Assignment 2

A. General Guidelines about Models used in the Report:

Six different classification models are used to predict the target classes. Three of the 6 models are Support Vector Classifiers with Linear, Radial, and Polynomial kernels. Remaining three classifiers are versions of decision trees, namely – Decision Tree Classifier, Decision Tree Classifier with Cost-Complexity Pruning, and AdaBoost Decision Tree Classifier.

Model Selection Process:

Grid Search Cross Validation is used to select the best hyperparameters that lead to a lower 5-Fold Cross-Validation Error Rates. Below table summarizes the various hyperparameters used in the 6 models.

Classification Model	**Hyperparameters**
SVM: Linear	Regularization Parameter (C)
SVM: Radial	C, Gamma (influence of a single training example)
SVM: Polynomial	C, Gamma, Degree (of Polynomial)
Decision Tree	Split Criterion, Maximum Depth, Minimum no. of samples required to split
Decision Tree with Cost-Complexity Pruning	Cost-complexity Pruning (CCP) Alpha
AdaBoost Decision Tree	No. of Estimators, Learning Rate

Model Evaluation:

The final selected 6 models are evaluated using test data to find the best classifier which maximizes area under the Receiver Operation Curve and follows Occam's Razor Principle.

B. Bike Sharing Dataset:

The Seoul Bike dataset contains information regarding number of bikes rented on an hourly basis for a year and includes prevailing weather conditions. The aim is to predict whether the bike rentals on a particular hour of a day are higher than the median value.

B1. EDA and Data Pre-processing:

Exploratory data analysis has resulted in following implications which are considered for modelling:

- The 'hour' variable is considered as a categorical variable and is included in our models.
- We do not need to include 'functioning_day' while training our models. On those days we do not to predict the number of bikes rented.
- The dataset does not contain day of week as variable. Hence a new variable is created to include in machine learning models.
- Dew point temperature is highly correlated with temperature (0.91) and including both these variables might lead to multi-collinearity issues while modelling. In addition, the VIF of all variables is less than 10 after removing dew-point temperature. Hence, dew point temperature is not included in the model building.
- The dataset is balanced.

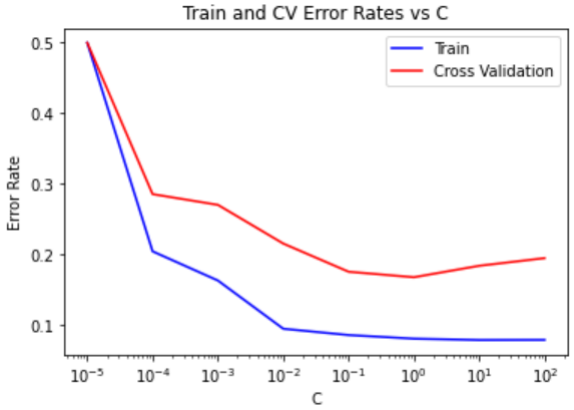
Final Dataset:

	bike_count	hour	temperature	humidity	wind_speed	visibility	solar_radiation	rainfall	snowfall	season	holiday	day
0	254	0	-5.2	37	2.2	2000	0.0	0.0	0.0	Winter	No Holiday	Friday
1	204	1	-5.5	38	0.8	2000	0.0	0.0	0.0	Winter	No Holiday	Friday
2	173	2	-6.0	39	1.0	2000	0.0	0.0	0.0	Winter	No Holiday	Friday

B2. Linear Kernel SVM Classification:

Model Selection process:

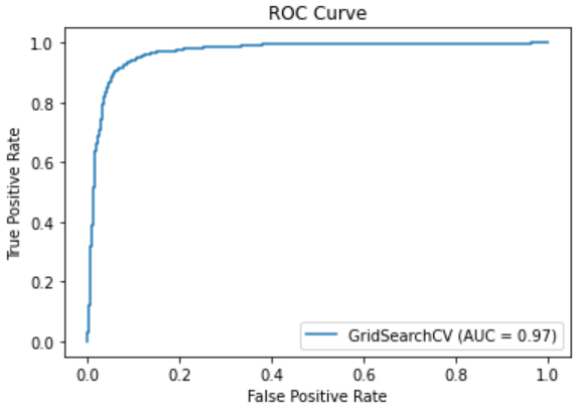
Hyperparameter: C = [0.00001, 0.0001. 0.001, 0.01, 0.1, 1, 10, 100]

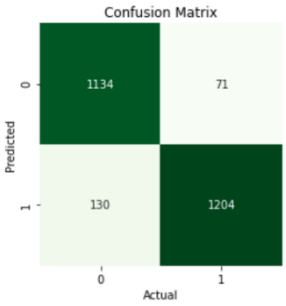


C controls the bias-variance trade-off. When C is small, both train and CV errors are high suggesting high bias and low variance as the margin is higher. On the other hand, when C is larger, the margin is smaller as bias is reduced since both Train and CV errors reduced.

Chosen parameters: C = 1.0. The average CV accuracy is 0.84.

Model Evaluation:





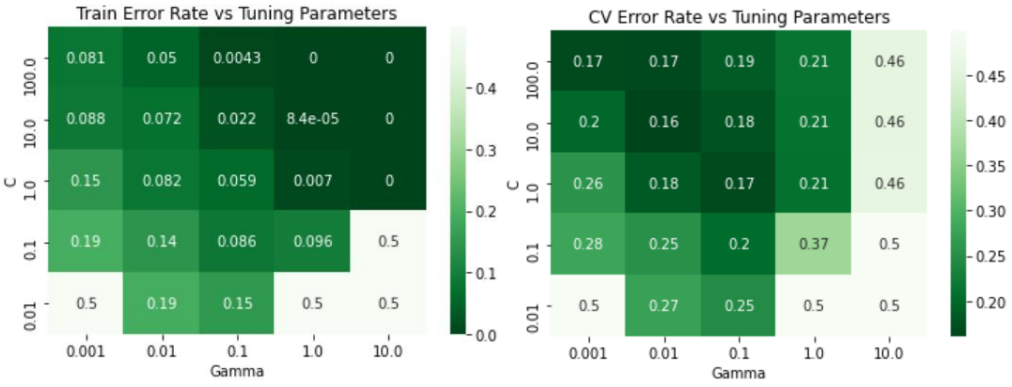

```
Accuracy: 0.92
Sensitivity: 0.94
Specificity: 0.90
```

The linear kernel SVM does a pretty good job in classifying the target variable as test accuracy is over 0.9. The area under the curve suggests that it is a very good classifier.

