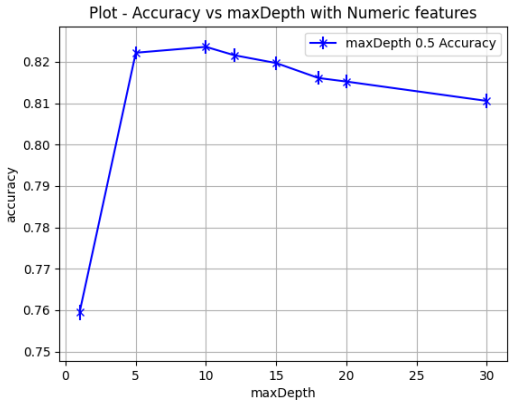
Submitting late by one day because of stress due to election week.

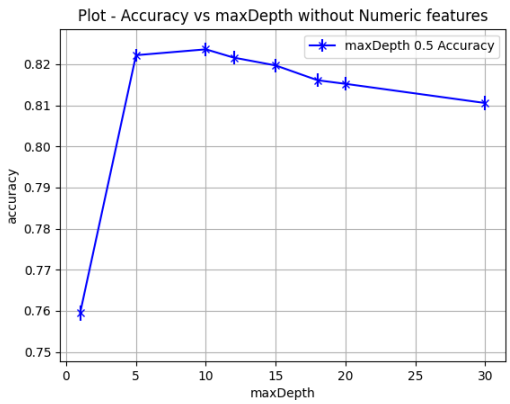
2 Points - Your implementation: DecisionTree.py. Make sure it's very easy for the TA to find the critical parts of the code:
such as the core recursion logic, the maxDepth, the entropy calculation, the selection of the split feature & threshold.

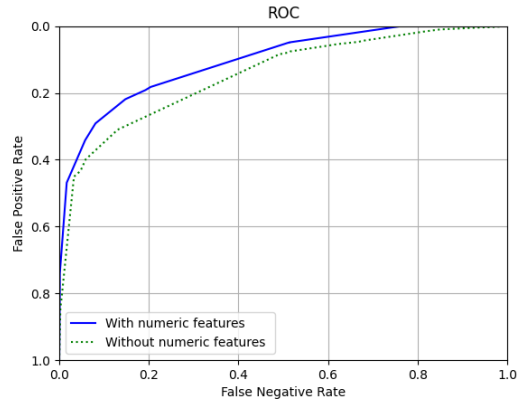
See Attached File

4 points - Tune maxDepth on the adult data set with and without numeric features: featurizer.CreateFeatureSet(xTrainRaw, yTrain, useCategoricalFeatures = True, useNumericFeatures = True)

Create a short (~3-4 figures and 200 words) writeup of what you learned from the tuning runs. Include all the usual elements (bounds, parameter sweeps, ROC curves). Is it better to use the numeric features or not? (Be precise)







The cross validation accuracy of decision tree based model reaches a maximum at about maxDepth = 10, after which it starts decreasing. The best cross validation accuracy of the model with numeric features is 82.22% (50% confidence interval 82.06% - 82.38%) that is achieved with a sweep on the parameter maxDepth = 5. Interestingly, the model without numeric features also has cross-validation accuracy 82.22% (50% confidence interval 82.06% - 82.38%) with maxDepth = 5. However, the accuracy of the model with numeric features is higher (84.44% 50% confidence interval 84.01% - 84.87%) on the test set than the accuracy of the model that does not use the numeric features (81.30% 50% confidence interval 80.84% - 81.77%). Just looking at the cross-validation accuracy I cannot say which model performs better. However, the test set results speak that the model with numeric features performs better than the one without it with atleast 75% confidence. The decrease in cross-validation accuracy for higher-maxDepths indicates the tendancy of a decision tree to overfit.