CS7641 Machine Learning First Assignment

T. Ruzmetov

September 20, 2017

Introduction

Purpose of this assignment is to investigate and compare the performance of five different machine learning techniques such as K-Nearest Neighbor, Decision Trees with pruning, Boosted Derision Trees, Support Vector Machines and Artificial Neural Networks by applying them to two distinct classification problems(distinct data sets). In order to fulfill the requirement, "Lending Club Data set" from Kaggle and Adult Data set from UCI machine learning repository are used. For all methods on both data sets cross validation is performed in order to find optimum hyper parameter. Learning curve experiment is done by altering training data size and monitoring performance via prediction accuracy.

Lending Club Dataset

Lending Club is the world's largest online marketplace connecting borrowers and investors. Getting loan has become common practice in developed countries, which makes it interesting to learn how banks determine customer eligibility or credit worthiness given some information. In this problem, we apply unsupervised learning methods to predict if customer is going to pay or default on their loan based on provided set of features with labels. Provided data set contain complete loan data for all loans issued through 2007 to 2015, including the current loan status(Current, Late, Fully Paid, etc.). Instances with "Current" loan status are discarded to make it simple binary classification problem with two labels(unpaid,paid). After cleaning and feature extraction, the data set contains 15 features and 285373 samples. 90% of entire data is allocated to training set and 10% is separated out for testing. Then only only 20% of the training set is used for model training due to large size resulting in high time consumption.

Adult Dataset

The adult data set contains census information from 1994. Our task is to predict whether a person makes more than \$50K/year. After preprocessing is applied, there are 11 features and 45000 remaining instances. The target feature "income" is labeled as "high" when income is greater than \$50K and "low" when it is less.

Methods and Tools

R-Studio and R are used for both coding and project writing. R-studio has natively built in markdown + latex + html via pandoc, that I used for project writing. Caret package in R provides really nicely build in libraries for a lot of machine learning techniques and very user friendly implementation of parallel computing for cross validation via hyper parameter tuning. For all ML methods, I used 5-fold cross validation with repetitions(2-5) in order to find parameters that correspond to maximum cross validation accuracy. For growing and pruning the tree "rpart" package is used.

K-Nearest Naighbour

Knn algorithm classifies a data points based on its K closest neighbors in distance, where over represented class within K will get the vote. K value and distance metric are the only parameters to tune for cross validation. It is slow with large data sets and high dimensional data. Also categorical features don't work well. 5-fold twice repeated cross validation by changing k value from 5 to 43 with step size 2 gave $k_{best} = 43$ for Lending Club data set and $k_{best} = 27$ for Adult Data set as depicted in Fig.1. This values of k correspond to maximum accuracy. Then, using the best model estimated accuracies on test set are 73.1% for LC data and 83.1% for Adult Data.

Although Adult Data set contain mostly categorical features it performed better than Lending Club Data set. Some features such "country", "education number" and "fnlwgt" are removed!

Learning curve results for LC Data set Fig.2 show decrease in training accuracy as training

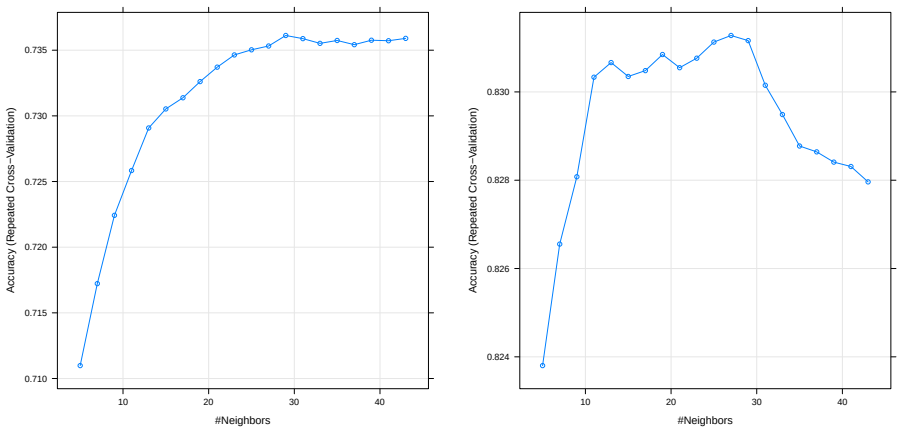


Figure 1: Cross validation plots for KNN model

Table 1: KNN confusion matrix for Lending Club(left) and Adult data(right)

