Supervised Learning Assignment

Student Name: Sathish Sampath
GT Account Name: ssampath33

Introduction

This paper describes the implementation of five supervised learning algorithm over two interesting datasets and their performance is analysed. The five supervised learning algorithm are Decision Tree, k Nearest Neighbor(KNN), Artificial Neural Network(ANN), Support Vector Machines(SVM) and Boosting. The algorithms are implemented using Scikit-learn library in Python language.

Datasets

The two datasets are Adult Income Dataset and QSAR biodegradation Dataset. Both the datasets are downloaded from UC Irvine Machine Learning Repository.

QSAR Biodegradation Dataset (Biodegradation)

Biodegradation experimental values of 1055 chemicals are available in this dataset. The chemicals were analysed in 41 different features (Number of heavy atoms, Number of substituted benzene C, Number of nitro groups, etc.) This data is used to develop models for the study of the relationships between chemical structure and biodegradation of molecules. The Classification model developed should discriminate the chemicals as Ready Biodegradable and Not Ready Biodegradable molecules. This is an interesting problem as the biodegradability of the chemicals cannot be determined using any simple rules. The machine model developed will help us in determining the Biodegradability of the chemical more accurately.

This dataset has totally 1055 instances with 41 attribute values each. The target function is discrete valued (RB- Ready Biodegradable and NB - Ready Biodegradable).

Adult Income Dataset

The dataset is a subset from the 1994 US Census, which is used to relate socio-economic attributes like education, heritage and age (among others) against the adult's income, in this case, whether income is above or below $50,000 per year. The data can be used to determine the important factors that influence the household income. The income class determination based on the known factors will help the Governments to focus more on the factors and develop their population with respect to those factors.

The dataset consists of 14 attributes, and over 19,000 instances. The data is used to develop models for the classification of adult profiles based on their socio-economic attribute values. The target function is discrete valued (<=$50k, >$50k).

Implementation

The algorithms are implemented using Scikit-learn library in Python language. The Scikit-learn is machine learning library and it features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and neural network.

For the evaluation of the performance of the algorithms, the overall dataset is split into two: 70% as training set and 30% as test set. For cross validation purposes, the K-Fold method with 5 folds is implemented on the training part using the Scikit-Learn's inbuilt function. The K-Fold method(for K=5) further splits the train dataset into learning dataset(80%) and cross validation dataset(20 %).

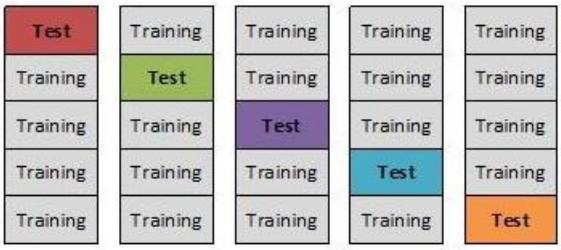

Figure 1. K-Fold Method

As the requirement for the analysis is classification and not regression, the performance of each algorithm will be measured in Accuracy and not in Root Mean Squared Error(RMSE). All five algorithms are implemented and their performance over both the datasets for different combination of hyperparameters are examined using Scikit-Learn's GridSearchCV function and the results are analysed below.

Decision Tree

The Decision Tree classifies instances by sorting them down the tree from the root to the leaf node. Each node specifies a test of some attribute of the instance and each branch descending from that node corresponds to one of the possible values for this attribute. The Decision Tree is implemented using the Scikit-Learn *DecisionTreeClassifier* method. The following parameters are tuned

- Criterion: parameter – determines the quality of the split. The possible values are 'Gini' and 'Entropy'.
- Pruning Alpha – determines the pruning level. As the pruning level increases, number of nodes pruned will increase.

Pruning is implemented to remove the less effective nodes and thereby reduce overfitting. The Pruning level is determined by the Pruning Alpha parameter. Pruning reduces both the size of the tree and the train accuracy but increases the test accuracy. The best parameters are determined using the test accuracy score.

Performance in Biodegradation dataset

The Figure 2 shows the performance of the Decision Tree with respect to different parameters. The log of Pruning Alpha('alpha') is taken along the x axis and the accuracy of both criterions over different values of alpha is analysed. As the alpha value increases beyond a limit, the graph shows that the accuracy for both the criterions reduce at a faster rate. This might be the result of *Over Generalization*, as the alpha increases, more number of nodes will be pruned and thereby the decision tree will be more generalized. From the graph, it is clear that the best accuracy score can be obtained using the parameter {criterion = 'entropy', pruning_alpha=[0.0,0.000316,0.001]}

This implies that the Information Gain is performing better than the Gini Index with respect to this dataset. The node count of the Decision Tree stays constant at 24 for all three values of the pruning_alpha.