B3. Radial Kernel SVM Classification:

Model Selection process:

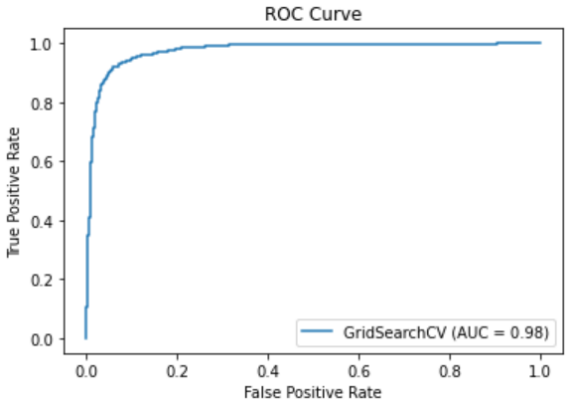
Hyperparameter: C = [0.01, 0.1, 1, 10, 100] and Gamma = [0.001, 0.01, 0.1, 1, 10]

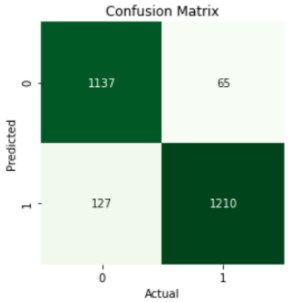


Similar to the discussion of C in SVM with linear kernel and the train and CV errors heatmap, lower values of C lead to both high train and CV errors (models suffering from high bias). But as C becomes very high, train errors have dropped whereas CV errors have increased (models suffering from high variance). In terms of Gamma, when it is too large, the radius of the area of influence of the support vectors only includes the support vector itself and no amount of regularization with C will be able to prevent overfitting. When gamma is very small, the model is too constrained and cannot capture the complexity or shape of the data.

Chosen parameters: C = 10 and Gamma = 0.01. The average CV accuracy is 0.84.

Model Evaluation:





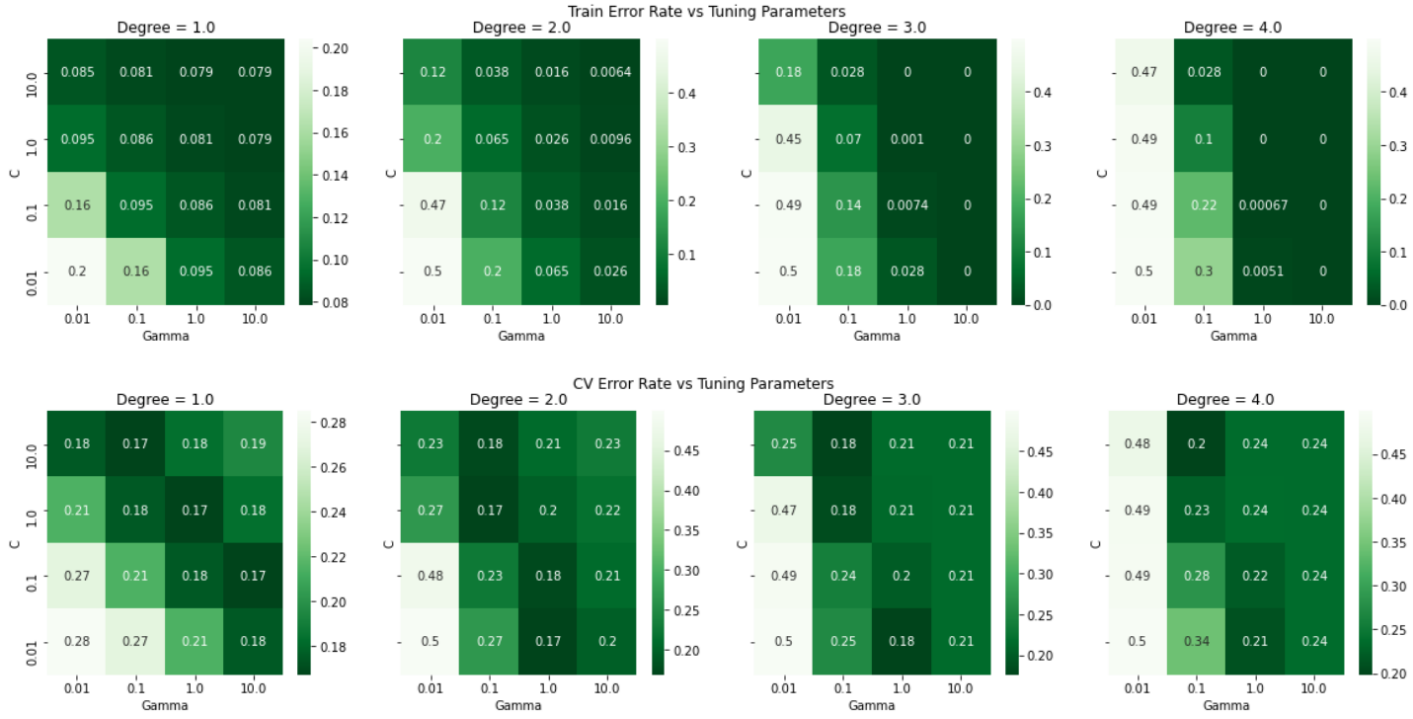
Accuracy: 0.92
Sensitivity: 0.95
Specificity: 0.90

The radial kernel SVM did a good job in classifying the target variable as test accuracy is over 0.9. However, the increase in the accuracy is same as that of linear kernel SVM. This shows that the data is linearly separable.

B3. Polynomial Kernel SVM Classification:

Model Selection process:

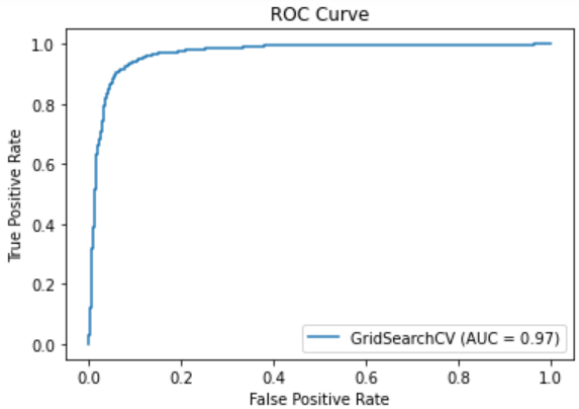
Hyperparameter: C = [0.01, 0.1, 1, 10], Gamma = [0.01, 0.1, 1, 10] and Degree = [1, 2, 3, 4]

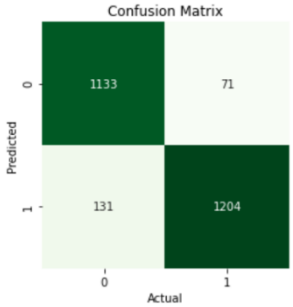


Higher polynomial orders have overfit the data as the train error becomes zero and the CV errors increase for polynomials beyond the optimal degree.

Chosen parameters: C = 0.1, Gamma = 1.0 and Degree = 1. The average CV accuracy is 0.83.

