**Adult Income Dataset**

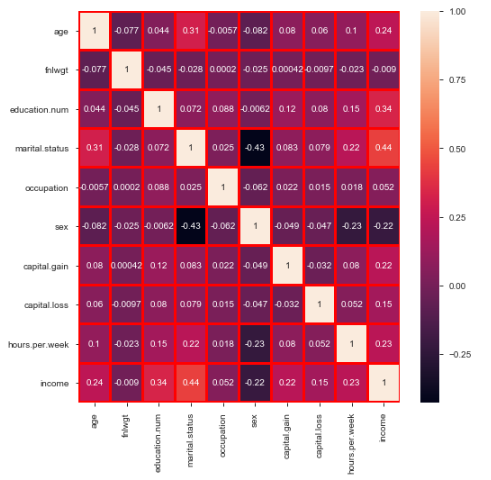
The dataset was extracted from the 1994 Census Bureau database. The prediction task is to determine whether a person makes over $50K a year. This will be done using three classification algorithms which are Support Vector Machine, Decision Tree and Boosting. The attributes are as follows:

Independent Variables: age, workclass, fnlwgt, education, education.num, marital.status, occupation, relationship, race, sex, capital.gain, capital.loss, hours.per.week, native.country

Dependent Variable: income

**Exploratory Data Analysis and Feature Engineering**:

* The dataset contains 15 variables and 32561 records
* Based on the correlation between different features and missing values some rows were dropped as explained the ipynb file
* The correlation matrix was used to identify the correlations among features



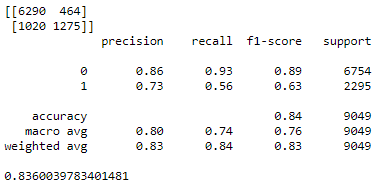
* The categorical features were given numeric labels. For e.g. sex was converted to 0 (Male) and 1 (Female). Similarly, all the different categories of marital statuses were classified as 1 (married) and 0 (unmarried).

**Tasks 1**

The dataset is split into train and test in the ratio 70:30 respectively. All the features are scaled by subtracting the mean from all Values of each feature and dividing the result by standard deviation of the features. This is done with the help of StandardScalar in scikit learn library in python. So, the resulting values of all the features lie between -1 and 1.

**Tasks 2 - Support Vector Machine**

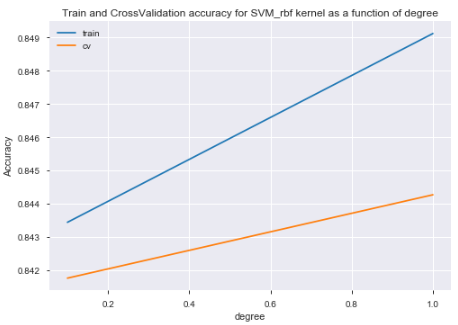
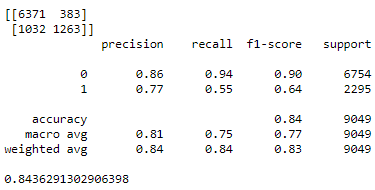
*Linear Kernel:* The model is trained on the train set using the linear kernel function of SVM with various values of ‘C’ and the optimum value of ‘C’ is identified that provides best accuracy for train and test sets. Below is the graph for train and cross validation scores for C = 0.1, 1, 10 & 100. The confusion matrix is also shown below for best ‘C’ i.e. 10.

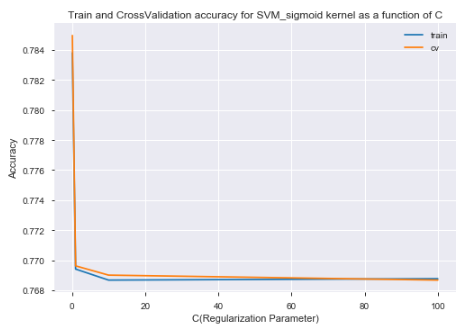
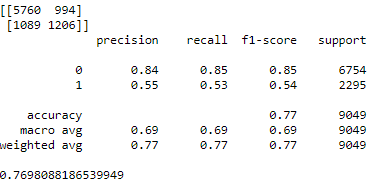
From the above graph we can say that the optimum values for train and test accuracy is obtained when C=10 as after this point the model overfits the data as training accuracy is increasing and cross validation accuracy is decreasing. Also the test accuracy is higher than train accuracy at C=0.1 which could be a bit mis-leading so C=10 is the best parameter to use and there is not significant difference between the results of these two parameters. The accuracy for test set is 83.6% at C=10.