The Figure 3 shows the Learning curve. For the smaller size of dataset, the cross validation accuracy score is lower than the train accuracy score. I suspect the possibility of a small bias in the initial model training. But as the dataset size increases, the cross validation accuracy score increases.

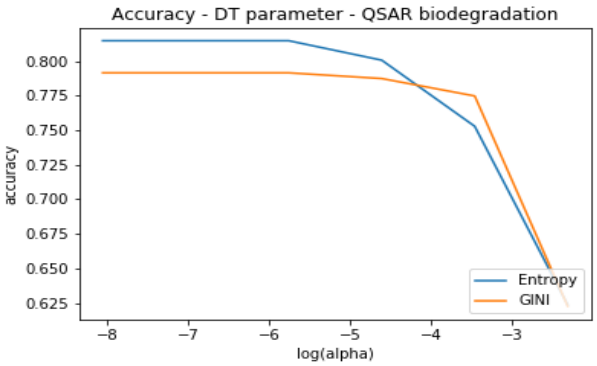

Figure 2. Biodegraation DT Accuracy Score vs Log(alpha)

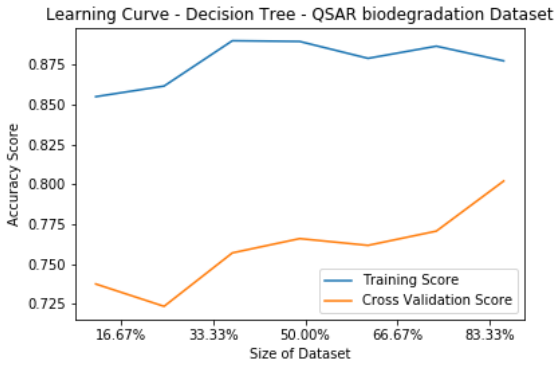

Figure 3. Biodegradation DT Learning Curve

Performance in Adult Income dataset

The Figure 4 shows the performance of the Decision Tree with respect to different parameters. The log of Pruning Alpha('alpha') is taken along the x axis and the accuracy of both criterions over different values of alpha is analysed. The accuracy of both the criterions increased for few values of alpha and after a point, starts to reduce at faster rate. As explained earlier, this might be the result of Over Generalization. From the graph, it is clear that the best accuracy score can be obtained using the parameter {criterion = 'entropy', pruning_alpha = 0.003162

}

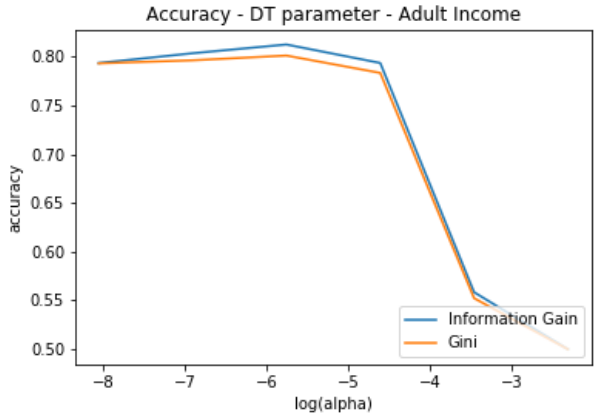

Figure 4 Adult DT Accuracy Score vs Log(alpha)

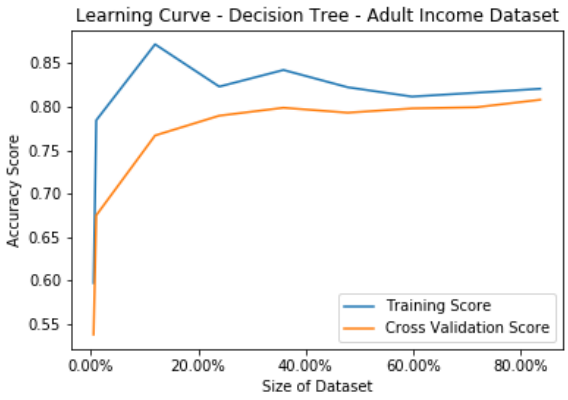

Figure 5 Adult DT Learning Curve

This implies that the Information Gain is performing better than the Gini Index with respect to this dataset. The number of nodes in the tree with respect to best alpha value is 44. The Figure 5 shows the Learning Curve for different sizes of the training data. The cross validation accuracy score was lesser than training accuracy score for the smaller dataset size, but as the size increase, the difference between both the scores reduced to a minimum amount.

