**Rain in Australia**

The objective is to predict whether rain will come tomorrow or not based on the location’s weather conditions. The dataset provides us with various variables such as Wind speed, humidity, temperature, pressure etc. Occurrence of rain is indicated by 1 and non-occurrence is indicated by 0. Independent variables are selected based on their correlations and through few trial and error methods.

**Independent Variables**: Mintemp, Maxtemp, Rainfall, WindGustSpeed, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Cloud3pm, Temp9am, Temp3pm, Raintod

**Dependent Variables**: Raintmr (Binary variable 0s and 1s)

There are null values in the dataset and they are removed using dropna () function. Min-max normalization is done on all the selected features. The dataset is divided into 70/30 split using train\_test\_split from sklearn. The 70% of the data is used for training various models and doing cross validation to select the best hyperparameters for all the models. The remaining 30% will be used as Test set to evaluate the accuracy of the best model taken through cross validation. For all the models, cross validation is done for kfold=3 as it will help us generalize the model and will not allow the model to overfit the training data.

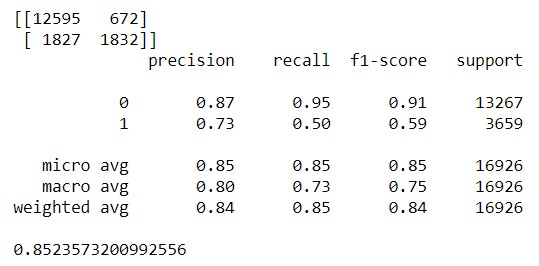
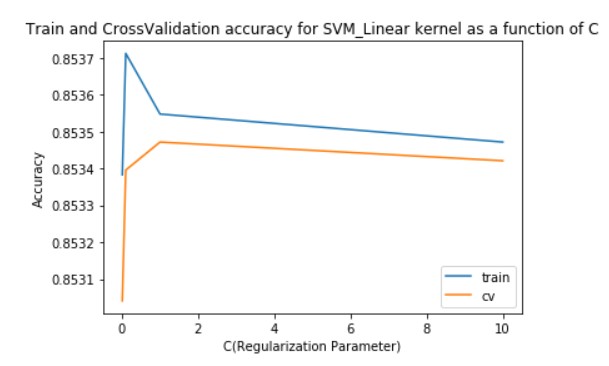
**Learning Algorithm 1: Support Vector Machine (SVM):**

Sklearn package in python is used to implement the SVM with various kernels for the dataset.

***Linear Kernel Function:***

The model is trained on various values of C (regularization parameter) to avoid overfitting /

underfitting of the data. Below are the graphs(left) for Train and CV scores for all values of C from 0.01,



0.1, 1, and 10 respectively and confusion matrix(right) when C=1 on Test set.

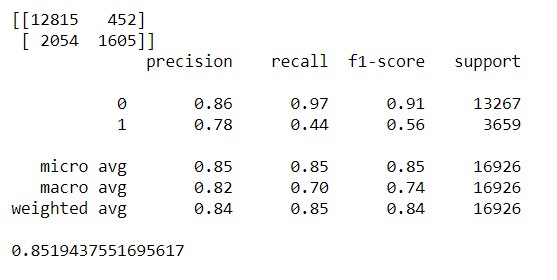
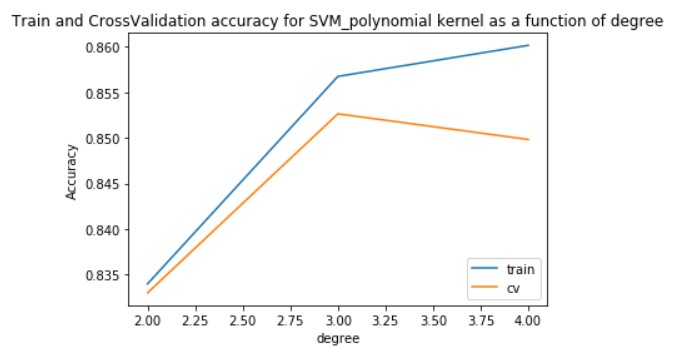
From the graph, we can conclude that at C=1, the model performs the best. As the C value increases, the model starts to overfit the data and test score starts to decrease. Initially, the model was underfitting the data at 0.01. As the model starts to generalize well for higher C values, Test score increased drastically. The accuracy for the Test set when c=1 is **0.85235**.