Further k-fold cross validation is used where k=10 and the average score is 83.85%

*RBF Kernel:* The model is trained on the train set using the rbf kernel function of SVM with various values of C = 0.1 & 1. From the graph below it is evident that C=1 provides better accuracy for both train and cross validation sets. The test accuracy is 84.36%

*Sigmoid Kernel:* The model is trained on the train set using the sigmoid kernel of SVM for C= 0.1,1,10 & 100. From the graph below it is evident that the best result is obtained at C=0.1 as after this point both the cross validation and train accuracies reduce. The accuracy for test set is 76.98%

**Performance comparison between all three kernels:**

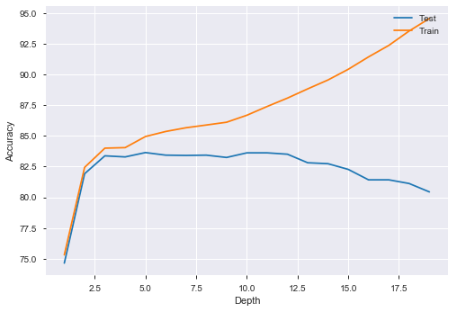
For the linear, rbf and sigmoid kernels the cv accuracies are 0.8390, 0.8445 & 0.769 and the test accuracies are 0.836, 0.8436 and 0.7698. Therefore, we can say that the rbf kernel is best for predicting the probability of income being greater than 50k.

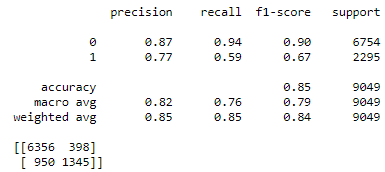
**Decision Tree**:

Decision tree with gini index is used as it is computationally less expensive than entropy. Gini index will select the best feature as root to split. Firstly, the decision tree was implemented without any depth specified. The test accuracy for this model is 79.3% but the training accuracy is 1 which could probably mean the model is overfitting the data. The max depth for the tree is 36. The tree is pruned further based on the optimum depth.

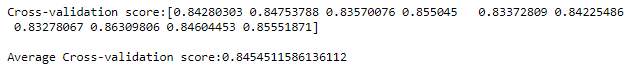


I also tried over-sampling and the accuracies didn’t show any improvement. To prune the tree the graph of depth vs accuracies is plotted. From the below graph it can be seen that depth=5 provides the best accuracy of test set and for training set the model overfits as depth increases beyond 5. So decision tree is nex implemented for depth=5. The training accuracy decreased to 84.2% which means the overfiting reduced. The test accuracy also increased to 85.1%



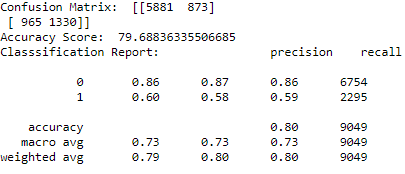
 

K-fold cross validation (K=10) for depth 5 provides the average accuracy as 84.54%.

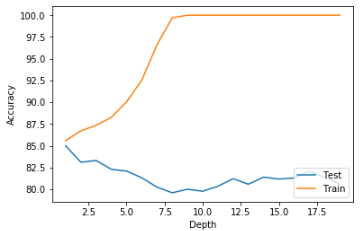


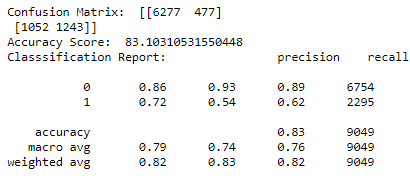
**Boosting:**

Adaboost classifier is used to boost the model trained by decision tree. Firstly, the model is boosted without specifying any depth and the accuracy is 79.68%. The confusion matrix is shown below.

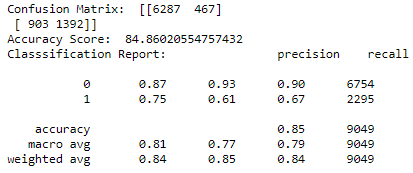


Next a graph is plotted for depth against accuracy to identify the optimum depth. Based on the optimum depth the tree will be pruned to provide the best accuracy.

