Assignment 1: Supervised Learning

Shailesh Tappe: stappe3@gatech.edu

Note: ***I'm retaking course for this semester, I'm using most part of my previous semester (Spring 2020) report in this report. This time I'm re-exploring same datasets that I used previously, and trying to analyze more for this semester.***

Objective: The objective of the assignment is to analyze and compare different supervised learning algorithm with two different datasets. An analysis is performed on each learning algorithm i.e. Decision Tree, Neural Network, boosting, Supported Vector Machine, k-Nearest Neighbor to measure model's complexity and learning curve.

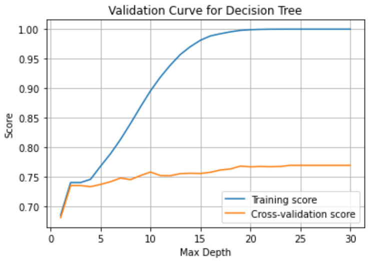
Datasets: The two datasets used for the analysis are Wine Quality dataset and Wisconsin Breast Cancer dataset from UCI library and openml.org dataset repository.

1. **Wine Quality dataset:** The wine quality datasets are variant of Portuguese 'Vinho Verde' wine, sample taken from a study performed by University of Minho in Portugal. Data contains 6497 instances and 12 attributes with classification attribute "*quality*". In preprocessing quality rating of wine is set to low quality (i.e. 0 for rating less than 6) and high quality (i.e. 1 for rating 6 or above).

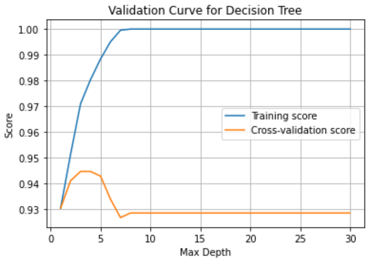
2. **Wisconsin Breast Cancer dataset:** The data was originally created by Dr. William Wolberg from University of Wisconsin Hospital in Madison WI. Data features are from digitalize image of FNA (fine needle aspirant), and represent feature characteristic of cell nuclei presented in the image. Data contains 699 instances with classification attribute "*Class*". Some of the data values are missing from datasets (i.e. marked as '?' instead of numeric value for attribute). In preprocessing all missing values are replaces with propagated fill forward values using python panda library.

Experiments: The experiments were performed to analyze performance of different supervised learning algorithm. Experiment was developed in python and scikit-learn library to analyze supervised learning algorithm. Each dataset was randomly split with ration of 80:20 for training and testing sets. These training and testing datasets are then validated with supervised learning algorithms to analyze accuracy, tuned up hyperparameters and cross validated for different training sets.

1. **Decision Tree Algorithm:** Decision tree classifier was experimented with Gini impurity index to split classification node at every instance. Gini impurity index helps to measure likelihood of incorrect classification of a random variable of new instance. To optimize decision tree, pruning technique is used to reduce incorrect classification of variable (overfitting), a common problem in decision tree. Decision tree is more prone to overfitting i.e. having incorrect classification of variable. Pruning helps to reduce this noise out from tree to avoid overfitting and generalize tree algorithm. In an experiment pre pruning was implemented by limiting maximum depth. Pre pruning helps in avoiding overfitting of decision trees by limiting depth of tree, as it helps to restrict tree from creating nodes and avoid split on bad features and generalizing while predicting labels.