	paid	unpaid
paid	19895	6808
unpaid	862	972

	high	low
high	2211	1065
low	1489	10295

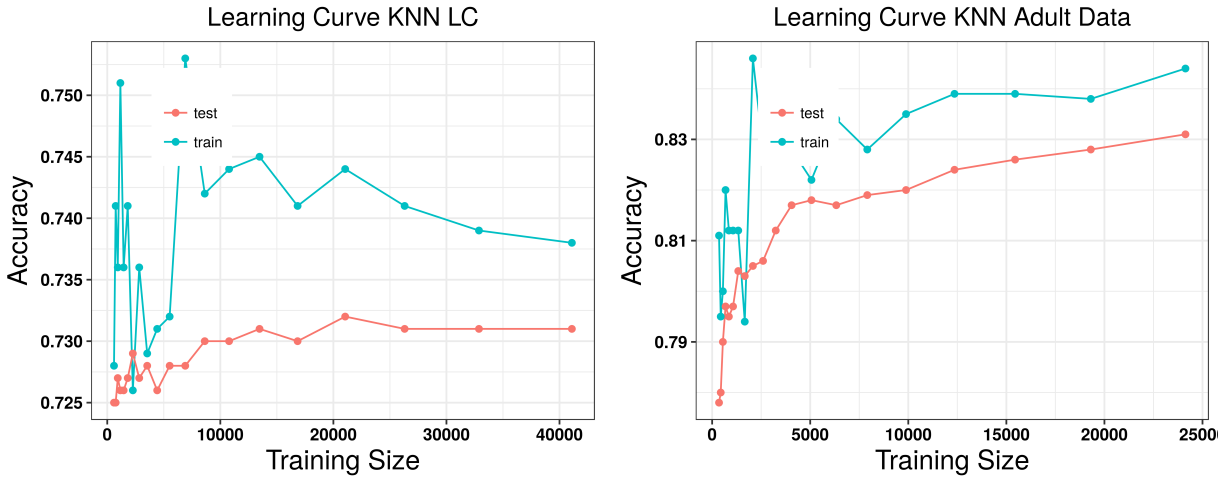
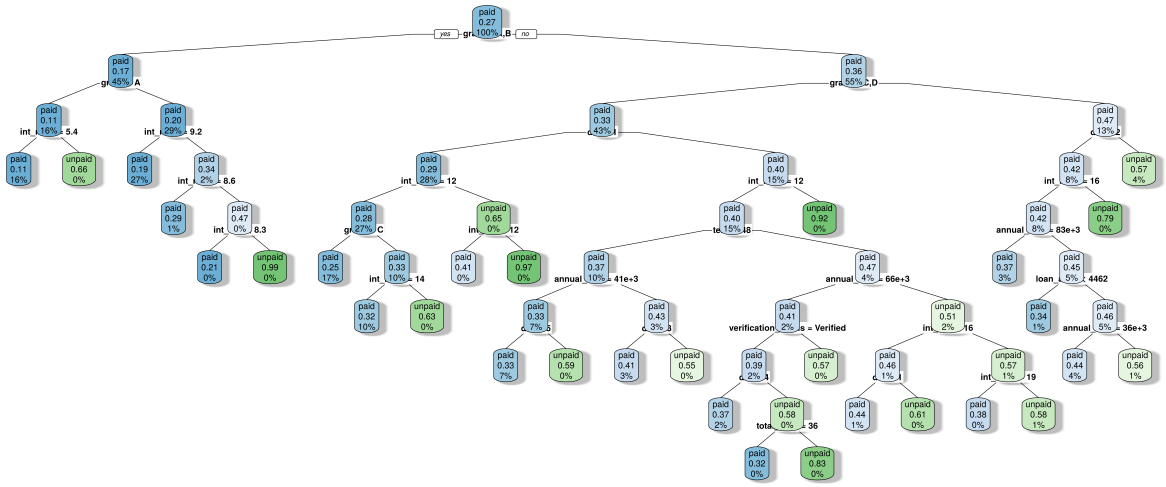


Figure 2: Learning curve plots for KNN model



Pruned Tree Diagram LC

size grow, but testing accuracy don't change much after it reaches 20000 sample size. On the other hand, for Adult Data both training and testing accuracies increase with data size meaning on could add more data to get better accuracy on unseen data.

Decision Tree Algorithm with Pruning

Decision tree classifier is used with maximum information gain to determine which feature is given a priority for splitting. Post pruning is used to prevent overfitting. Tree is grown to a large size (1000-6200) and complexity parameter with least x-value error is chosen with corresponding optimum size of tree to proceed further. Number of splits(tree size) is reduced to from 6200 to 29(cp=0.00059) for Lending Club data and from 2284 to 82(cp=0.00044) for Adult Data respectively. For both data sets accuracy of training set reduced due to pruning, while accuracy of testing set is improved as expected. Effect of pruning on test set is more pronounced for Lending Club Data set showing around ~8% increase, while Adult Data set show ~3% improvement. Pre and post pruning results are summarized in table2 via accuracy and kappa value metric.