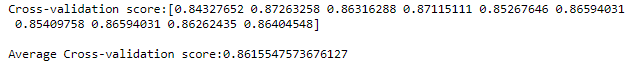


Depth of 2 provides the best accuracy for test set and beyond 2 the model overfits. So, the tree is pruned at depth 2  


Confusion matrix and accuracy are shown above for depth =2. The accuracy increased to 83.103% after boosting. Further I experimented with n\_estimators = 25 and the accuracy increased to 84.860% for depth=2. Below is the matrix.

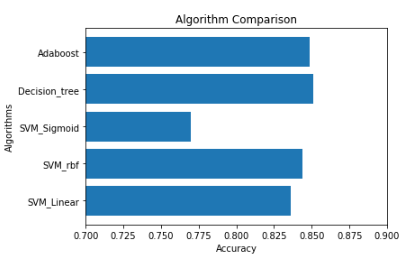


Cross-validation score for k=10 is 86.15% as shown below:



**Comparison of all the models**:

It is evident from the plot shown below that Decision tree is the best performer among all the algorithms with accuracy of 85.1% followed by Adaboost (84.86%), SVM\_rbf (84.36%), SVM\_Linear (83.6%) and SVM\_Sigmoid (76.98%) respectively.



**Appliances Energy Prediction Dataset**:

The Appliances Energy Prediction dataset is used in this assignment which can be downloaded at

<https://archive.ics.uci.edu/ml/datasets/Appliances+energy+prediction>

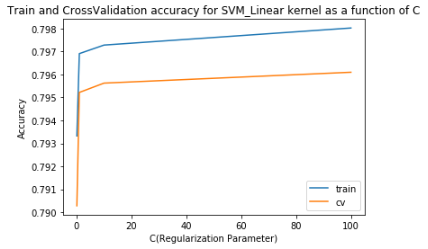
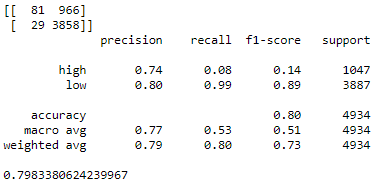
For this dataset various classification algorithms are to be used like SVM, Decision Tree and Boosting.

A brief description of the dataset is provided below:

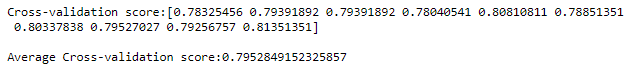
* The dataset consists of 29 features and 19735 records
* The dependent variable for the regression model is ‘Appliances’ which is energy use in Wh

**Support Vector Machine:**

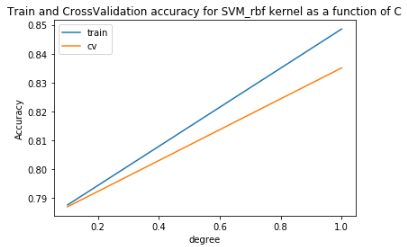
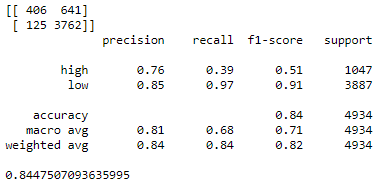
*Linear Kernel:* The model is trained on the train set using the linear kernel function of SVM with various values of ‘C’ and the optimum value of ‘C’ is identified that provides best accuracy for train and test sets. Below is the graph for train and cross validation scores for C = 0.1, 1, 10 & 100. The confusion matrix is also shown below for best ‘C’ i.e. 10. We can see that the best test accuracy is obtained at C=10 which is 79.83%. After this point the accuracies are constant.

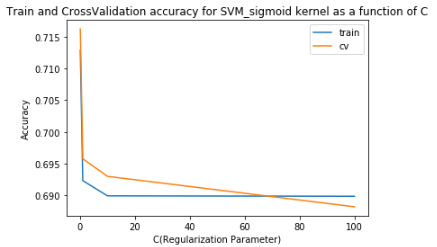
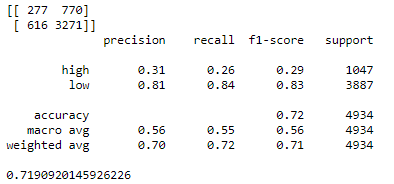
K-fold (k=10) cross validation gives the average train accuracy as 79.52% as shown below



*RBF Kernel:* The model is trained on the train set using the rbf kernel function of SVM with various values of C = 0.1 & 1. From the graph below it is evident that C=1 provides better accuracy for both train and cross validation sets. The test accuracy is 84.47%