Model Evaluation:





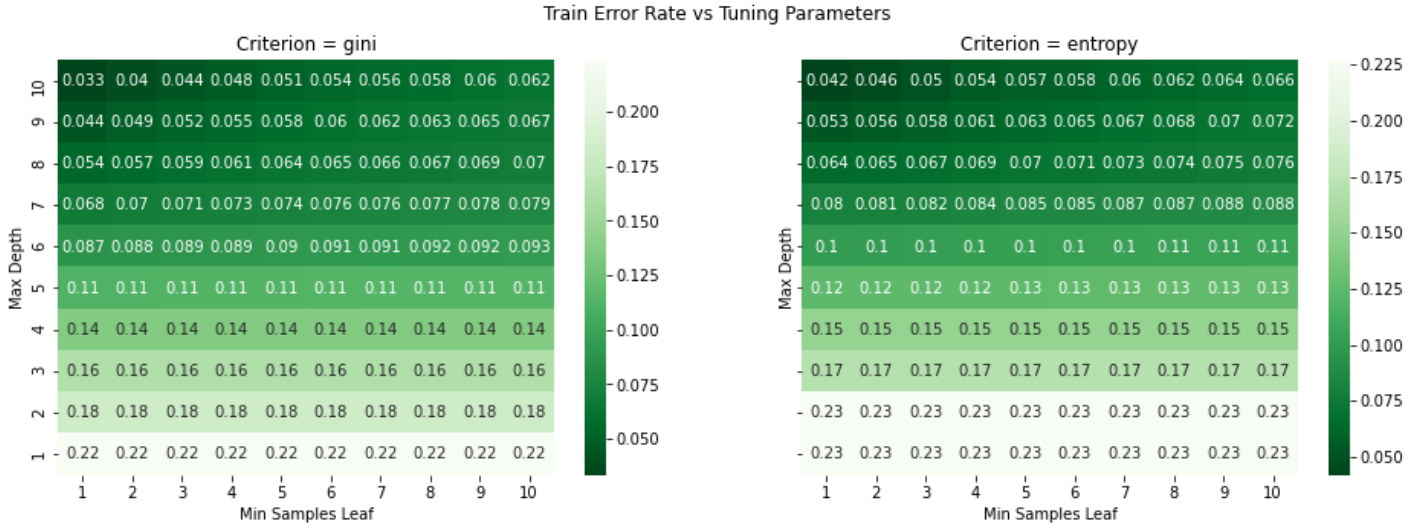
Accuracy: 0.92
Sensitivity: 0.94
Specificity: 0.90

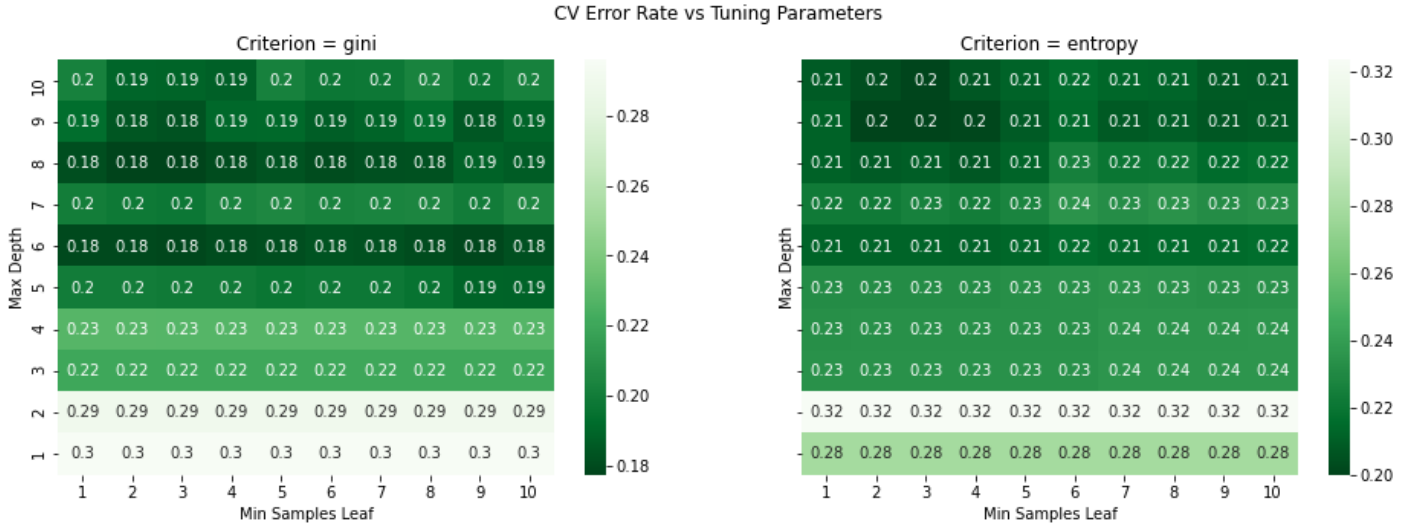
There is no change in the accuracy for the chosen parameters. Since the data is linearly separable, higher order polynomial kernel has resulted in higher error rates. In fact, the chosen degree is 1.

B4. Decision Tree Classification:

Model Selection process:

Hyperparameter: Criterion = [Gini, Entropy], Max Depth = [1 to 10] and Min Samples Leaf = [1 to 10]

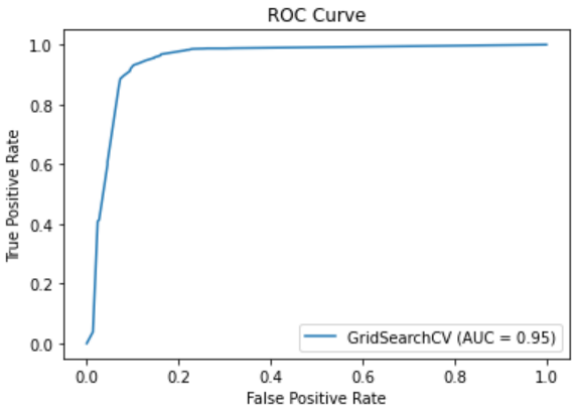


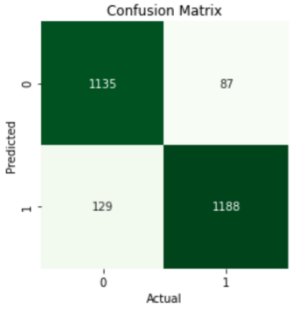


Compared to Entropy, Gini is a better criterion for splitting as both the train and CV errors are lower. A single deeper tree will overfit the data whereas a shallow tree might underfit the data.

Chosen parameters: Criterion = Gini, Max Depth = 8 and Max Samples Leaf = 2. The average CV accuracy is 0.82.

Model Evaluation:





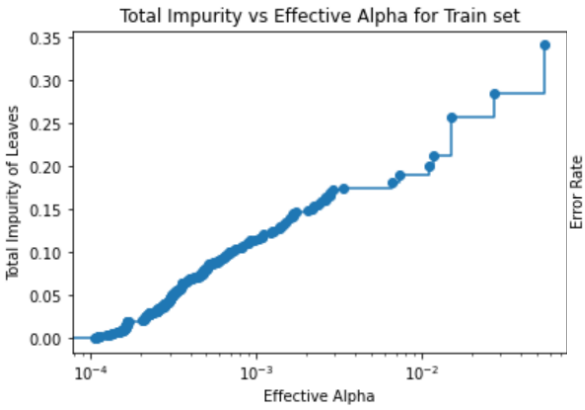
Accuracy: 0.91
Sensitivity: 0.93
Specificity: 0.90

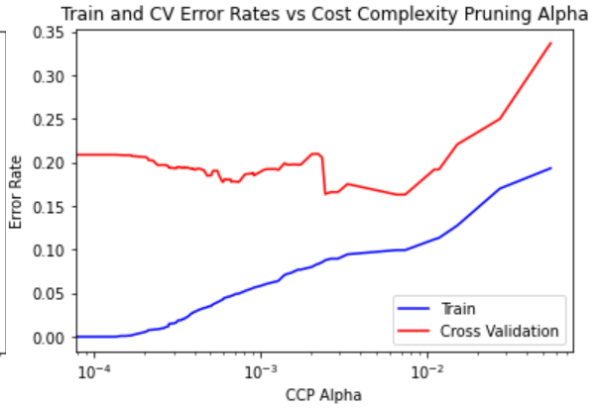
Decision Tree Classifier did a decent job in classifying the target class with accuracy over 0.9. The area under the curve is in similar range to that of Support Vector Machine Models. Decision tree provides an advantage of easily explainable model which are based on rule-based splits. However, the complexity increases with depth.