Learning curve results for decision tree method for both data sets form a plateau at about 70% of training data size for Lending Club Data and at around 80% of training size of Adult

Table 2: Decision Trees pre and post pruning performance tabulated for Lending Club(left) and Adult data(right)

	Accuracy	Kappa
ori_test	0.6559204	0.1281274
ori_train	0.9460126	0.8617475
pruned_test	0.7387952	0.1509791
pruned_train	0.7394718	0.1548779

	Accuracy	Kappa
ori_test	0.8121514	0.4855458
ori_train	0.9297238	0.8091243
pruned_test	0.8456175	0.5647173
pruned_train	0.8654144	0.6270364

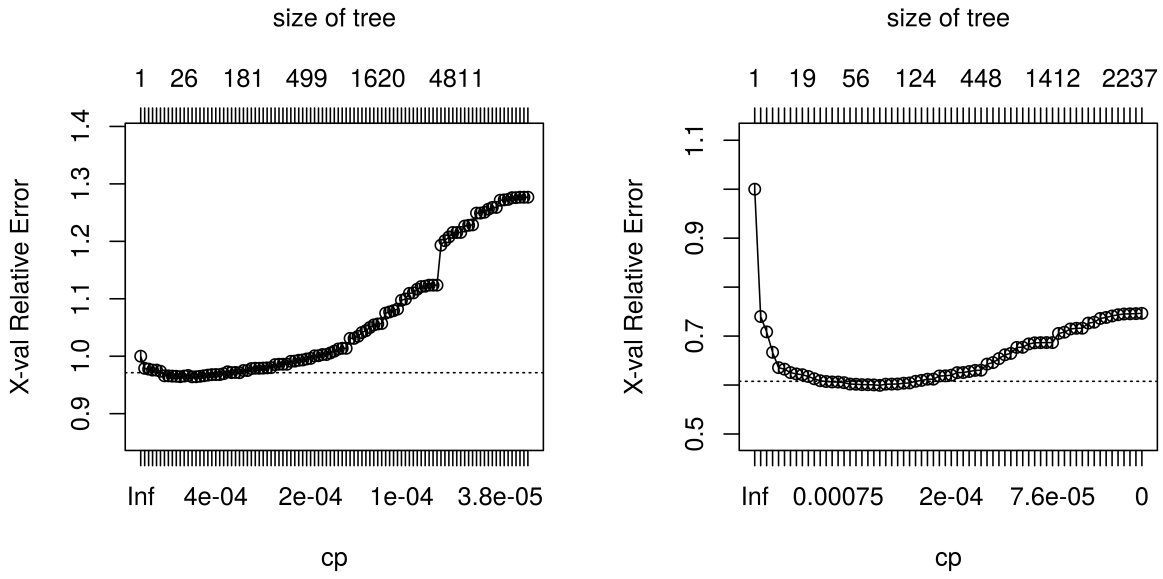


Figure 3: Cross validation plots for Tree model

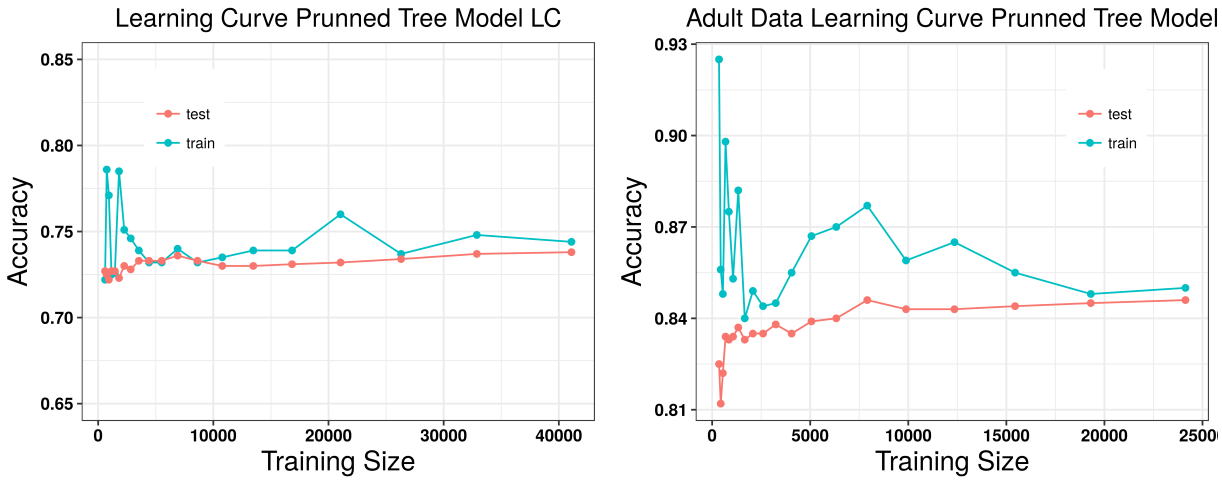


Figure 4: Learning curve plots for tree model

Table 3: Pruned Decision Trees confusion matrix for Lending Club(left) and Adult data(right)

