Supervised Learning

Justin Duan

Classification problem

For this assignment, I chose the two following datasets:

Titanic Dataset: This is a dataset regarding the RMS Titanic, one of the most infamous shipwrecks in history. The data represents around 900 passengers, providing us with their Passenger class on the ship, name, sex, age, number of siblings and spouses, number of parents and children, their ticket number, how much fare they paid, their cabin number, and where they embarked.

The classification problem for this dataset I chose was to predict whether or not a passenger survived the Titanic crash from their passenger class, sex, age, number of siblings and spouses, number of parents and children, and the price of their fare. This is an interesting problem to classify as it brings up the question of whether or not evacuators prioritized certain demographics of people over others, such as women and children, people with large families, or even higher class wealthy people.

Before using the various algorithms to analyze this dataset, I first had to process the data. When taking a look at the various features in this dataset, a few give a lot to be desired. Things such as ticket number and cabin number seem trivial in the prediction of survival.

Analysis

**Decision tree:** An algorithm that utilizes a tree like structure to best split the data into their label categories based on certain attributes. The initial root node is based on the attribute that best splits the data and then as you go further down the tree, every node you encounter is another attribute that splits the data even further, eventually having the leaf nodes as the classifications.

Titanic: For this dataset, I was worried about the accuracy of the decision tree due to the number of features I was predicting off of. Having too many features could lead to the algorithm generating overly complex trees, leading to overfitting. Overfitting this data would lead it to try and model the training data too closely to accurately predict the testing data. In order to combat this, I implemented pruning via a max depth hyperparameter in the decision tree classifier that would prevent the tree from growing past a certain depth node. In order to find the most optimal max depth of the decision tree, I used cross validation while testing a range of possible max depths, from 1 up to 100 nodes.