Validation curve for Wine Quality Dataset

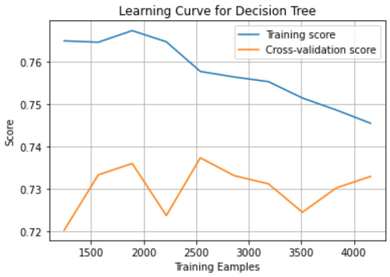


Validation curve for Breast Cancer Dataset

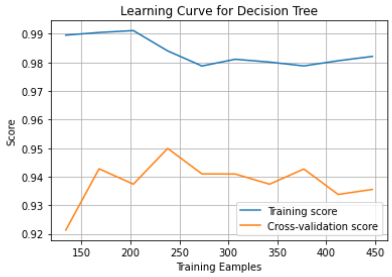
Observing validation curve for both datasets, it is found that decision tree tends to more overfit as it gets more deeper. This is because at each level of tree, data partition is performed at smaller set of data and hence it is more tends to overfitting, by creating more noise in data classification. Generalization and optimization hyperparameters tunning for decision tree was done on various range of max_depth using scikit-learn library's GridSearchCV function and analyzing validation curve for both training and cross validation set. GridSearchCV function optimizes parameters by cross-validating grid search over a parameter grid.

Dataset	**Accuracy**	**Best Parameters GridSearchCV**	**Tuned hyperparameter with GridsearchCV and validation curve analysis**
Wine quality	77.07%	{'max_depth': 24.0}	max_depth = 7
Wisconsin breast cancer	91.42%	{'max_depth': 8.0}	max_depth = 7

With these tuned hyperparameters, decision tree classifier was applied to different training set sizes to determine cross validation training and testing scores.



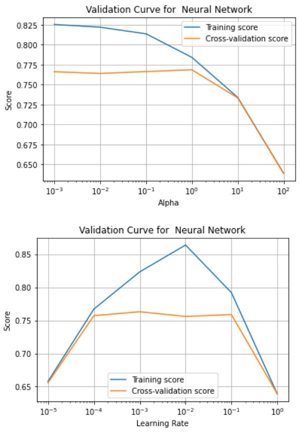
Learning curve for Wine Quality Dataset



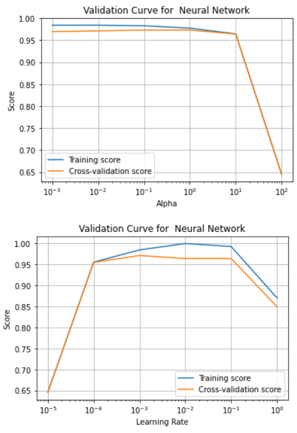
Learning curve for Breast Cancer Dataset

Observing learning curve for decision tree, there is high variance between training and cross validation scores that leads to overfitting and, therefore adding more data to training set can converge together training and cross-validation data and generalize more effectively.

2. **Neural Network:** MultilayerPerceptron (MLP) classifier was experimented with two datasets to model neural network. Neural network is helpful in nonlinear function approximation to identify data relationship pattern. The hidden layer in experiment was using 3 layers with layer size of (20,20,20) neurons i.e. 20 perceptron on each layer. Also learning rate and alpha was experimented to measure and analyze accuracy of model.



Validation curve for Wine Quality Dataset



Validation curve for Breast Cancer Dataset

With validation curve for both datasets plotted with alpha and learning rate vs accuracy score, it is observed that neural network is oversensitive to training and test data. High value of alphas tends to fix high variance which is sign of overfitting; likewise, lower value of alpha tends to fix high bias, a sign of underfitting. Similarly, for learning rate parameter which controls step-size in parameter space search, a very high value begins to diverge, and with low values of learning rate impedes loss function improvement. The ideal parameters for alpha are between 0.1 to 1, and for learning ideal range is observed between 0.01 and 1.0

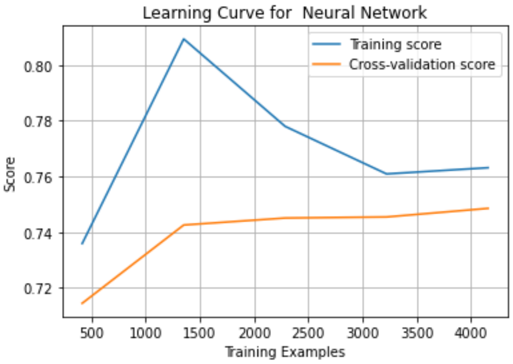
Hyperparameter tuning was performed for varying degrees of alphas, learning rate and hidden layer for datasets using scikit-learn library's GridSearchCV function and analyzing validation curve for both training and cross validation set.