	paid	unpaid
paid	19821	6518
unpaid	936	1262

	high	low
high	2301	926
low	1399	10434

Data set. This suggests that both data sets have more than enough amount of training data for model to converge to optimum performance. For learning curve iterations by changing training set size I performed post pruning for every iteration. For that reason a learning curve for Lending Club Data set looks different than all other learning curves done by different models. More specifically rest of the models have a gap between training and testing error after convergence except decision tree model.

Gradient Boosting

For Decision Tree Boosting the 'gbm' method is used under caret package. It implements extensions to Freund and Schapire's AdaBoost algorithm and Friedman's gradient boosting machine. Repeated cross validation is performed by tuning complexity of the tree(interaction.depth) and learning rate(shrinkage) for 50 iterations with step size 2. All values for above mentioned hyper parameters are shown in Fig.5. Optimum values chosen for Lending Club Data n.trees = 100, interaction.depth=9, shrinkage=0.15 ,and for Adult Data: n.trees = 80, interaction.depth=5, shrinkage=0.15. The minimum number of of training set

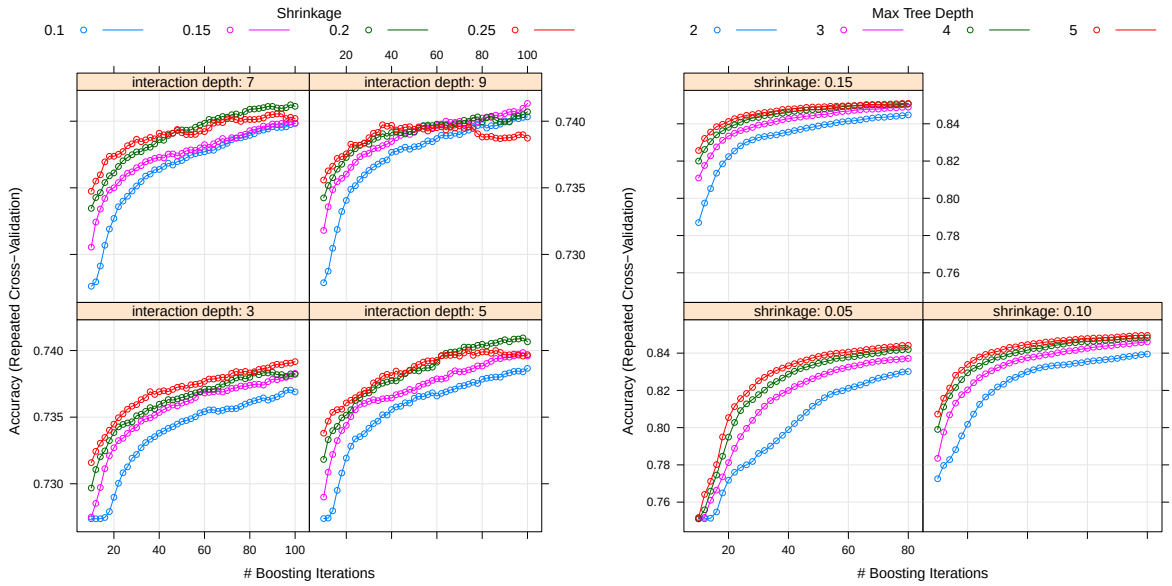


Figure 5: Cross validation plots for boosted tree

Table 4: Boosting confusion matrix for Lending Club(left) and Adult Data(right)

	paid	unpaid
paid	19730	6336
unpaid	1027	1444

	high	low
high	2249	759
low	1451	10601

samples in node to commence splitting is kept constant at 20.

Learning curves for Lending Club Data set show that both training and testing accuracies converge at training size 20000 having big gap in between. It is likely due to model suffering from overfitting. Also for this data set we can see that cross validation curves did not yet reach plateau. I think adding more iterations could result in better performance. On the other hand, Adult Data set performed very good achieving ~85% accuracy over testing data set. For a given number of iterations(n.trees) cross validation curves seem to have converged. Learning curves converge quickly as training size grow having almost no gap in between training an testing curves. This suggests that there is a good balance between bias and variance.