The overall training time for this dataset is 7 seconds. The DT class weight 'Balanced' is used for both the datasets. As training time for both the datasets (of varying size) are small, it is clear that Decision Tree is much faster in computation.

K Nearest Neighbor(KNN)

K Nearest Neighbor method is the most basic instance-based method. The KNN algorithm is implemented using Scikit Learn's 'KNeighborsClassifier'. The parameters used for tuning the performance of the KNN are

- Neighbor count(n_neighbors): Number of neighbors to use.
- Weights(weights): weight function used in prediction. 'Uniform' for applying uniform weight to all n neighbors. 'Distance' for applying higher weightage to nearby neighbors.
- Distance Metric(metric): The distance metric to use for the tree, like 'Manhattan', 'Euclidean', etc.

Performance in Biodegradation dataset

The KNN is implemented with wide range of parameters and the accuracy of the model with respect to each parameter combination is noted, to determine the best combination of the parameters. The best performances for different values of KNN parameters for the dataset is mentioned in the table below.

Accuracy Score	Distance Metric	N Neighbors	Weight
0.85400924	manhattan	7	distance
0.860057374	manhattan	10	distance
0.865056682	manhattan	13	distance
0.859999049	manhattan	13	uniform
0.859005604	manhattan	16	distance
0.859967719	euclidean	19	distance
0.856976141	manhattan	19	distance
0.856935641	manhattan	22	distance
0.85385489	euclidean	22	distance
0.854903092	euclidean	25	distance

With respect to this dataset, the model performs better in the k-range of 10 to 25. The Distance metric 'manhattan' and 'euclidean' are equally performing well. Many top performing combinations are having weight 'distance'. The best parameter combination, by highest accuracy score is obtained for 'Manhattan' distance metric, 13 Nearest neighbor count and 'distance' weight.

The parameters with poor performance are listed below.

Accuracy Score	Distance Metric	N Neighbors	Weight
0.788006715	chebyshev	34	uniform
0.786115152	chebyshev	37	uniform
0.785124722	chebyshev	25	uniform

All the combinations have 'Chebyshev' distance metric and have higher values of N (>25). This might be due to Overfitting.

The figure 6 shows the Learning Curve for the Biodegradation dataset. The model is having very high train accuracy, even with the smaller dataset. The cross validation accuracy is increasing with the size of the dataset. This behaviour of KNN model implies that, the variation is very small in the dataset.

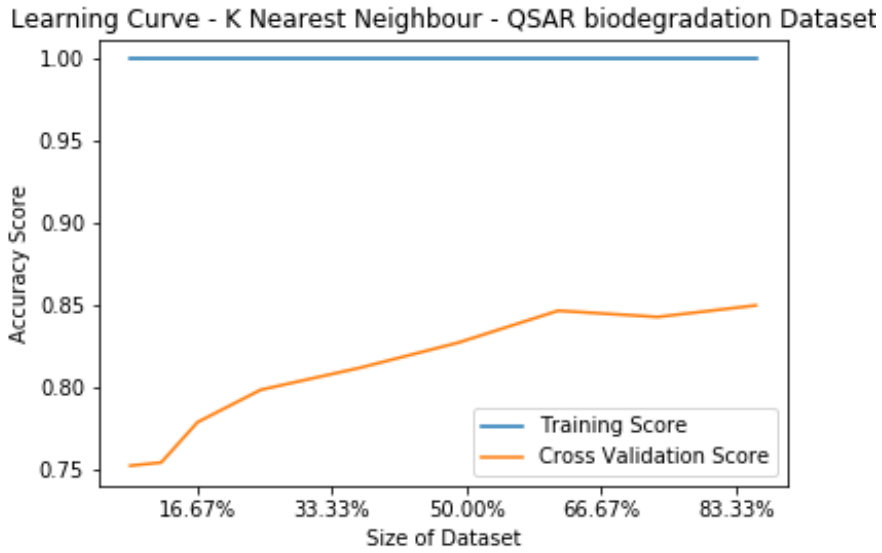

Figure 7 Biodegradation KNN Learning Curve

Performance in Adult Income dataset

The performance is quite similar to Biodegradable dataset's performance, except the fact that higher accuracy is achieved with lesser N value. The best performances of the model for different combinations of parameters are listed below.