Dataset	**Accuracy**	**Best Parameters GridSearchCV**	**Tuned hyperparameter with GridsearchCV and validation curve analysis**

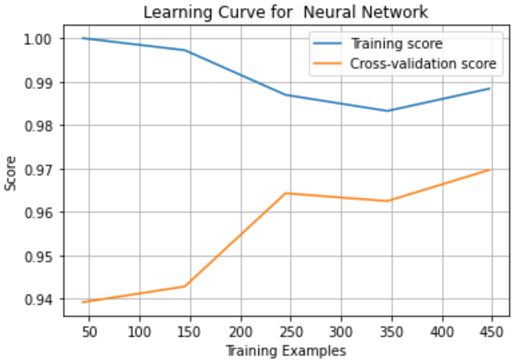
Wine quality	77.69%	{'alpha': 1.0, 'hidden_layer_sizes': (10, 5, 5), 'learning_rate_init': 0.01}	alpha = 0.1
			hidden_layer_sizes = (20,20,20)
			learning_rate_init = 0.1

Wisconsin breast cancer	96.43%	{'alpha': 0.01, 'hidden_layer_sizes': (10, 5, 10), 'learning_rate_init': 0.001}	alpha = 0.1
			hidden_layer_sizes = (20,20,20)
			learning_rate_init = 0.1

With these tuned hyperparameters, MLP classifier was applied to different training set sizes to determine cross validation training and testing scores.



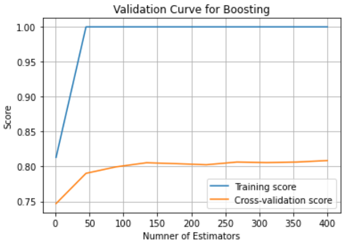
Learning curve for Wine Quality Dataset



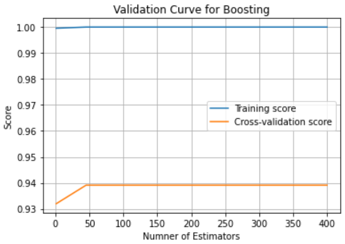
Learning curve for Breast Cancer Dataset

Observing learning curve, although for both datasets training and cross validation are converging together. Specifically wine quality dataset it showing desired performance and learner can effectively generalize. For breast cancer dataset it can be observed that, althoguh learner is converging, but have high variance leading to overfitting. More training data and hyperparameter optimization can help model to generalize effectively.

3. **Boosting**:   Boosting is an ensemble method of improving model, as it converts weak learner to strong learners for prediction. Adaboost algorithm was used in boosting with decision tree as learner. Decision tree is weak learner and Adaboost can improve decision tree performance, as at every iteration Adaboost misclassify incorrect observation with higher weight, so that next iteration can classify correct with more weight. In the experiment decision tree estimator with max_depth of 7 was implemented with Adaboost classifier to analyze accuracy of model. With AdaBoost, decision tree performs well and does not require high parameter tuning, and also it learns from mistakes of models that was previously build and readjust model based on weight, until tree is build.