B5. Decision Tree Classification with Cost Complexity Pruning:

Model Selection process:

Hyperparameter: CCP Alpha

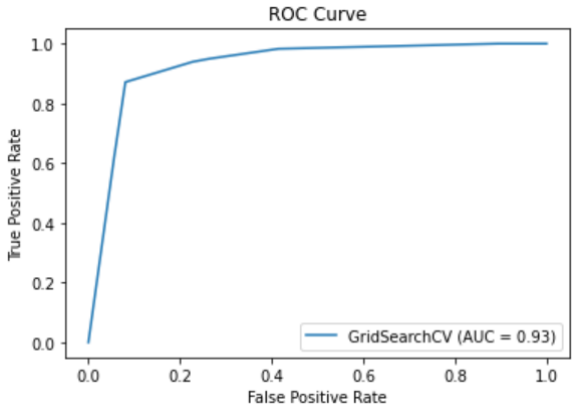


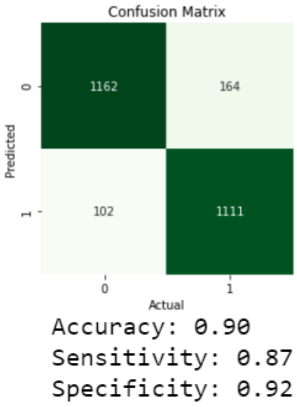


Minimal cost complexity pruning recursively finds the node with the "weakest link". The weakest link is characterized by an effective alpha, where the nodes with the smallest effective alpha are pruned first. As alpha increases, more of the tree is pruned, which increases the total impurity of its leaves. Models with alpha less than the optimum value suffers from high variance and greater than the optimum suffers from high bias.

Chosen parameters: CCP Alpha = 0.0066. The average CV accuracy is 0.84.

Model Evaluation:





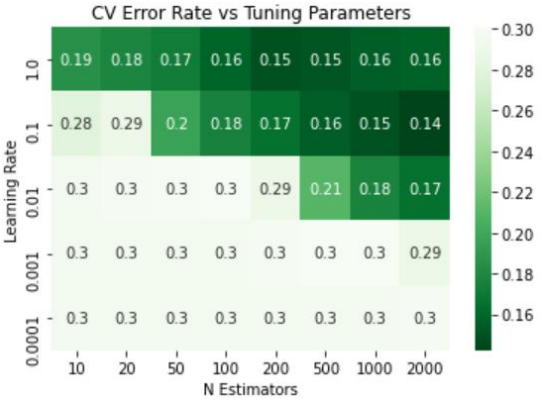
Accuracy: 0.90
Sensitivity: 0.87
Specificity: 0.92

Post pruning after completely growing the tree has resulted in test accuracy of 0.90. It is slightly lower than that of decision tree with grid search selected parameters but it is very comparable with the results. The area under the curve is lowest among all the other models with 0.93.

B6. AdaBoost Decision Tree Classification:

Model Selection process:

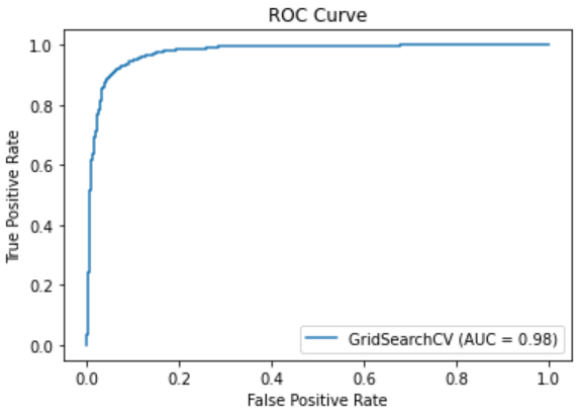
Hyperparameter: N Estimators = [10, 20, 50, 100, 200, 500, 1000, 2000], Learning Rate = [0.0001, 0.001, 0.01, 0.1, 1]

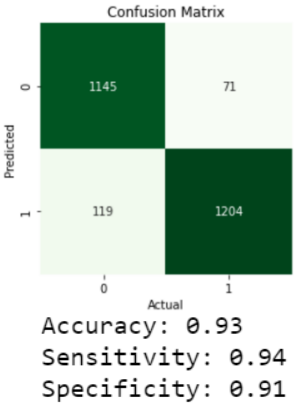


Mutiple weak decision tree classifiers with depth of 1 are iteratively added to create a strong classifier. In each step the examples that are misclassified gain weight and those that are classified correctly lose weight. As the number of estimators increase, the train and CV errors continue to decrease showing that the model doesn't suffer from high variance (bias-variance trade-off) with increasing estimators. The learning rate works exactly opposite to the number of estimators. There is a trade-off between number of estimators and learning rate.

Chosen parameters: N Estimators = 1000 to 2000, Learning Rate = 0.1. The average CV accuracy is 0.86.

Model Evaluation:





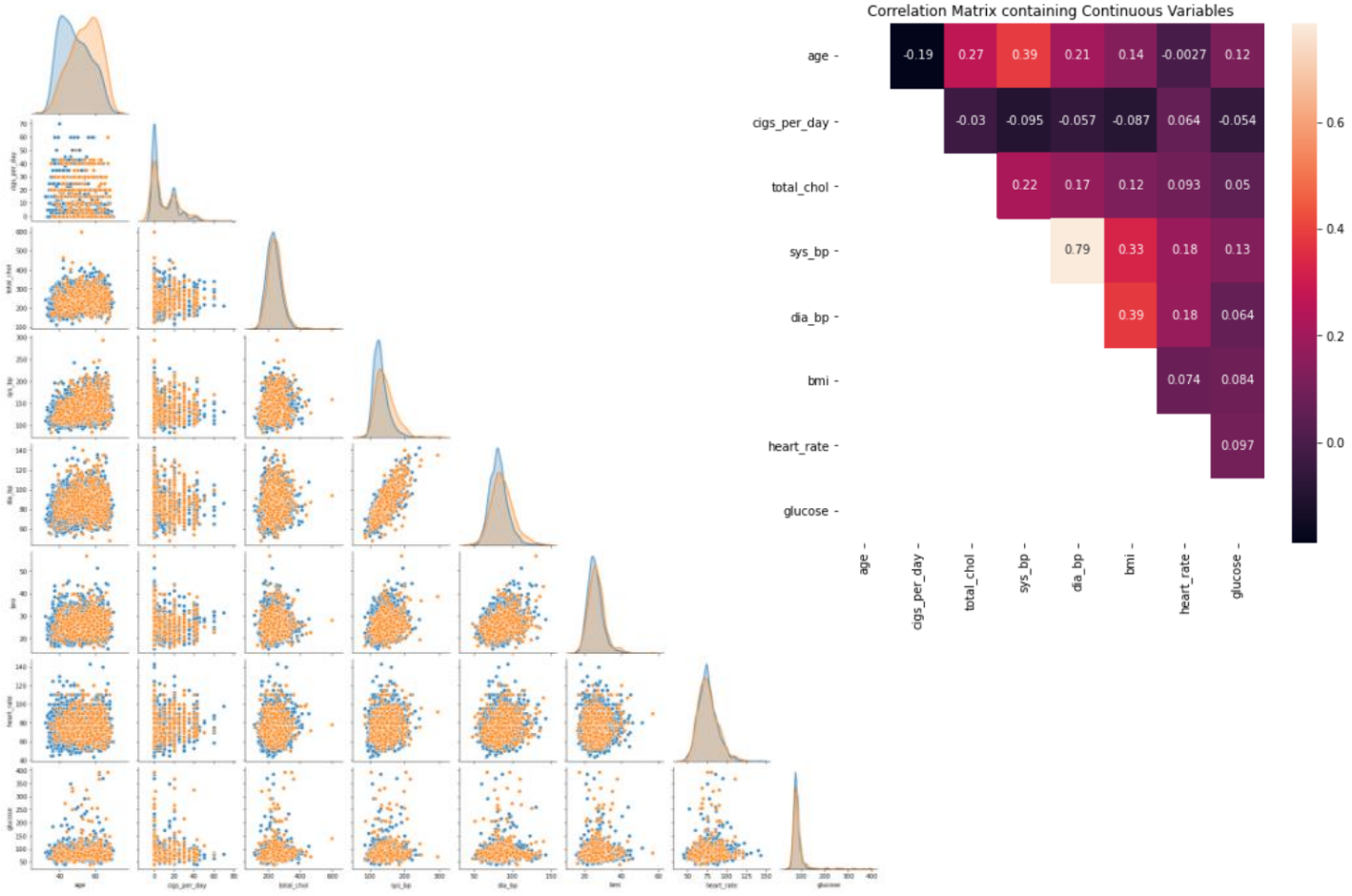
The Adaboost Decision Tree Classifier has an accuracy of 0.93 which is in the same range as the other classifiers. The explanability power however reduces greatly compared to that of other decision tree models.

C. Chronic Heart Diseases Dataset:

C1. EDA and Data Pre-processing:

The chronic heart diseases dataset contains information regarding the chance of contracting a chronic heart disease after 10 years based on following parameters:

- Demographic: male, age, education
- Behavioral: current_smoker, cigs_per_day
- Medical (history): bp_meds, prevalent_stroke, prevalent_hyper, diabetes
- Medical (current): tot_chol, sys_bp, dia_bp, bmi, heart_rate, glucose

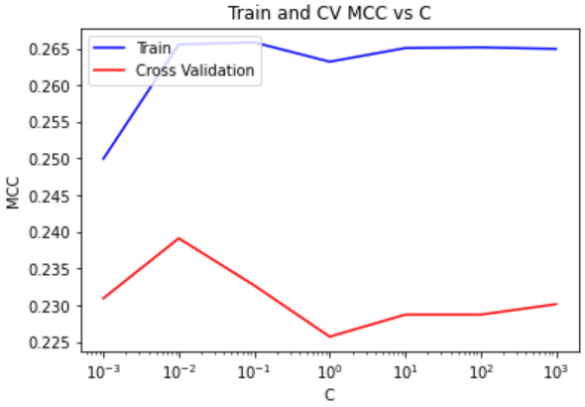


None of the continuous variables are correlated as the highest VIF is less than 10. Hence, we use all the variables in modelling. In the pair-plots, the blue colored graphs represent samples who are not likely to get heart disease and the orange ones represent the samples who are likely to get heart disease. The distribution of most numerical variables looks the same for both the classes except ‘age’.

The dataset is unbalanced with 3101 negative class records out of 3658. In such a case, model evaluation using accuracy would be misleading. One needs to investigate sensitivity and specificity or misclassification costs to select and evaluate models. For such scenarios, Matthews Correlation Coefficient (MCC) is a better model scoring method. It considers true and false positives and negatives and is generally regarded as a balanced measure which can be used even if the classes are of very different sizes. The MCC is in essence a correlation coefficient value between -1 and +1. A coefficient of +1 represents a perfect prediction, 0 an average random prediction and -1 an inverse prediction. The statistic is also known as the phi coefficient. Suitable class weights must be given to both target variables for better classification.

C2. Linear Kernel SVM Classification:

Model Selection process:

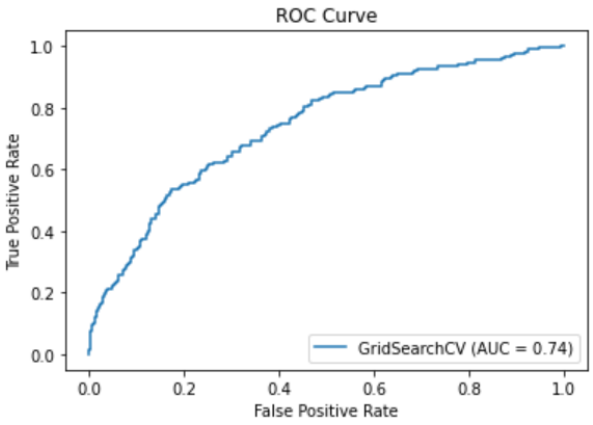


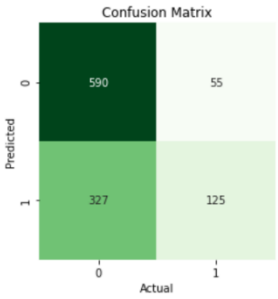
Hyperparameter: C = [0.00001, 0.0001. 0.001, 0.01, 0.1, 1, 10, 100]

When C is small, the margin is bigger. MCC of train and CV follow the same trend for C below optimal value. As the value of C increases, the train and cross-validation MCC vary a lot suggesting overfitting above optimal value.

Chosen parameters: C = 0.01. The average CV MCC is 0.24.

Model Evaluation:





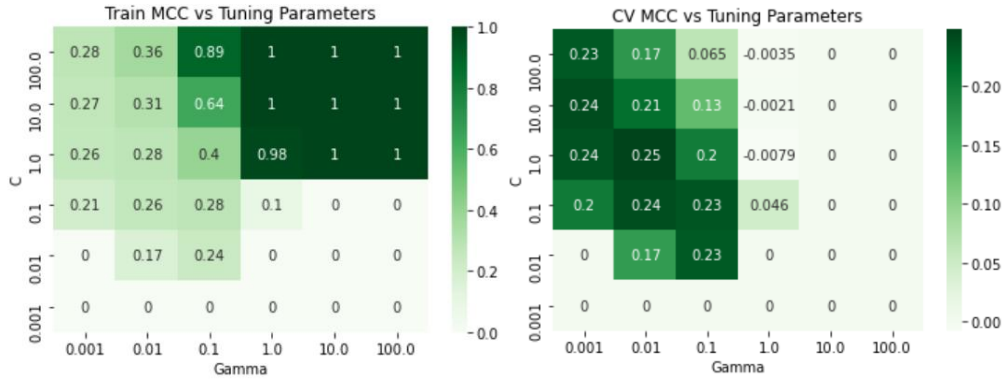
Accuracy: 0.65
Sensitivity: 0.69
Specificity: 0.64

The area under the curve is 0.74. By giving the right weights to classes and using MCC as scoring, the classifier using linear kernel SVM did a decent job for this data.