Accuracy Score	Distance Metric	N Neighbors	Weight
0.756997213	euclidean	4	uniform
0.756291143	manhattan	4	uniform
0.745274971	chebyshev	4	uniform
0.750090186	manhattan	10	uniform
0.748676021	euclidean	10	uniform
0.745890181	euclidean	16	uniform
0.74578718	manhattan	16	uniform
0.743702306	euclidean	22	uniform
0.742412628	manhattan	22	uniform
0.742006309	manhattan	28	uniform

As the model is performing well with low N Value, the dataset might be having many instances with similar attribute value and output class. The parameters with poor performance are listed below.

All the poor performance are having higher N value(>30), so the model might have overfitted the train dataset.

Accuracy Score	Distance Metric	N Neighbors	Weight
0.692952394	chebyshev	43	uniform
0.692121836	chebyshev	46	uniform
0.684063589	chebyshev	49	uniform

The figure 6 shows the Learning Curve of the KNN model for Adult Income Dataset. Both the accuracy scores vary for very small dataset, but later stabilize. The Cross Validation Score changes by a small amount as the size of the dataset increases.

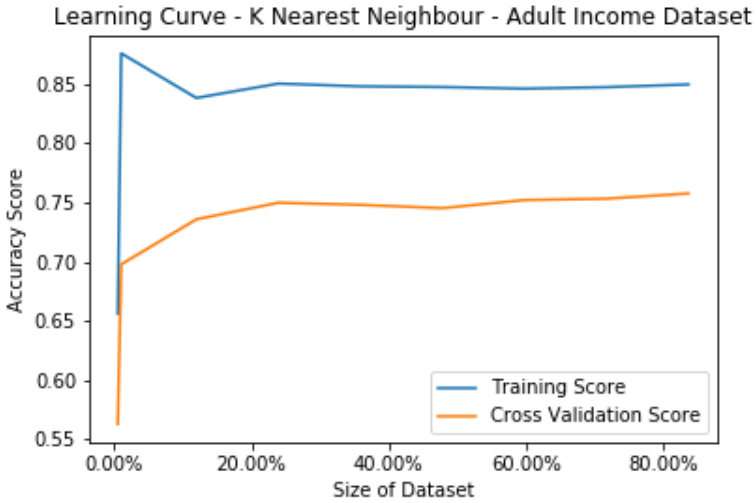

Figure 8 Adult Income KNN Learning Curve

Artificial Neural Network(ANN)

The Artificial Neural Network mimics the working of the neural network in the human brain. Each Perceptron is connected to N number of perceptron and the data is processed together. For analysis purpose, Multi-Layer Perceptron(MLP) is implemented using Scikit Learn's MLPClassifier method. The parameters used for tuning the performance of ANN are

- Hidden Layer Size.
- Activation function for the hidden layer.
- Regularization term(alpha)

For this analysis, the default solver ('stochastic gradient descent optimizer') is used. The artificial neural network algorithm is an eager learner because it uses back-propagation to determine appropriate weights between perceptrons.

Performance in Biodegradation dataset

The MLP is implemented with wide range of alpha, different hidden layer sizes and two different activation functions 'Relu' and 'Logistics'. The best performance of the ANN with respect to this dataset is given in the table below.

Accuracy score	Activation	Alpha	Hidden Layer Size
0.78729271	relu	3.16E-05	(82,)
0.78729271	relu	0.0001	(82,)
0.78729271	relu	0.000316228	(82,)
0.78729271	relu	0.001	(82,)
0.78729271	relu	0.003162278	(82,)
0.78729271	relu	0.01	(82,)
0.78729271	relu	0.031622777	(82,)
0.789327989	relu	0.1	(82,)
0.790358991	relu	1	(82,)

The poor performance of the ANN is

Accuracy score	Activation	Alpha	Hidden Layer Size
0.488613581	relu	1.00E-05	(20, 20, 20)
0.487630002	relu	3.16227766	(20, 20, 20)
0.48758902	relu	1	(20, 20, 20)

The MLP performs good for lesser number of hidden layers with more number perceptrons than with more number of hidden layers with less number of perceptrons. The Learning Curve shows dips in the accuracy with the increase in dataset size. This clearly indicates that the Variance is present in the dataset.