***Polynomial Kernel Function:***

The train set is trained on SVM\_polynomial kernel with various degree values from 2 to 4. As the degree of polynomial increases, the model parameters get more flexible and starts to fit the noise in the data. Hence for higher degree of polynomial, model tends to overfit the data which leads to very less train error but very high-test error. Hence, we will plot the learning curve as a function of degree after cross-validation to find the best polynomial degree suited for the model. The below graph shows Train and cross-validation scores(left) for SVM\_polynomial kernel and confusion matrix(right) when degree is

3 on Test set.

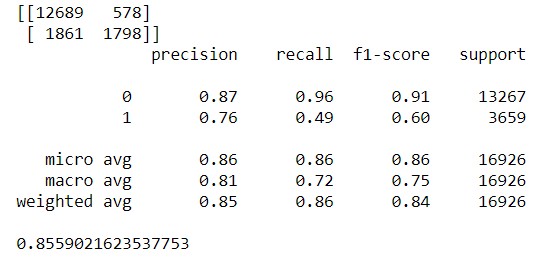
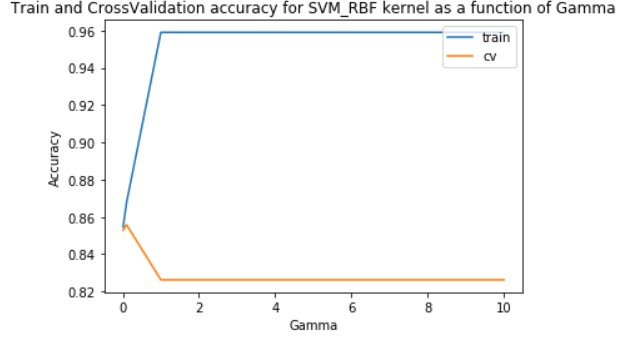
From the below graph, we can conclude that at degree =3, the model performs the best. As degree goes above 3, model overfits the data and cannot able to generalize well on the unseen data. It results in very less test accuracy. The accuracy on the Test set when degree = 3 is **0.85194**.



***RBF (Radial Basis Function) Kernel Function***:

For Rbf function, the influence of the support vectors depends on the gamma values. Higher the gamma values, lower the influence of the support vector and vice versa. Hence for lower values of gamma, model is too constrained and cannot able to capture the complexity of the model. When

gamma has high values, it tends to overfit the data. Hence, we must find the optimum gamma value where model does not overfit or underfit the data. This is done through cross-validation. The below graph(left) plots the cross-validation score and the train score as a function of gamma values and confusion matrix(right) for the Test set for gamma= 0.1

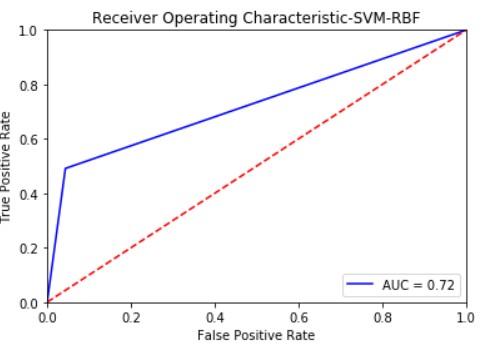
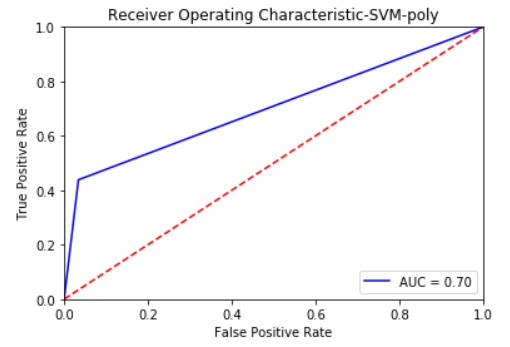
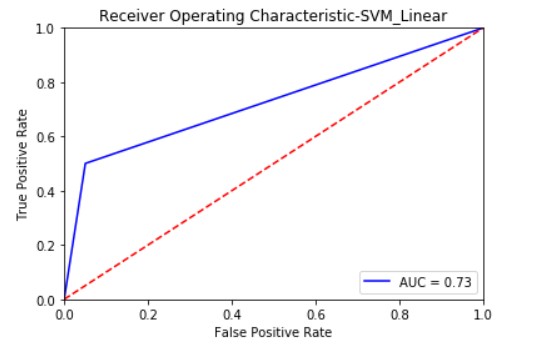