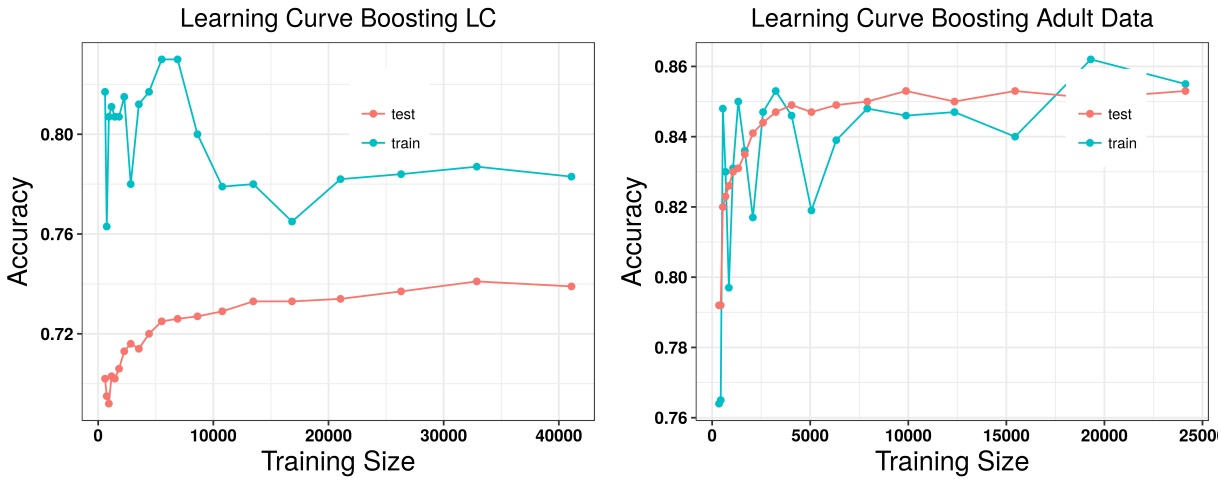


Figure 6: Learning curve plots for tree model

Neural Networks

"nnet" function is used through "caret" package in order to construct simple neural network model with single hidden layer of sigmoid activation function neurons. It uses backpropagation algorithm to minimize training error. Optimization parameters are number of hidden units and weight decay. Parameter set chosen for both data sets are displayed in Fig.7. Best parameters for Lending Club Data set are n.hidden.unit=7, weight.decay=0.08, where for Adult Data set n.hidden.units=5, weight.decay=0.01. Cross validation curves represented via ROC type accuracy show that there is a room for further improvement for Lending Club Data set because for all values of weight decay parameter ROC value is making upward progress as we increase number of hidden units. In contrast, plot shown for Adult Data set have very close ROC value for hidden unit number 4 and 5 which gives more reliable optimum parameters.

Again Adult Data winning over the Lending Club Data set by illustrating 10% better accuracy on testing set and %7 on training set. Training and testing accuracies for both data sets reach plateau fast suggesting 10000 data points could be sufficient to train.