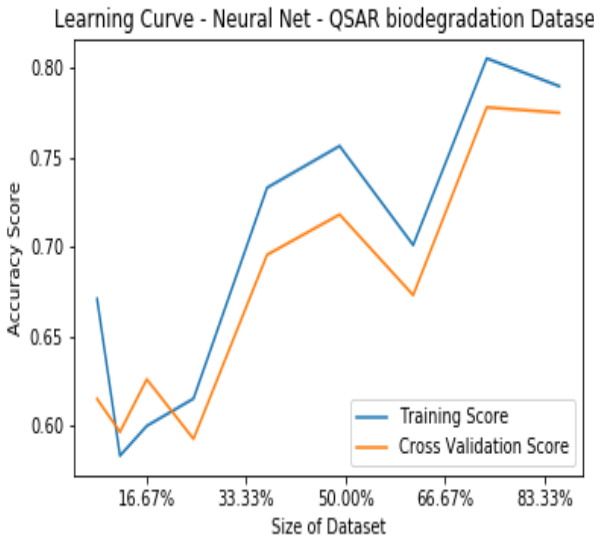

Figure 9 Biodegradation ANN Learning Curve

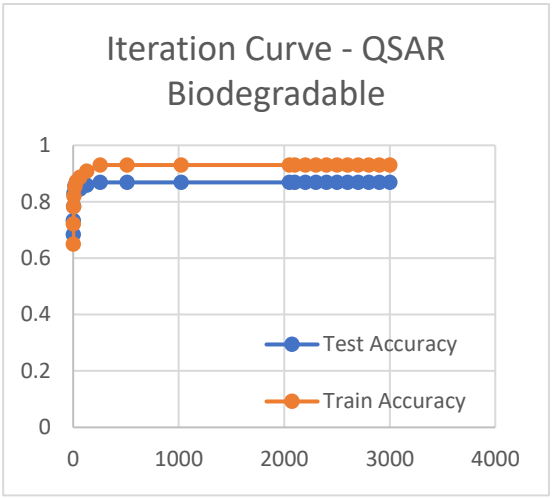

Figure 10 Biodegradation ANN Iteration Curve

In the iteration curve, the accuracy (test and train) increases with increase in iteration count. But after count 250, the accuracy does not alter much.

Performance in Adult Income dataset

The performance in this dataset is different from the biodegradable dataset. The best performance for different combination of parameters are mentioned below.

Accuracy score	Activation	Alpha	Hidden Layer Size
0.772800466	logistic	1.00E-05	(58, 58, 58)
0.772800466	logistic	3.16E-05	(58, 58, 58)
0.772800466	logistic	0.0001	(58, 58, 58)
0.772976212	logistic	0.000316228	(58, 58, 58)
0.771130099	logistic	0.001	(58, 58, 58)
0.771106011	logistic	0.003162278	(58, 58, 58)
0.773074588	logistic	0.01	(58, 58, 58)
0.769322912	logistic	0.031622777	(116, 116)
0.768603544	logistic	0.031622777	(58, 58, 58)
0.768180268	relu	1	(116, 116, 116)

The 'logistic' activation function has better performance and the multiple perceptron in multiple hidden layer configuration is performing better with respect to this dataset. The poor performance is from the configuration with lesser number of perceptron in multiple hidden layer.

The Learning Curve clearly indicates that the model is perfect for the dataset, as difference between the Train accuracy score and cross validation accuracy score is too small. The learning curve

has lower accuracy levels for small data sets. The instances selected might have been much diverse with respect to attribute value.

The Iteration curve is similar to the Biodegradable dataset. The accuracy(test and train) increases with increase in iteration count up to 64, then accuracy remains constant.