C3. Radial Kernel SVM Classification:

Model Selection process:

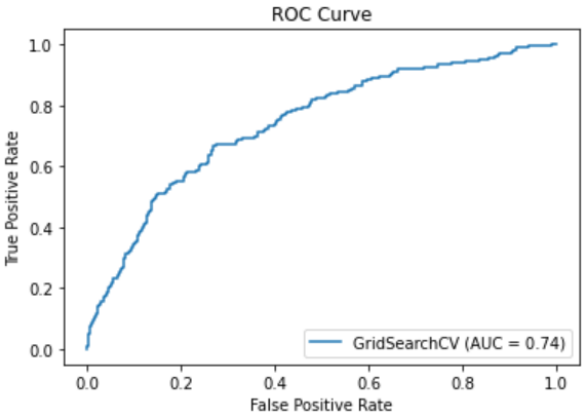
Hyperparameter: C = [0.01, 0.1, 1, 10, 100] and Gamma = [0.001, 0.01, 0.1, 1, 10]

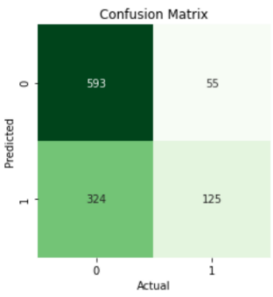


As C and Gamma increases, the training MCC becomes 1. However, the cross validation MCC reduces towards the top right of hyperparameter grid. The second plot suggests that the apt parameters are found in the top left of hyperparameter grid.

Chosen parameters: C = 1.0 and Gamma = 0.01. The average CV MCC is 0.25.

Model Evaluation:





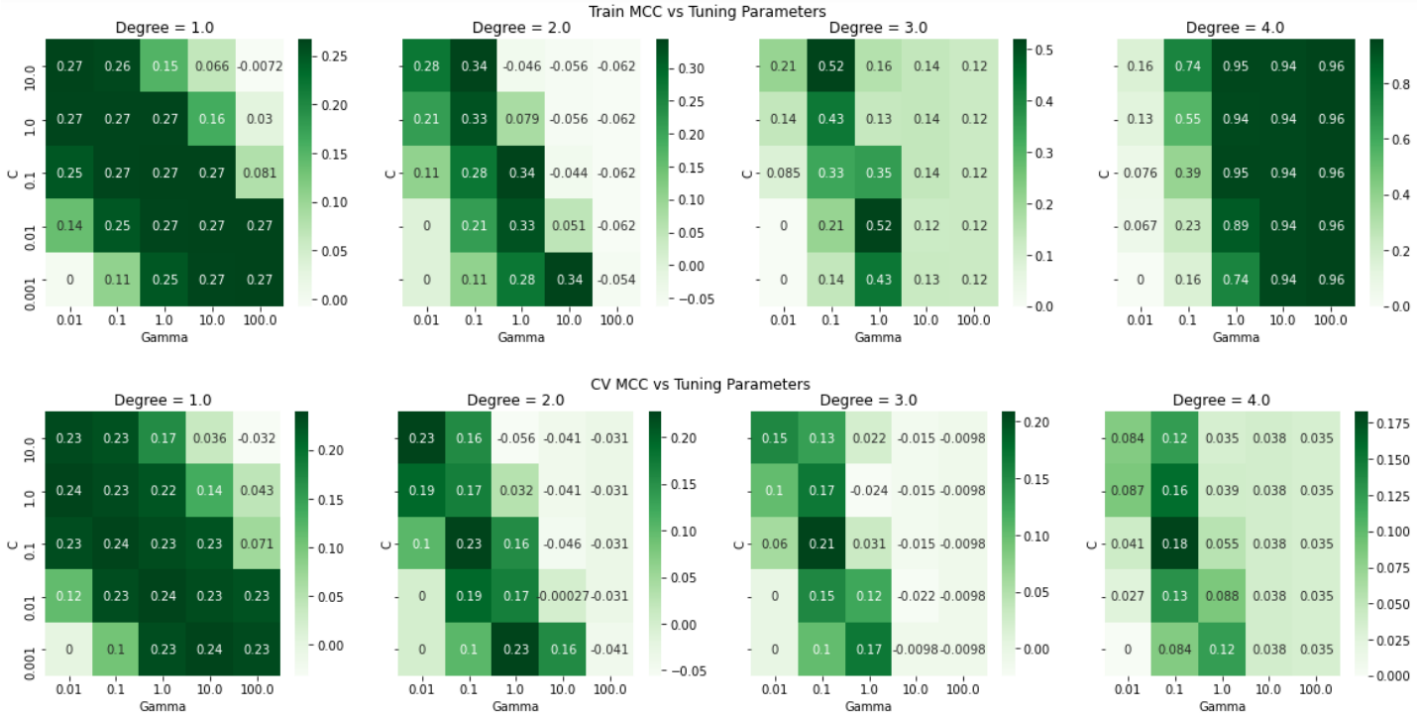

```
Accuracy: 0.65
Sensitivity: 0.69
Specificity: 0.65
```

The area under the curve is same as that of linear kernel SVM Classifier. Increasing to 2nd order polynomial has not improved the MCC performance. This validates that with appropriate class weights and MCC as scoring parameter, the linear kernel SVM does the best job.

C3. Polynomial Kernel SVM Classification:

Model Selection process:

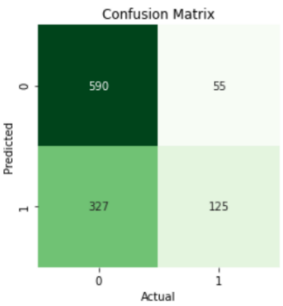
Hyperparameter: C = [0.001, 0.01, 0.1, 1, 10], Gamma = [0.01, 0.1, 1, 10, 100] and Degree = [1, 2, 3, 4]



Due to the non-normality of data, the grid search takes lot of time/iterations to compute optimal solution. Hence the maximum iterations are capped at 10000. For higher order polynomials, one can see that training MCC increases but CV MCC reduces greatly suggesting higher polynomials overfitting the data. The best performance on CV data comes from lower order polynomials suggesting the same inference about the data derived from the last two models.

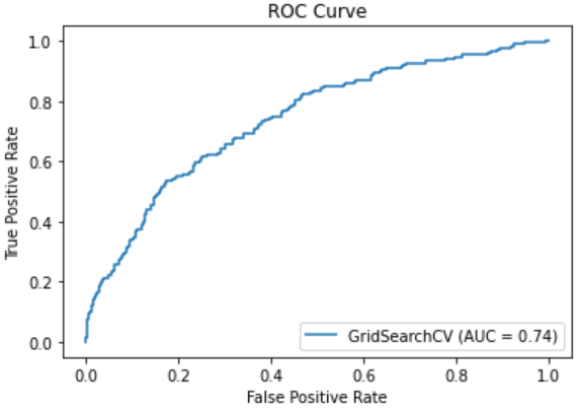
Chosen parameters: C = 0.001 and Gamma = 10.0 and Degree = 1. The average CV MCC is 0.24.

