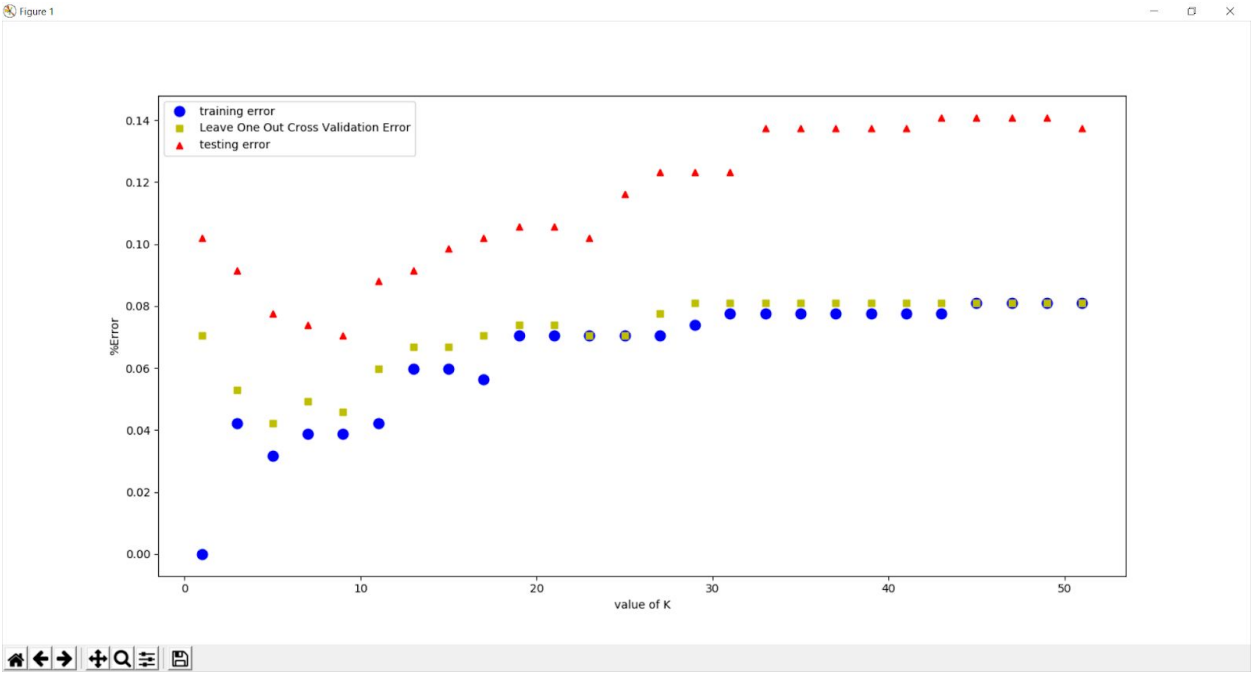
Danh Nguyen
Coulby Nguyen
Mingyu Zhang

Implementation Assignment 2

Part I

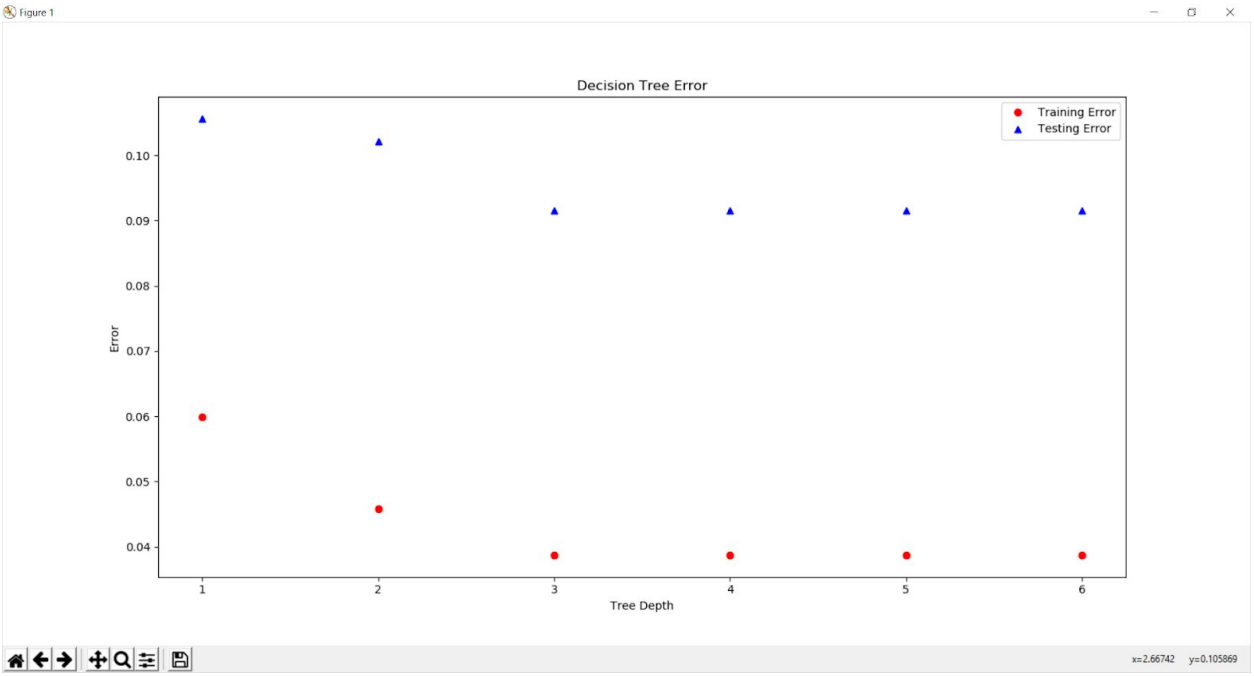


1) For the K-Nearest Neighbors algorithm, there is a clear trend that when K increases, the leave one out cross validation error, and the training error converge to the same value. This is due to the fact that leaving one sample out of training data when K is large has little effect on the overall vote. Another trend that should be noted is that when K increases, the testing error also increases. This is because with more votes that are coming from data points that may not be close, but rather they are relatively closer than others points, might yield inaccurate predictions. Using the model selection of leave one out cross validation, K = 5 is the best choice as it has the lowest leave one out cross validation. Using this K, the training error provides the second lowest out of the K in the set {1..51} and the testing error is the 3rd lowest.

Part II

1) For the learned decision stump, the result we got was that the data set needs to have Feature 23 < 115.7. And the reason this was the decision stump is because it gave our model the highest information gain of 0.001523. The training error that we got was 5.98% which is really good because that is around 94% accurate. On

the other hand the testing error was 10.6% which is only around 90% accurate. However the model perform decently because the data set was sufficient large; therefore, the result can be trusted.



2) As expected, the more we go in depth with our tree the better the model will be, which should result in a lower error for training data with a higher value of d. The reason this is the case is because as we partition our data into a tree we come up with questions that reduces the entropy of the data set, which leads to a more accurate decision that fits to the training data. We might have implemented our entropy function incorrectly, as instead of the training data going to 0 it converges to .0387. Ideally the training data should converge to 0 as the tree depth increases, as the tree overfits to the training data. But we are still able to see with larger tree depth the error in the testing data still remains. This shows that the tree works really well for the training data but works really poorly for testing data.