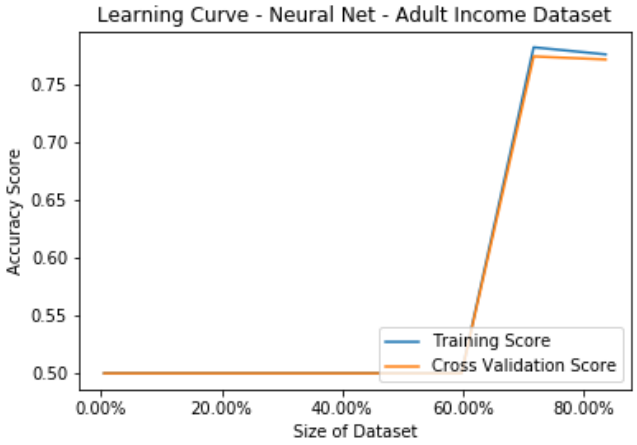

Figure 11 Adult Income ANN Learning Curve

Support Vector Machines (SVM)

The Support Vector Machines is implemented using Scikit Learn library. The SVM is implemented using two kernels – 'Linear' and 'RBF'. The Linear SVM is implemented with varying parameters of Alpha(inverse of Penalty parameter) and the RBF SVM is implemented with wide range of Gamma (Kernel Coefficient) and Alpha values(inverse of Penalty parameter).

Performance in Biodegradable Dataset

The SVM RBF kernel has higher accuracy with alpha value 0.001 and gamma value 0.2 than the SVM with Linear Kernel. The Linear Kernel maximum accuracy score is with also alpha score 0.01.

Accuracy score	Alpha	Gamma
0.867198	0.000316	0.4
0.866171	0.000316	0.2
0.866063	0.000316	0.8
0.871249	0.001	0.2
0.869151	0.001	0.4
0.867123	0.001	0.6
0.86299	0.001	0.8
0.862809	0.003162	0.6
0.861672	0.003162	1

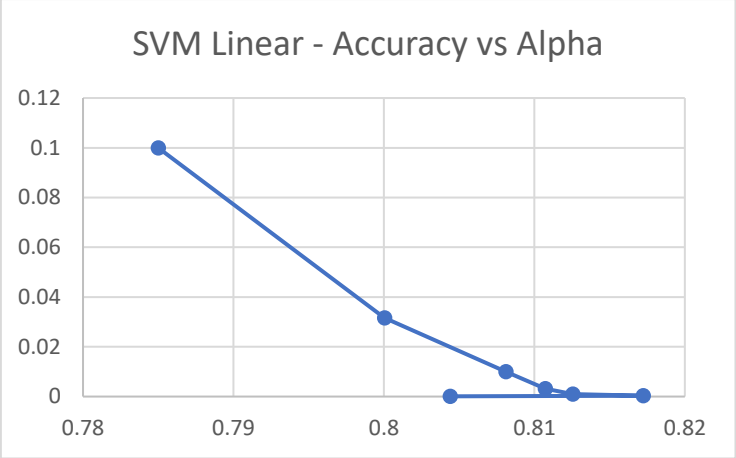

Figure 12 Biodegration SVM Linear Alpha vs Accuracy

The Learning Curve of both the kernels(SVM RBF and SVM Linear) have similar properties with respect to the dataset. The train accuracy score for the smaller dataset is higher than the cross validation accuracy score and as the dataset size increases, the cross validation accuracy score increases and the difference between the train curve and cross validation curve reduces.

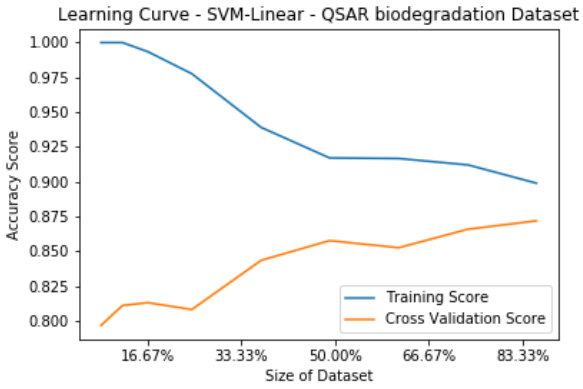

Figure 13 Biodegradation Learning Curve SVM Linear

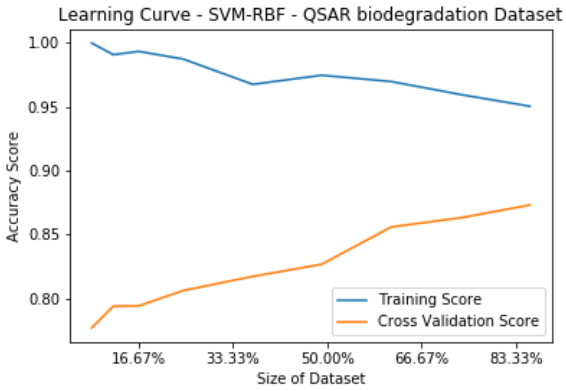

Figure 14 Biodegradation Learning Curve SVM RBF

Performance in Adult Income dataset

The SVM with Linear kernel performs better than SVM with RBF kernel. The performance of the SVM RBF kernel is mentioned in the table below.