Model Evaluation:



The automatic selection method for SVM using polynomial kernel has chosen the first order poynomial as the best performing SVM model.

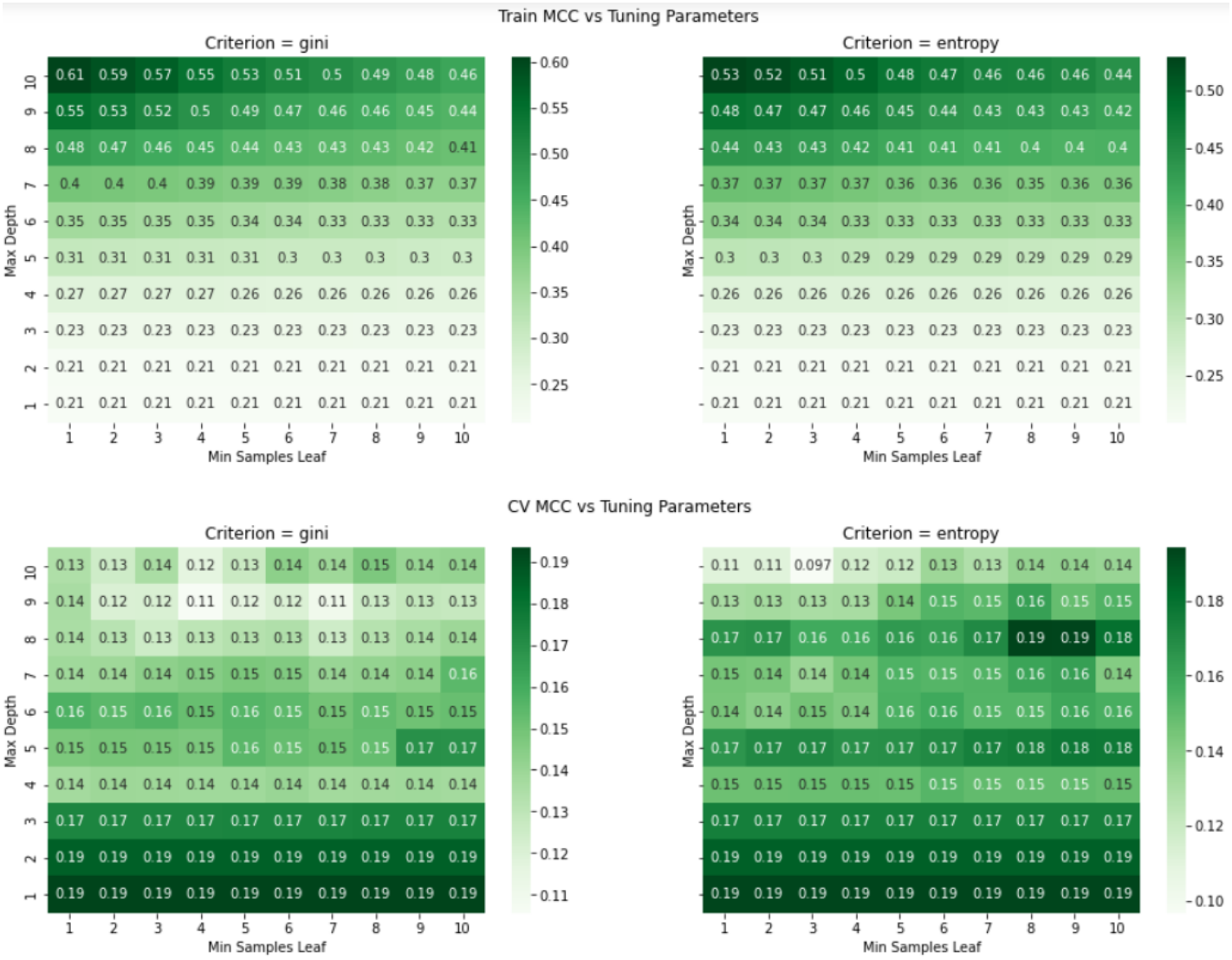
```
Accuracy: 0.65
Sensitivity: 0.69
Specificity: 0.64
```



C4. Decision Tree Classification:

Model Selection process:

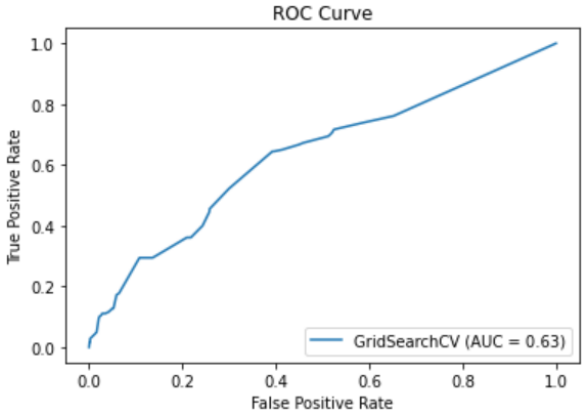
Hyperparameter: Criterion = [Gini, Entropy], Max Depth = [1 to 10] and Min Samples Leaf = [1 to 10]

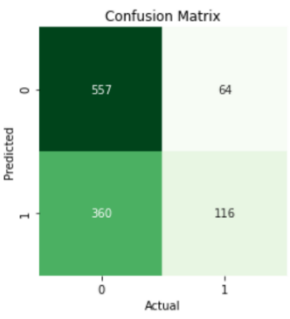


As the depth increases and minimum samples at split decreases, the train MCC decreases but the CV MCC increases. The first split is on variable 'age' which validates the pair-plot distribution of age for both the target classes. Compared to Gini, Entropy is a better criterion for splitting as both the train and CV MCC are greater.

Chosen parameters: Criterion = Entropy, Max Depth = 8 and Max Samples Leaf = 8. Average CV MCC is 0.19.

Model Evaluation:





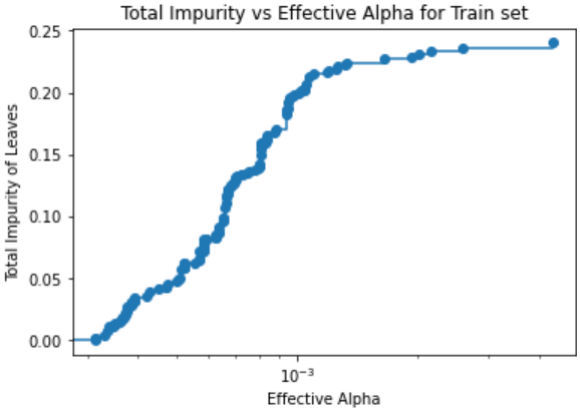

```
Accuracy: 0.61
Sensitivity: 0.64
Specificity: 0.61
```

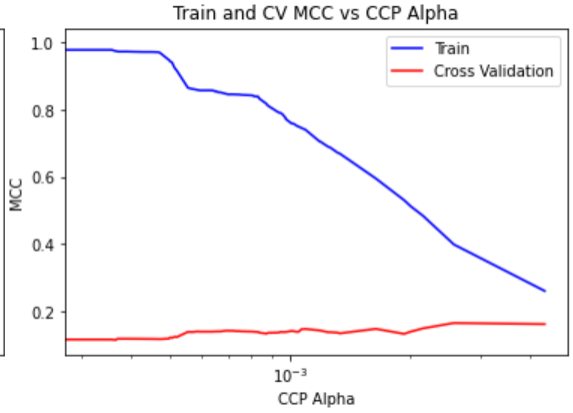
Due to the imbalance nature of the dataset, the threshold needs to be changed to have a balance between sensitivity and specificity. The decision tree does a decent job on the test sample. However, the area under the curve of decision tree classifier is lower compared to that of SVM models.

