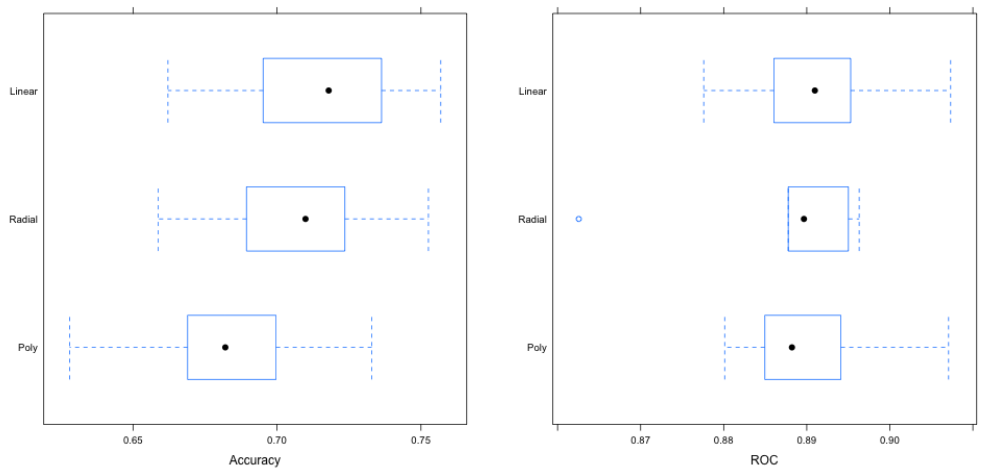
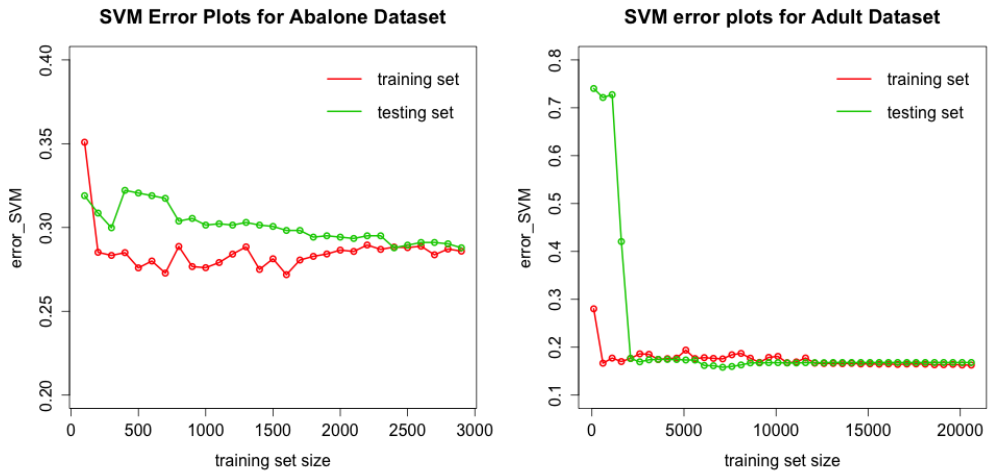


Figure 5: Testing different kernels (a) for abalone data (b) for adult data.



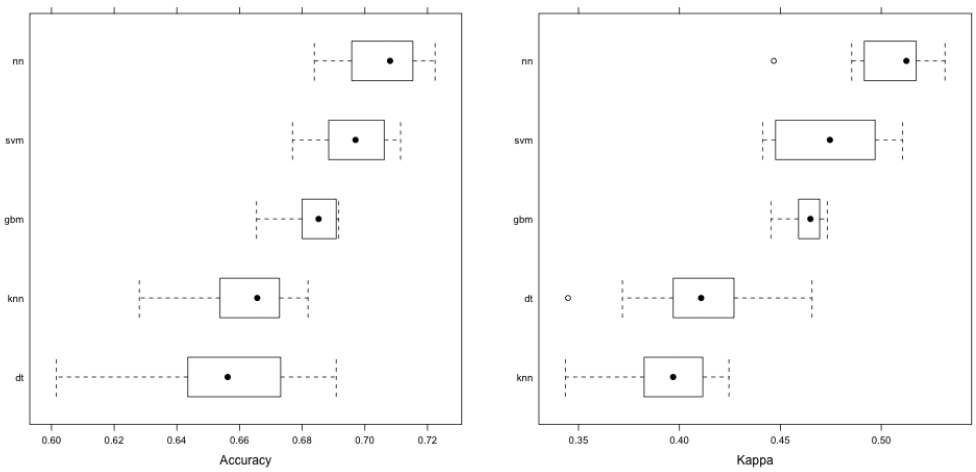


Figure 6: Accuracy and Kappa comparison among models for abalone data

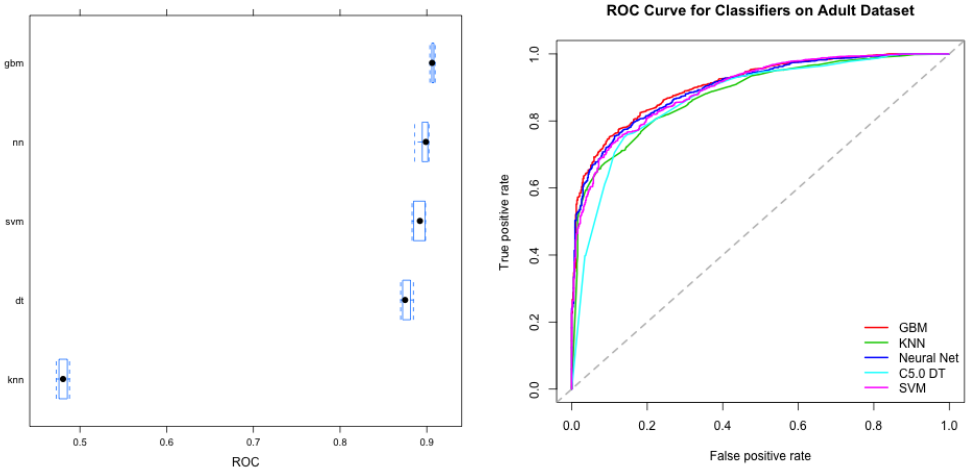


Figure 7: ROC comparison for different models for adult data set.