Accuracy score	Alpha	Gamma
0.802167	0.003162	0.6
0.802546	0.01	0.8
0.804485	0.01	1
0.805592	0.01	1.2
0.802295	0.031623	1.4
0.801044	0.01	1.4

0.802258	0.031623	1.6
0.803349	0.031623	1.8
0.80147	0.01	1.8
0.802993	0.031623	2

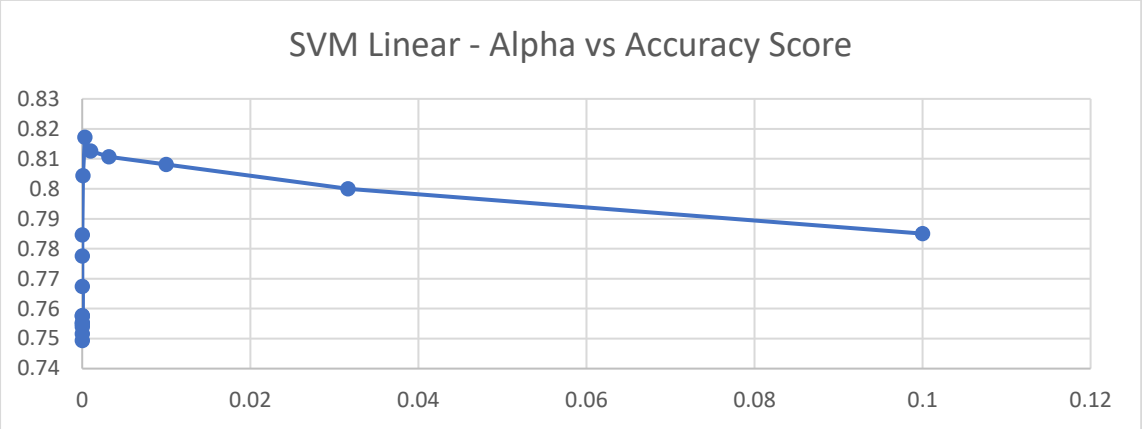

Figure 15 Adult Income SVM Linear Alpha vs Accuracy

The highest accuracy score is obtained by SVM Linear kernel with parameter alpha value 0.000316

The learning curves of both the kernel methods show similar properties.

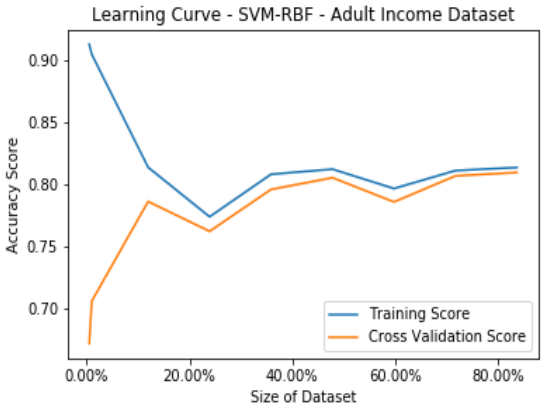

Figure 16 Adult Income Learning Curve SVM RBF

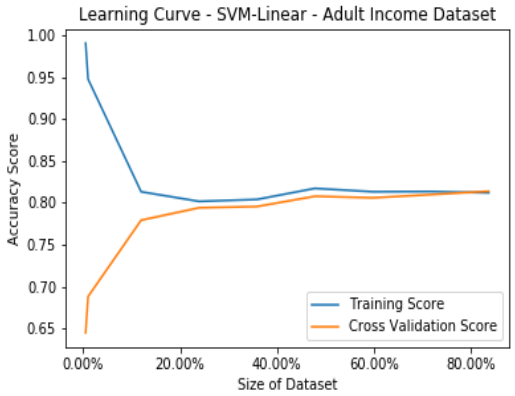

Figure 17 Adult Income Learning Curve SVM Linear

Boosting

Adaptive Boosting(AdaBoosting) is implemented for the analysis purpose using Scikit Learn Library. Boosting is an iterative process that applies information gain to learn over a subset of data. The process can be applied to other machine learning algorithms. The AdaBoosting is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases. The AdaBoosting when applied over the decision tree, will improve the learning of the model.