*Sigmoid Kernel:* The model is trained on the train set using the sigmoid kernel of SVM for C= 0.1,1,10 & 100. From the graph below it is evident that the best result is obtained at C=0.1 as after this point both the cross validation and train accuracies reduce. The accuracy for test set is 71.90%

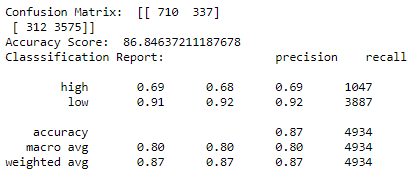
 

**Performance comparison between all three kernels:**

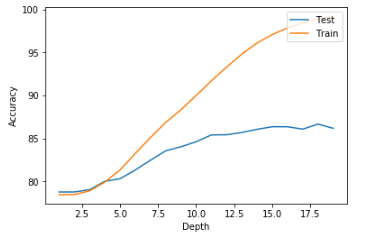
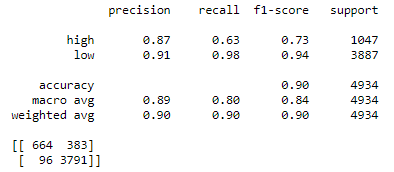
For the linear, rbf and sigmoid kernels the cv accuracies are 0.7961, 0.8350 & 0.6955 and the test accuracies are 0.7983, 0.8447 and 0.7190. Therefore, we can say that the rbf kernel is best for predicting energy usage.

**Decision Tree**:

Decision tree with gini index is used as it is computationally less expensive than entropy. Gini index will select the best feature as root to split. Firstly, the decision tree was implemented without any depth specified. The test accuracy for this model is 86.84% but the training accuracy is 1 which could probably mean the model is overfitting the data. The max depth for the tree is 26. The tree is pruned further based on the optimum depth.

To prune the tree the graph of depth vs accuracies is plotted. From the below graph it can be seen that depth=10 provides the best accuracy of test set and for training set the model overfits as depth increases beyond 10. So decision tree is next implemented for depth=10. The training accuracy decreased to 81.3% which means the overfiting reduced. The test accuracy also increased to 90.3%

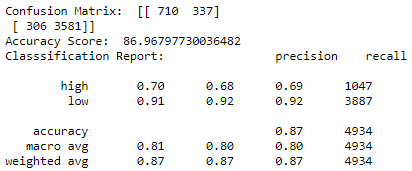


K-fold cross validation (K=10) for depth 10 provides the average train accuracy as 84.55%.

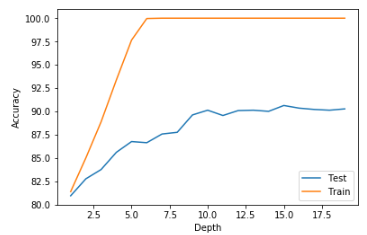


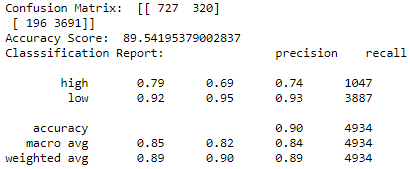
**Boosting:**

Adaboost classifier is used to boost the model trained by decision tree. Firstly, the model is boosted without specifying any depth and the accuracy is 86.96%. The confusion matrix is shown below.

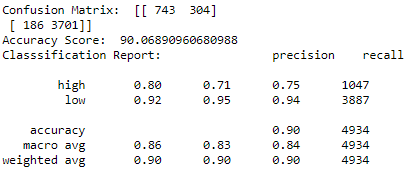


Next a graph is plotted for depth against accuracy to identify the optimum depth. Based on the optimum depth the tree will be pruned to provide the best accuracy.

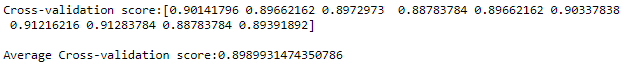
Depth of 12 provides best accuracy for n\_estimators=50 so the tree is pruned at depth=12. The confusion matrix and accuracy are shown below.



The accuracy increased to 89.54% after boosting. Experimentation was done with n\_estimators=100 and the accuracy obtained is 90.06%. The confusion matrix and accuracy are shown below.



K-fold cross validation (k=10) provides the average train accuracy as 89.89%.



**Comparison of all the models**:

It is evident from the plot shown below that Decision tree is the best performer among all the algorithms with test accuracy of 90.3% followed by Adaboost (90.06%), SVM\_rbf (84.47%), SVM\_Linear (79.83%) and SVM\_Sigmoid (71.90%) respectively.