C5. Decision Tree Classification with Cost Complexity Pruning:

Model Selection process:

Hyperparameter: CCP Alpha

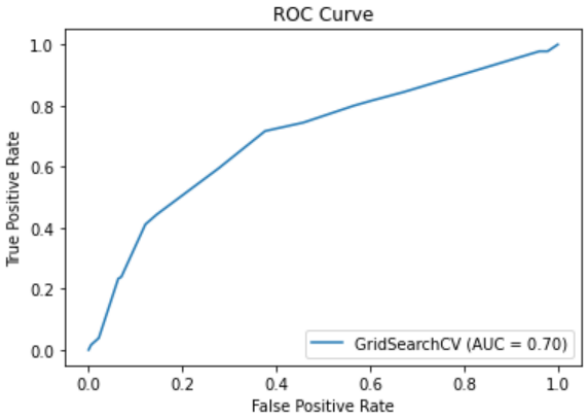


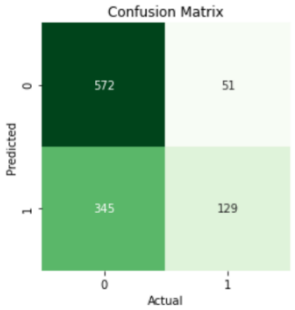


Models with CCP alpha less than the optimum value lead to greater train MCC showing perfect fit. But this resulted in overfiiting with lower CV MCC. As a result, at lower alphas, the model suffers high variance.

Chosen parameters: CCP Alpha = 0.0026. The average CV MCC is 0.17.

Model Evaluation:





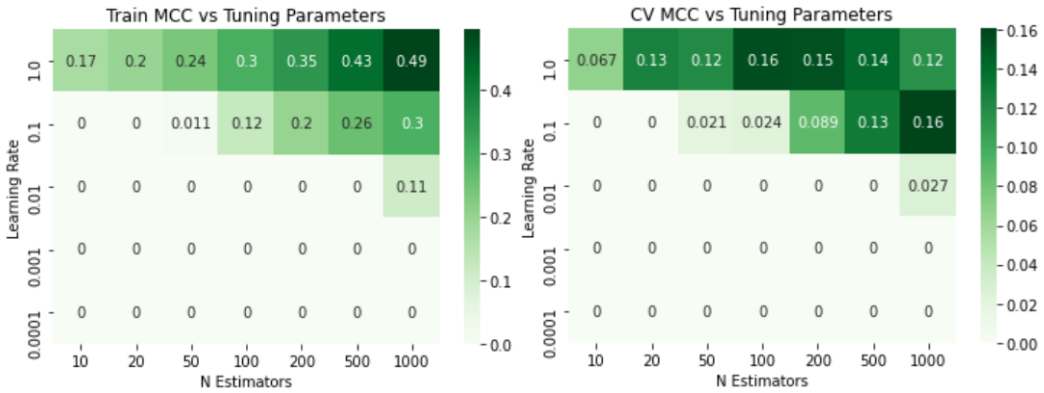
Accuracy: 0.64
Sensitivity: 0.72
Specificity: 0.62

The classification metrics are similar to that of previous decision tree. The higher CCP alpha suggests that most of the decision tree is pruned. In a way, only certain predictors are important.

C6. AdaBoost Decision Tree Classification:

Model Selection process:

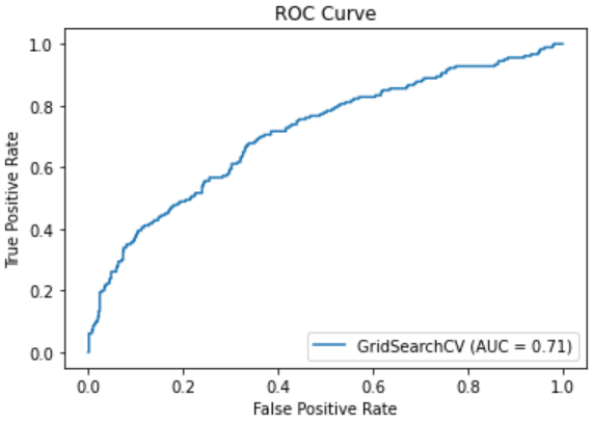
Hyperparameter: N Estimators = [10, 20, 50, 100, 200, 500, 1000] and Learning Rate = [0.0001, 0.001, 0.01, 0.1, 1]

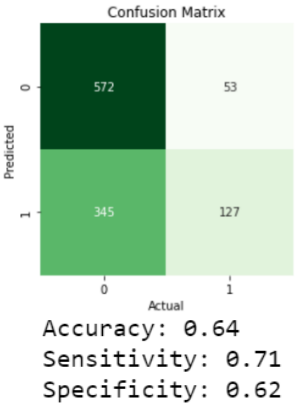


As N Estimators increases, the CV errors decrease to a level but after that it overfits the data. The reverse is true with the learning rate.

Chosen parameters: N Estimators is between 1000 to 2000, Learning Rate = 0.1. The average CV MCC is 0.16.

Model Evaluation:





Accuracy: 0.64
Sensitivity: 0.71
Specificity: 0.62

Sensitivity is 0.71 and specificity is 0.64 at 0.495 threshold level. The AdaBoosted decision tree does the best classification compared to other decision tree classifiers. The area under the curve is maximum at 0.71. However, the performance of SVM models are superior to the decision tree models.

D. Conclusion:

The two datasets have different properties. The Bike dataset is balanced and linearly separable. Therefore, linear kernel SVM Classification did a very good job. By increasing the polynomial order, it only improved train errors but did not improve the CV errors significantly. The AdaBoosted Decision Tree Classifier performed the best in terms of decision tree models. As N Estimators increased, both the train errors and CV errors have reduced. In terms of performance, all the classifiers are similar. However, for better interpretability based on Occam's Razor Principle, logistic regression model should be chosen which has similar test accuracy, sensitivity, and specificity.

The heart disease dataset is an unbalanced dataset and only few predictors are strong enough to explain the class labels. For such a scenario, model evaluation using accuracy would be misleading. Sensitivity and Specificity are better metrics. We have used Matthew's Correlation Coefficient as a scoring method which gives equal weightage to sensitivity and specificity. Higher polynomial kernel SVM overfit the training data and hence it did not lead to good models. Decision trees did a good job in classifying the dataset and identifying important predictors. The AdaBoosted Decision Tree Classifier led to the best area under the curve with intermediate sensitivity and specificity level for a decision tree model. With appropriate class weights and MCC as scoring parameter, Support Vector Classifier with Linear Kernel did the best job.