Validation curve for Wine Quality Dataset

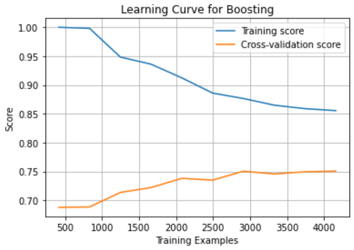


Validation curve for Breast Cancer Dataset

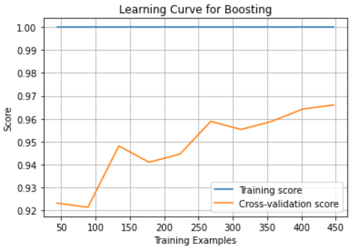
Observing validation curve for both datasets, it is found that training boosting improves training as well as testing accuracy compared with decision tree. Training accuracy quickly tends to converge to maximum value of 1, and also improves cross-validation accuracy. This is due to that fact that, in boosting weight of next model in decision tree will learn from weight of previous model. Finally, prediction of each weighted tree is gathered to make final prediction. Hyperparameter tuning was performed with varying degree of n_estimators_range and determine best parameters for dataset using scikit-learn library's GridSearchCV function and analyzing validation curve for both training and cross validation set.

Dataset	**Accuracy**	**Best Parameters GridSearchCV**	**Tuned hyperparameter with GridsearchCV and validation curve analysis**
Wine quality	81.23%	{'n_estimators': 35}	n_estimators= 18
Wisconsin breast cancer	93.57%	{'n_estimators': 14}	n_estimators= 18

With these tuned hyperparameters, Adaboost classifier was applied to different training set sizes to determine cross validation training and testing scores.



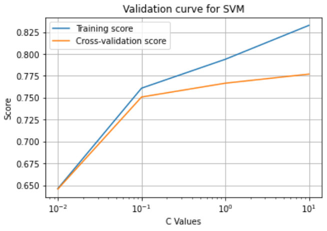
Learning curve for Wine Quality Dataset



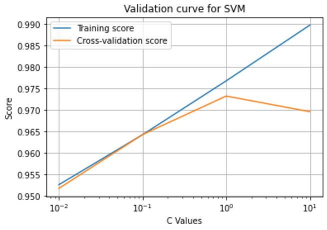
Learning curve for Breast Cancer Dataset

Observing learning curve, training and cross validation for wine quality dataset converging effectively with boosting, but with high variance between training and cross validation accuracy suggest more training data may help in generalizing model. Likewise; for Wisconsin breast cancer dataset training curve it at maximum value, while cross validation curve is growing with training sets suggest high variance problem i.e. overfitting. More training data can help improving performance of model and help in generalization.

4. **Support Vector Machine:** Support Vector Machine (SVM) is an eager learner. SVM is maximum margin classifier and it maximizes the hyperplane to reduce generalization error to maximum extent. SVM can use both linear and non-linear kernel functions in order to fit decision boundaries. In the experiment, C parameter is used to avoid misclassification of training data.



Validation curve for Wine Quality Dataset



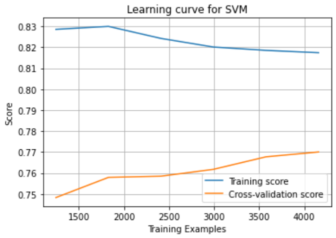
Validation curve for Breast Cancer Dataset

After validation curve for both datasets are evaluated, it is observed that overfitting is happening if C value becomes greater than 0.1. Larger C value tends to overfitting, as it will choose smaller margin hyperplane for training classification, conversely lower C values gives higher bias and lower variance and also look for higher hyperplane that leads to misclassification on the wrong side of decision margin.

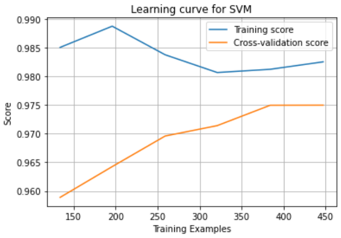
Hyperparameter tuning was performed with varying degree of C values using scikit-learn library's GridSearchCV function and analyzing validation curve for both training and cross validation set.

Dataset	**Accuracy**	**Best Parameters GridSearchCV**	**Tuned hyperparameter with GridsearchCV and validation curve analysis**
Wine quality	78.69%	{'C': 10}	C=4.2
Wisconsin breast cancer	96.42%	{'C': 1}	C=4.2

With these tuned hyperparameters, SVM classifier was applied to different training set sizes to determine cross validation training and testing scores.



