For code, please visit: https://github.com/lumalav/CAP5610/blob/master/HW2/HW2.ipynb

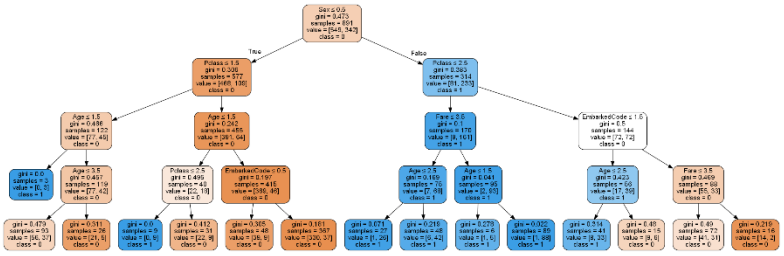
Task 1:

Q1: See section #Q1 of HW2.ipynb

Q2: The final Set of features used are: Sex, Age, Fare, Pclass, and Embarked

Part of the selection was due the results of HW1. During that homework, we saw that the strongest features that had a strong correlation with Survived were Pclass and Sex. Age also played an important role since a lot of younger people in lower classes died. Embarking port C had a strong survival rate and, also the higher paying passengers. I tried in this HW2 making some analysis in SibSp and Parch. However, since these values are aggregated into two columns it became too difficult to make sense of this data, so I ended dropping the features. I tried also grouping the tickets giving a discrete value to each ticket and found and interesting correlation with the Survived feature. But, at the end decided to drop it from the training since I got better results without it.

Q3: For code, See section #Q3 of HW2.ipynb. For, a better resolution of the image check DecisionTree.png



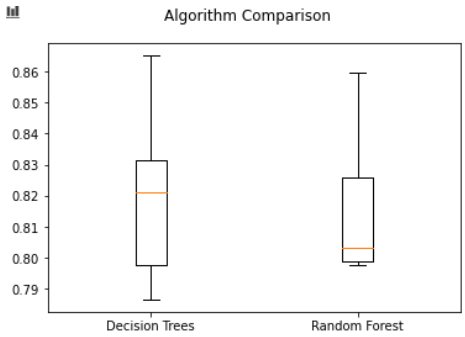
Q4: See section #Q3 and #Q4 of HW2.ipynb.

```
Decision Tree average five-fold cross validation accuracy: 0.8204255853367648
```

Q5: See section #Q5 of HW2.ipynb.

```
Random forest average five-fold cross validation accuracy: 0.817079907099366
```

Q6-Q7: Since I tried to optimize both algorithms by tweaking the max depth and the estimators, I've got similar results for both. I think decision trees are simpler since you can see the decisions that are being made and possibly write rules about it. It also seems 'faster' to generate and predict over a decision tree than the random forest. However, the random nature of a random forest might be a strong feature against overfitting. In a box chart we see greater difference between the min and max scores for the decision trees. At the end, I think both options are suitable for the problem. However, to make an educated decision, there is a lot more to learn in algorithm comparison analysis.



See section #Q7 of HW2.ipynb

Task 2:

a) 29%. The algorithm always chooses the biggest values of each node. So, if we count the smaller values (5+6+2+6+5+5) that will give us the error rate.

b) It would choose a "-" The path would be like A → B → E → "-". C = 1 is discarded since we need A = 1 to reach to C.

Task 3:

Q1: $E_{ori} = -\frac{6}{10}\, log_2\left(\frac{6}{10}\right) - \frac{4}{10}\, log_2\left(\frac{4}{10}\right) \approx 0.97$

Q2: $E_{A=T} = -\frac{4}{7}\, log_2\left(\frac{4}{7}\right) - \frac{3}{7}\, log_2\left(\frac{3}{7}\right) \approx 0.98$ $\quad E_{A=F} = -0\, log_2(0) - 1\, log_2(1) = 0$

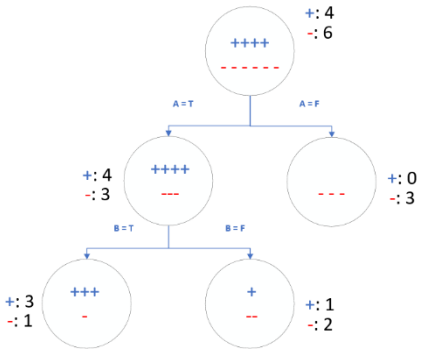
$IG_A = \left|0.97 - \left(\frac{7}{10}\, x\, 0.98 + \frac{3}{10} x\, 0\right)\right| \approx 0.284$

Q3: $E_{B=T} = -\frac{3}{4}\, log_2\left(\frac{3}{4}\right) - \frac{1}{4}\, log_2\left(\frac{1}{4}\right) \approx 0.81$ $\quad E_{B=F} = -\frac{1}{6}\, log_2\left(\frac{1}{6}\right) - \frac{5}{6}\, log_2\left(\frac{5}{6}\right) \approx 0.65$

$IG_B = \left|0.97 - \left(\frac{4}{10}\, x\, 0.81 + \frac{6}{10} x\, 0.65\right)\right| \approx 0.256$

Q4: It would choose attribute A since it gained more information from it.

Q5:



Task 4:

Q1: No. They are nonlinear since there is no equation to express dependency between variables.

Q2: Decision trees don't seem a good fit for regression problems. Also, adding new datapoints to a decision tree can lead to complete restructure of the tree and since it keeps adding new nodes, it leads to overfitting of the data which leads to wrong predictions, high variance and inaccuracy of the data. Decision trees are not good fit for large datasets and also are very susceptible to noise in the data.

Q3: No. Gini Index is better. Even though the range of the two functions is the same [0,0.5], misclassification is not that sensitive to measure impurity. For instance, if we have a decision tree where the splitting criteria is as pure as possible (0), the information gain based on misclassification error would be zero.