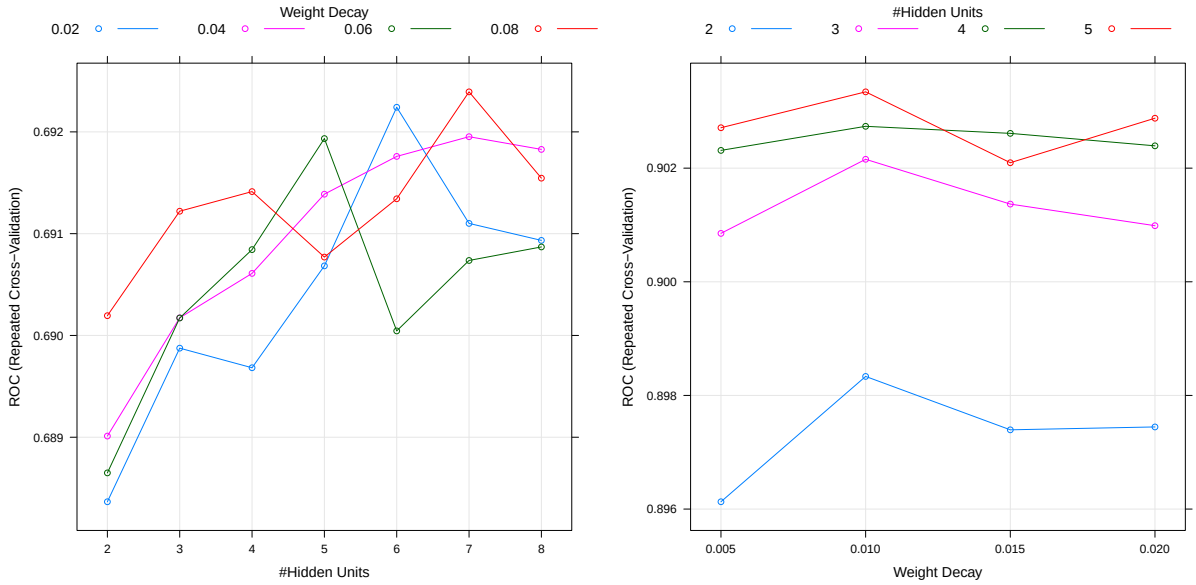


Figure 7: Cross validation plots for NNet

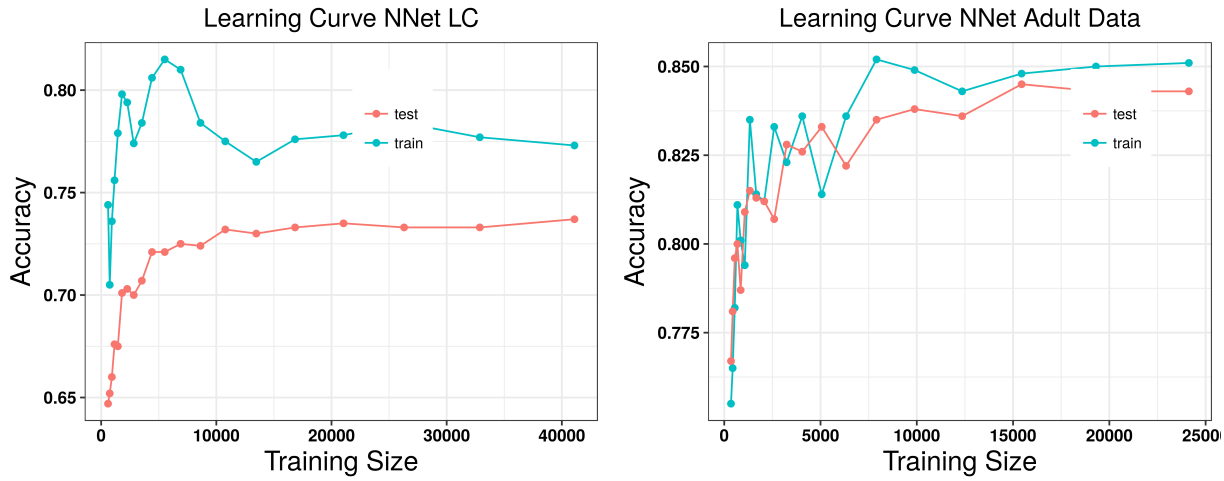


Figure 8: Learning curve plots for tree model

Table 5: NNet confusion matrix for Lending Club(left) and Adult Data(right)