From the graph, we can conclude that, we get the best Cv score when gamma is 0.1. As we increase the gamma value, model overfits the data hence train score goes up and Cv score went down.

The accuracy for test set when gamma is 0.1 is **0.8559**.

***Comparison between all the kernels in SVM:***

From the above three graphs, the best cross validation accuracy for Linear, polynomial and Rbf are **0.85, 0.852 and 0.855** respectively. from the above three confusion matrix on test set, the test accuracy for Linear, polynomial and Rbf are **0.8523, 0.8519 and 0.8559** respectively. Hence, we can clearly state that for the Rain in Australia dataset, SVM\_Rbf function performs the best. When we plot the ROC (Receiver Operating Character) curve and calculate the AUC (Area Under Curve) for all the three kernel functions, we get the below graphs.

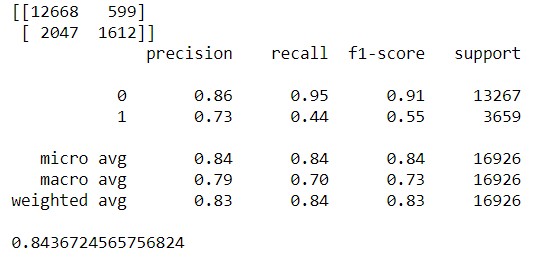
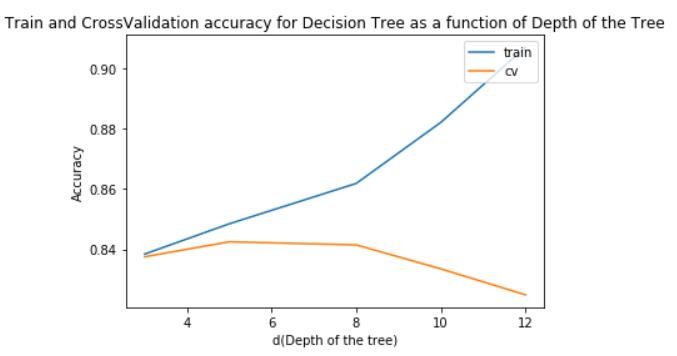


Based on Cv scores and Test scores, SVM\_Rbf performs a little bit better than SVM\_Linear. Both SVM\_Rbf and SVM\_linear performs better than SVM\_polynomial. On the Roc Curve, SVM Linear (Auc=0.73) performs better than SVM Rbf (Auc=0.72) after 3 decimals. On overall basis, we can conclude that both SVM Linear and SVM Rbf performs similarly on dataset with rbf performing slightly better in generalizing the dataset. We can say that both a linear hyperplane and a plane made my normalized cumulative curve of all the support vector points perform similarly on the dataset.

**Learning Algorithm 2: Decision Tree:**

When we run the decision tree without any depth criteria, the cv and train score are 0.79 and

1.0. From train score, we can clearly say that the model has fully overfit the data. The value of Train score is 1 because it has numerous continuous attributes. Hence, we do pruning with various depths to find the optimum depth at which the model generalizes the data well.