The Adaboosting has two parameters-

- Base Estimator - The base estimator from which the boosted ensemble is built.
- N Estimator - The maximum number of estimators at which boosting is terminated.

Performance in Biodegradable Dataset

The AdaBoosting performs with higher accuracy when Base Estimator is zero. The overall performance is also quite comparable with the Decision Tree's performance.

Accuracy	Alpha	N Estimator
0.838049	0	5
0.838049	0.003162	5
0.838049	0.000316	5
0.838049	0.001	5
0.837033	0	10
0.837033	0	20
0.837033	0	30
0.837033	0	45
0.837033	0	60
0.837033	0	80
0.837033	0	100

The difference between the training accuracy curve and cross validation accuracy curve is almost constant.

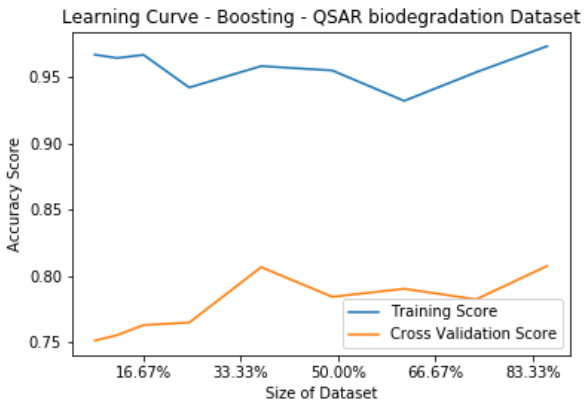

Figure 18 Biodegradation Boosting Learning Curve

Figure 19 Adult Income Boosting Learning Curve

Performance in Adult Income dataset

The AdaBoosting performance for different combination of parameter is mentioned below.

Accuracy	Alpha	N Estimator
0.8144	0.031623	20
0.8144	0.031623	30
0.8144	0.031623	45
0.8144	0.031623	60
0.8144	0.031623	80

0.8144	0.031623	100
0.813392	0.01	10
0.812588	0.003162	1
0.812588	0.003162	2
0.811425	0.031623	10

The best accuracy score is reached for Base Estimator(alpha) value 0.0316 and N Estimator value 20.

Unlike the previous dataset, the learning curve converges. The train accuracy curve and cross validation accuracy curve converge as the size of the dataset increases.

Summary

Comparison of Different Classifiers

QSAR Biodegradation Dataset

Algorithm	Parameter	Accuracy Score
Decision Tree	Pruning Alpha = 0, Class Weight= 'Balanced', Information Gain based method	0.8166
ANN	Activation='Relu', Alpha= 1.0, Hidden Layer Size=(82,)	0.78609
SVM	RBF Kernel, Inverse of Penalty = 0.001, Gamma = 0.2	0.8564
KNN	Manhattan Distance, K=13, Weight='distance'	0.84252
Boosting	Base Estimator=0, N Estimator=5	0.818981

Adult Income Dataset

Algorithm	Parameter	Accuracy Score
Decision Tree	Pruning Alpha =0.00316, Class Weight= 'Balanced', Information Gain based method	0.814843
ANN	Activation='Logistic', Alpha=0.01, Hidden Layer Size=(58,58,58)	0.77795
SVM	Linear Kernel, Alpha= 0.000316	0.81154
KNN	Euclidean Distance, K=4, Weight='Uniform'	0.766833
Boosting	Base Estimator=0.0316227, N Estimator=20	0.81508

Biodegradation Dataset has achieved the best performance from SVM(RBF Kernel) and Adult Dataset has achieved the best performance from AdaBoosting Classifier. The Decision Tree performed good in both the dataset.

References

1. QSAR Biodegradation Dataset, UCI ML Repository, https://archive.ics.uci.edu/ml/datasets/QSAR+biodegradation
2. Adult Dataset, UCI ML Repository, https://archive.ics.uci.edu/ml/datasets/adult
3. Code Source: Jonathan Tay(https://github.com/JonathanTay/CS-7641-assignment-1)
4. Scikit Learn, http://scikit-learn.org/stable/index.htm