Robert Dryden

Q 1.1 Why does it make sense to discretize columns for this problem?

When using continuous data there are an excess number of variables and any quantitative analysis process such as using the Decision Tree process would take an excessive time to finish computing.

Q 1.2 What might be the issues (if any) if we DID NOT discretize the columns?

The act of discretizing the data reduces the number of possibilities the programming algorithm needs to account for - basically it simplifies the data - and the resulting analysis is 'good enough'. If continuous data could be used it would produce better results if was computationally feasible.

Q.7.1 Decision Tree Hyper-parameter variation vs. performance.

Altering the parameters does change the performance of the tree fitting. By careful individual progression though parameters one at a time some semblance of optimization can be obtained. The process is rapid if only one parameter is changed at time. The assumption is that individualized parameter optimization results in overall optimization when combined, but this assumption may be unfounded.

Q.8.1 How long was your total run time to train the model?

Runtimes were rapid using Jupyter Notebook. All training was complete in order or seconds. Even multiple sequential training processes were rapid. The slowest part of the process was rendering the PNG tree map.

Q.8.2 Did you find the BEST TREE?

I found a reasonably good tree with a ROC of 0.8395320915530495. Is unlikely that this is the best tree possible and it may not be possible to determine the ideal best fit has been achieved. Using several trial and error, progressive, and optimizing routines the tree model presented is simply the best tree that I have found.

Q.8.3 Draw the Graph of the BEST TREE Using GraphViz

NOTE: Resultant tree map was too large for this document. Please see png image on IPYNB file submitted.

Q.8.4 What makes it the best tree?

I used the best ROC_AUC value found as the basis for my Tree optimization.

Q.10.1 What is the probability that your prediction for this person is accurate?

From the internet: “the AUC of a classifier is equal to the probability that the classifier will rank a randomly chosen positive example higher than a randomly chosen negative example”.

If reading this correctly, my model has an ~ 83.95% of correctly identifying a true positive as a positive value.