	paid	unpaid
paid	19730	6336
unpaid	1027	1444

	high	low
high	2249	759
low	1451	10601

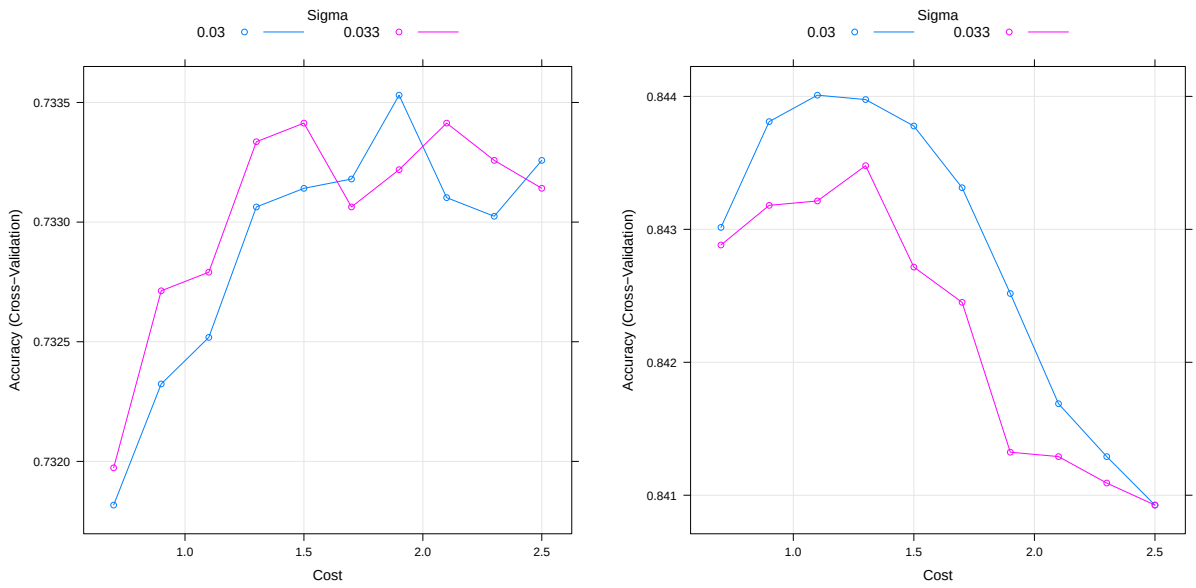


Figure 9: Cross validation plots for SVM

Support Vector Machines

Support Vector Machine is unsupervised learning algorithm mostly used for classification problems. It classifies instances by choosing an optimum decision boundary(can be both linear and non-linear) with maximum margin, where length of the margin is half distance between two support vectors. Here I applied SVM classifier for both data sets using two different kernels: Linear and Radial. Tried polynomial one too, but since training was very time consuming decided to drop that one. For each choice of kernel, I performed 5 fold cross validation, Fig.9, to choose optimum hyper parameters. On Lending Club Data set radial kernel(acc=73.4%) performed a little better than linear kernel(72.8%), but performance of different kernels on Adult Data set was very close. So I decided to go further with radial kernel Fig.11.

A surprising result I detected with SVM radial kernel classifier is that learning curves Fig.10 for training and test sets in Adult Data set did not even come close. They both seem to have reached the plateau, but there is a gap, which is different than behavior of all other models. I think it has to do with Adult Data set having too many categorical features since SVM uses distance metric.

Table 6: SVM RBB confusion matrix for Lending Club(left) and Adult Data(right)

	paid	unpaid
paid	20642	7645
unpaid	115	135

	high	low
high	2062	708
low	1638	10652

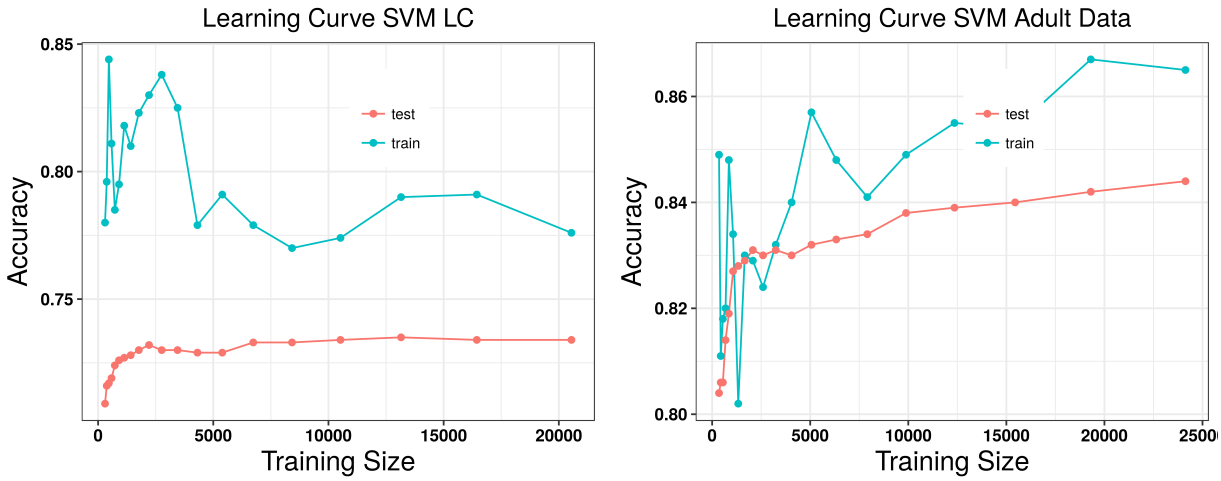


Figure 10: Learning curve plots for SVM model for Lending Club(left) and Adult Data(right)