Learning curve for Wine Quality Dataset

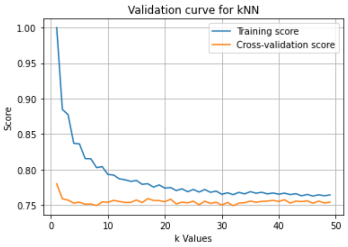


Learning curve for Breast Cancer Dataset

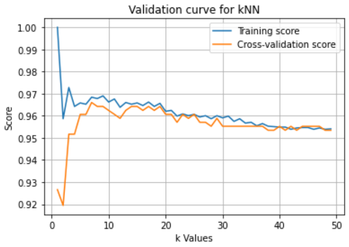
Observing learning curve, it is found that both wine quality and Wisconsin breast cancer datasets are converging, but are overfitting due to high variance. More training data to the model training can help in generalizing model.

The performance of algorithm is measure by anayzing confusaion matrix and ROC-AUC curve.

5. **K Nearest Neighbor (KNN):** K-Nearest Neighbor (KNN) is instance based simple learning algorithm that analyzes all available cases and classify them based on similarity measure to its nearest neighbor to make prediction. In the experiment default K value was tuned to determine accuracy model.



Validation curve for Wine Quality Dataset

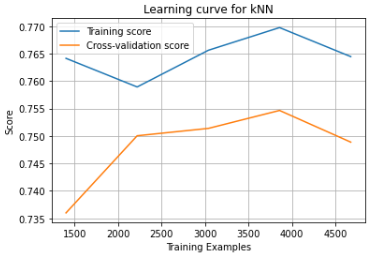


Validation curve for Breast Cancer Dataset

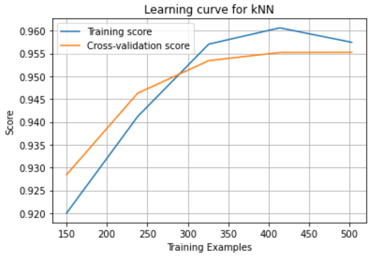
Observing validation curve for both datasets, it is found that small value of K is giving more accuracy but may lead to overfitting, but it is highly variant and have high influence in classification result. Conversely larger K value is computationally expensive and often leads to underfitting. Hyperparameter tuning was done with varying values of K values using scikit-learn library's GridSearchCV function and analyzing validation curve for both training and cross validation set.

Dataset	**Accuracy**	**Best Parameters GridSearchCV**	**Tuned hyperparameter with GridsearchCV and validation curve analysis**
Wine quality	88.00%	{'n_neighbors': 9}	n_neighbors = 8
Wisconsin breast cancer	97.71%	{'n_neighbors': 7}	n_neighbors = 8

With these tuned hyperparameters, KNN classifier was applied to different training set sizes to determine cross validation training and testing scores



Learning curve for Wine Quality Dataset



Learning curve for Breast Cancer Dataset

Observing learning curve, there is high variance between training and cross validation scores for wine quality dataset, although breast cancer dataset is converging to desired performance. Adding more training data will help reducing overfitting issue and help in generalizing model.

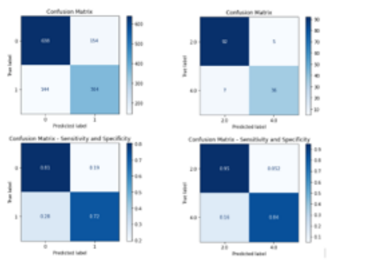
Performance Analysis of Learner: To measure performance of machine learning learner, accuracy provides valuable information about how well model is performing with given dataset, but it lacks metrics on how good is model performing in predicting correct labels of classification. To identify model ability from imbalance dataset to identify correctness of model, confusion metrices and ROC-AUC curve (Receiver Operating Characteristic – Area Under Coverage) are useful metrices.