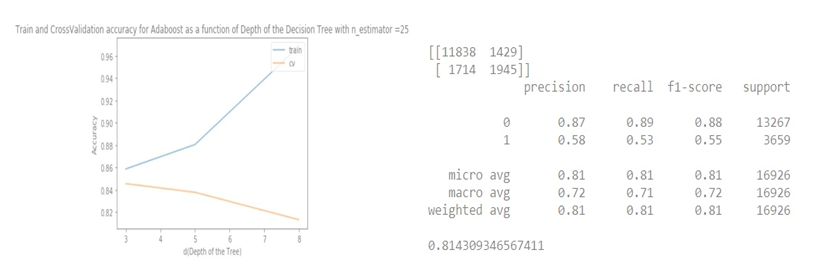


The above graph(left) gives the Cross-validation (Cv) and train scores as a function of depth of the tree and confusion matrix(right) for test set when depth =5. From the graph, we can conclude that when depth=5, we get the maximum Cv score. When we increase the depth more than 5, the model overfits the data. That can be clearly seen as Cv score goes down and train score goes up. The accuracy on the test set for decision tree at depth 5 = **0.8436**.

**Learning Algorithm 3: Adaboost with Decision Tree:**

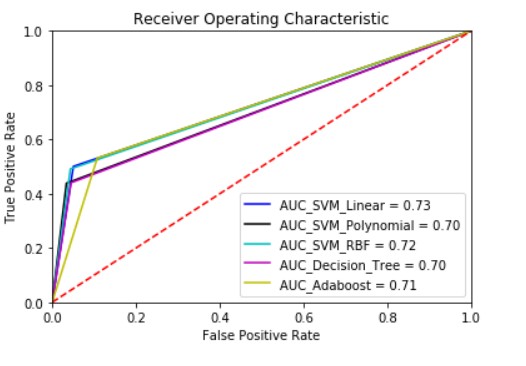
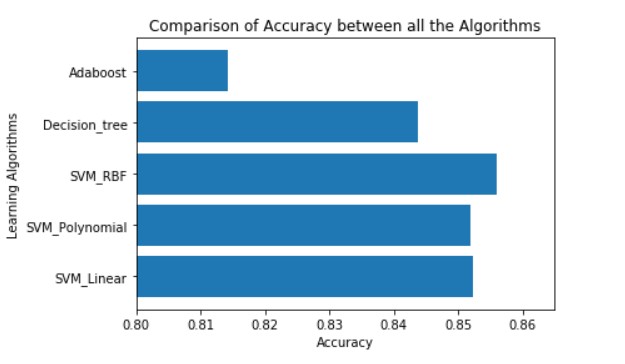
We perform the same decision tree algorithm in Adaboost method with n\_estimators as 25 and

50 for various depths of the tree. For the both the n\_estimators, model is performing similarly. We will try to find the optimum depth at which the model does not overfit or underfit the data. The below graph(left) gives the Cross-validation (Cv) and train scores as a function of depth of the tree and confusion matrix(right) for test set when depth =3.



***Comparison between all the Learning Algorithms*:**

By using the best hyperparameters from all functions, we have predicted the test set and computed the accuracy on all the functions. The below bar chart (Left) shows the Accuracy of all the functions and the graph (Right) ROC curve with AUC for all the functions based on the predicted Test set values.



From the bar chart, it is clear that SVM Rbf is the best model for the dataset based on the Test Accuracies. In the ROC curve, both SVM Linear (AUC=0.73) and SVM Rbf (AUC=0.72) performs like each other. Adaboost is performing poorly on unseen data as it has overfit the training data. Decision tree also overfits the data as it can not able to perform well on the unseen data. In SVM, The higher dimensional model made through normalized curve for support vectors classifies the data in the best possible way.

**Conclusion:**

The final verdict is that for the current dataset, SVM Rbf function performs the best when gamma is 0.1. It gives a test accuracy of 0.86 (86%) with an AUC of 0.72.

***Additional things that can be improved****:*

➢ *The selection of Independent variables on the both the datasets. Special functions like PCA could have been used to access the important of the variables and select accordingly.*

➢ *For the dataset Rain in Australia, Missing values could have been imputed rather than dropping them and tried the models again*.

➢ *Could have done grid search for the optimum gamma and c values by plotting a heatmap between them.*