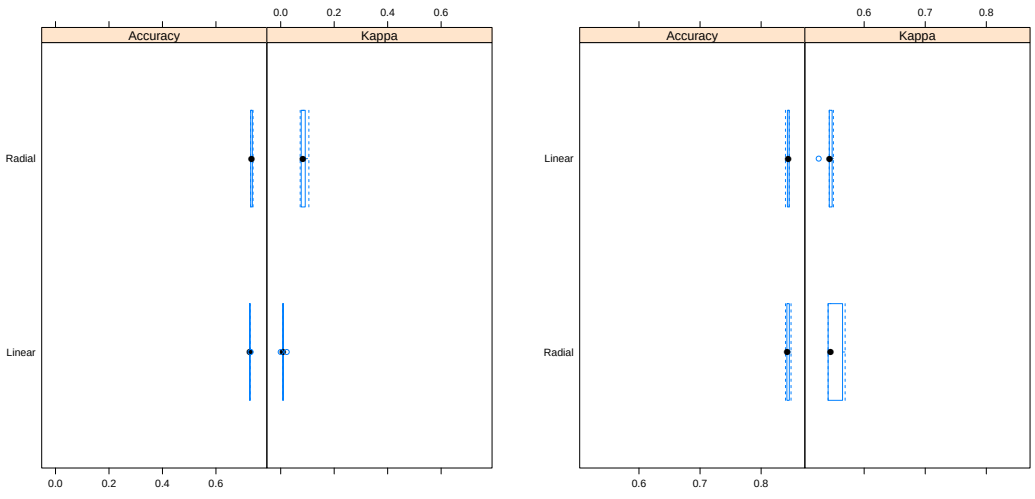
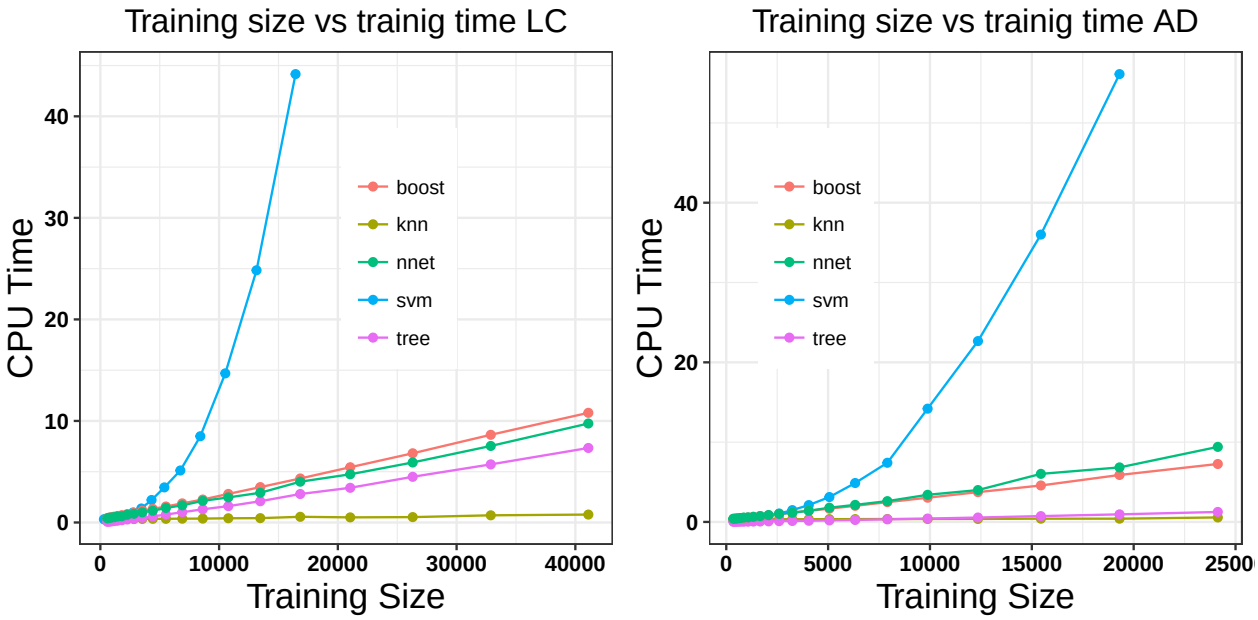


Figure 11: Comparison between linear and radial Kernels for SVM model for Lending Club(left) and Adult Data(right)

Table 7: Accuracy Report Across Models and Data Sets

model	Accuracy LC	Accuracy AD
boost	0.741	0.853
knn	0.732	0.831
nnet	0.737	0.845
svm	0.735	0.842
tree	0.738	0.846



Conclusion

Gradient Boosting was the best model compared to all other algorithms for both data sets reaching the accuracy of 74.2% for Lending Club Data set and 85.3% for Adult Data set. This is actually not so surprising because combination of many weak learners should naturally result in better prediction for specific type of data sets. Looks like both data sets have this property. Pruned decision tree performed as second best (LC.acc=73.8%, Ad.acc=84.6%). Lending Club Data did not perform as good as Adult Data set supposedly due to noise, and it needs more and careful feature engineering. Training data size vs training clock time plots show that computation time is linear with data size except SVM model, which showed exponential increase in CPU time. I learned that models are problem specific and when we compare them, 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.