Confusion metrics helps visualize models ability to identify true positives (TP) or sensitivity (classification that is correctly identified by model from samples of all positively identified classification), false positive (FP) (classification that is incorrectly identified by model from samples of all positively identified classification), true negatives (TN) or specificity (classification that correctly identified by model from samples of all negatively identified classification) and false negative (FN) (classification that incorrectly identified by model from samples of all negatively identified classification).

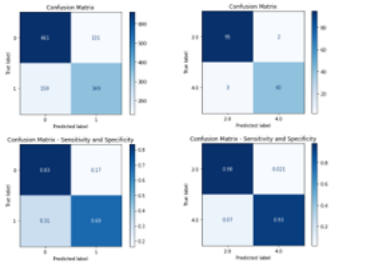
ROC-AUC curve helps in visualizing how well TP performs with TN, basically charting for each threshold calculate true positive rate and true negative rate. AUC is used to identify area under curve.

Precision (TP/(TP+FP) and Recall (TP/ (TP+FN)) metrices that identifies fraction of true positives among all actual samples and fraction of true positives among all predicted samples. F1-score is a natural mean of precision and recall (2 * (Precision * Recall)/ (Precision + Recall)). Using these metrices it can be predicted that model with high Precision, Recall rate and F1-score is performing well in identifying accuracy of model.

Learner	**Dataset**	**Classification Label**	**precision**	**recall**	**f1-score**	**support**
Decision Tree	Wine Quality	0	0.82	0.81	0.81	792
Decision Tree	Wine Quality	1	0.7	0.72	0.71	508
Decision Tree	Breast Cancer	2.0	0.93	0.95	0.94	97
Decision Tree	Breast Cancer	4.0	0.88	0.84	0.86	43
Neural Network	Wine Quality	0	0.81	0.83	0.82	792
Neural Network	Wine Quality	1	0.73	0.69	0.71	508
Neural Network	Breast Cancer	2.0	0.97	0.98	0.97	97
Neural Network	Breast Cancer	4.0	0.95	0.93	0.94	43
Boosting	Wine Quality	0	0.82	0.89	0.85	792
Boosting	Wine Quality	1	0.8	0.69	0.74	508
Boosting	Breast Cancer	2.0	0.96	0.95	0.95	97
Boosting	Breast Cancer	4.0	0.89	0.91	0.9	43
SVC	Wine Quality	0	0.78	0.88	0.83	792
SVC	Wine Quality	1	0.77	0.62	0.69	508
SVC	Breast Cancer	2.0	0.98	0.97	0.97	97
SVC	Breast Cancer	4.0	0.93	0.95	0.94	43
KNN	Wine Quality	0	0.77	0.83	0.8	815
KNN	Wine Quality	1	0.68	0.58	0.63	485
KNN	Breast Cancer	2.0	0.97	1	0.98	84
KNN	Breast Cancer	4.0	1	0.95	0.97	56

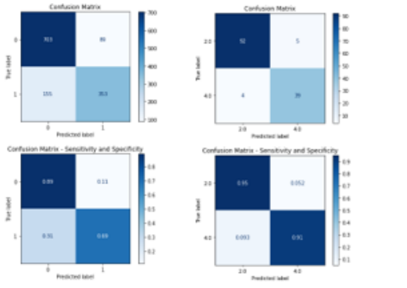


Confusion Matrix for Wine Quality Dataset Confusion Matrix for Breast Cancer Dataset

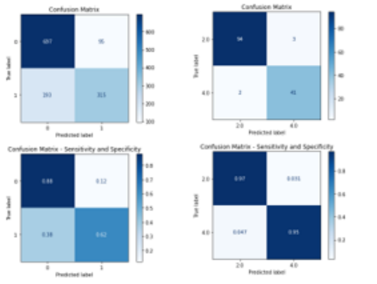


Confusion Matrix for Wine Quality Dataset Confusion Matrix for Breast Cancer Dataset

Decision Tree Confusion Metrics Neural Network Confusion Metrics

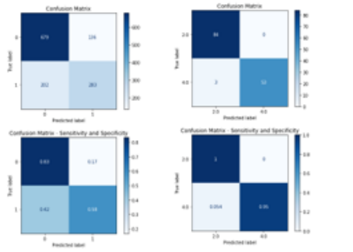


Confusion Matrix for Wine Quality Dataset Confusion Matrix for Breast Cancer Dataset



Confusion Matrix for Wine Quality Dataset Confusion Matrix for Breast Cancer Data

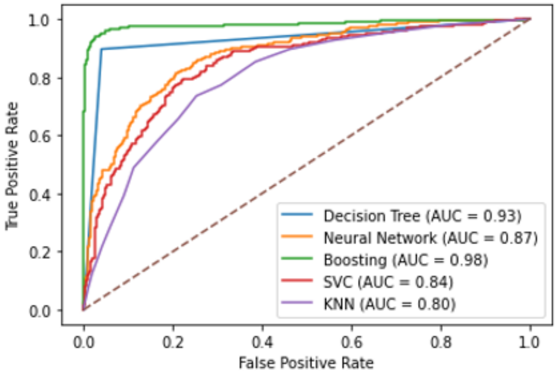
Boosting Confusion Metrics SVM Confusion Metrics

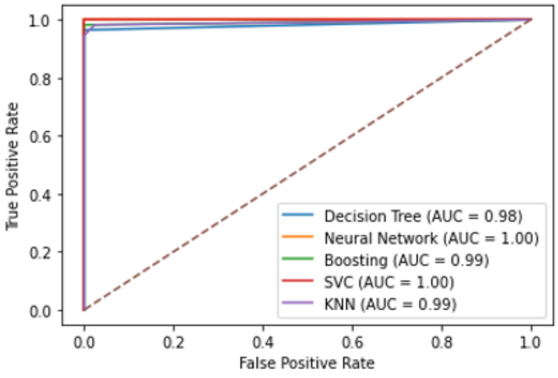


Confusion Matrix for Wine Quality Dataset Confusion Matrix for Breast Cancer Dataset

KNN Confusion Metrics

Observing confusion matrices for both datasets, high rate of specificity (true positive values of classification) and sensitivity (true negative values of classification) and high F1-score (a natural mean of precision and recall) indicates that learner has performed well in identifying correct labels.





ROC-AUC curve for Wine Quality Dataset ROC-AUC curve for Breast Cancer Dataset

Observing ROC-AUC curve for both datasets, it is found that boosting, neural network is performing better and have more area under coverage.

Conclusion: Overall project experience was unique and interesting. Exploring with these algorithms, it is observed that decision tree and KNN accuracy scores fare poorly in comparison with Adaboost, SVM and in neural network. Decision tree and KNN is faster in training speed than other algorithms. Although there was higher variance in learning curve for neural network, boosting and supported vector machine (SVM) they were close to convergence and with more training data can generalize them effectively. Neural network tuning needs more parameters and still may overfit with training and cross validation scores, conversely KNN needs minimum tuning but may lead to higher variances in learning curve. Decision tree have issue with high variances in learning curve, but boosting algorithm can help eliminate issues with algorithm.

This experiment has helped me understanding engineering aspect of supervised learning. In future I intent to explore more hyperparameters and metrices to understand and analyze supervised learning algorithms. Also, I like to explore these algorithms with more datasets to see affect of more diverse and tentatively heterogenous datasets on parameters.

References:

- Tim Mitchell: book – Machine Learning
- Dr. Charles Isabell and Dr. Michael Littman: video lectures supervised learning chapter 1 – 7
- Dr. William Wolberg from University of Wisconsin Hospital
- Paulo Cortez, University of Minho, Guimarães, Portugal
- UCI Machine Learning Repository
- Openml.org
- scikit-learn library and documentation
- TA’s during office hour and piazza forum