1.

b.

pclass I added this for two reasons. The first is that the data shows that there was a higher survival rate for people of a high class cabin. Second, logically, it would make more sense for people to be prioritized based upon cabin class since they are also more likely wealthy individuals that operate businesses and the like. Therefore, they were more likely to make it onto the ships

sex(gender for data purposes, but gender is not the same as sex and I prefer to not use the term gender whenever possible) I added this for two reasons. The first is that the data shows that there was a higher survival rate for women than men. Second, logically, women and children are prioritized in large crisis. This was especially true during those times so it only makes sense that this would have an impact.

age I added this for two reasons. The first is that the data shows that there was a higher survival rate for younger people. Second, logically, women and children are prioritized in large crisis. This was especially true during those times so it only makes sense that this would have an impact.

f.

The results of the trees show that the decision tree is more accurate. While the last three results for both trees are the same, the first two are higher for the decision tree by 0.025.

g.

My conclusion is that overall, there isn't that much difference between the results of the two trees. The results were very close or identical. I am sure there could be more pronounced differences with much larger functions, but as it stands, there wasn't a significant enough difference to say that one is superior over the other. I would lean though to a non-random decision tree because the program design seems more prone to error.

The rest of A is in the notebook in the repository

2.

a.(5+6+2+6+5+5)/(19+13+12+14+22+20)=0.29

the error rate is 29%

According to the slides, the error rate for a single leaf is equal to 1-max P(i|t)

which translates to 1- max(#,#) and in a binary case the lower number

to find the entire rate, I summed up all the misclassified objects (aka smaller numbers) on each leaf and then divided it by the total of all objects on all the nodes. Math is above

b. it would go into E's left leaf. A is 0, so it moves to B. B is 1, so it moves to E, E is 0, so it goes to the left.

3.

a. entropy before splitting means that they are still all in 1 class so the entropy is 0

b. -(4/7)*log(4/7) - (3/7)*log(3/7) = 0.297

- (3/3)*log(3/10) - (0/3)*log(0/3)= 0

0.297*(7/10)+0*(3/10) = 0.2079

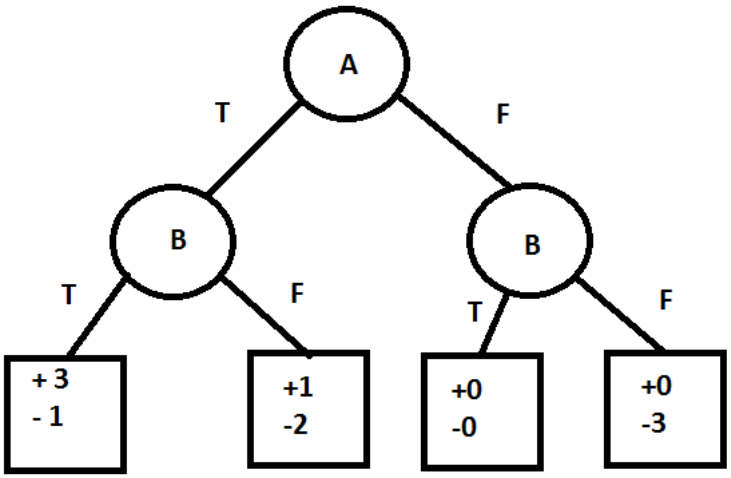
c. -(3/4)*log(3/4) - (1/4)*log(1/4) = 0.244

- (1/6)*log(1/6) - (5/6)*log(5/6) = 0.1957

* 4/10 + # * 6/10 = 0.215

d. Since we learn the most from minimal entropy, It will choose A as its attribute

e.



4.

a. Decision Trees are a linear classification because they categorize data based upon a set number of binary choices. They break down problems into smaller yes or no questions and using statistics based upon those decisions decide whether or not something will occur.

b. Decision Trees cannot classify anything that is not numerical. In order to overcome this people have to add numerical values to these categories so the computer can process them.

Decision Trees are reliant on the data order given to them. The paths can be completely changed like say when you subset and start with a different data point. So you can get a variety of trees from different partitions of data or slight alterations to data.

Decision trees can get big really quickly. They can become unmanageable from the sheer amount of decisions required. That is why some data is blocked into quarters so that the tree becomes small enough to be usable.

c. You would not want to create the tree based on misclassifications. Using that as a splitting criteria would lead to less interesting information since the end statistic of how many were in the minority of each group. It doesn't really pay attention to where or why. Gini notices smaller changes within the actual results and will help notice changes when we change break points. Gini will change as the graph does, but the error will always be the same no matter which order the break points are put in.

- 6-7 hours I think. Far more than I should have.
- To be honest, A was the hardest part because I was unaware of certain code lines. I then tried to basically reverse engineer the code. During this process I would find the function and then be done in 5 minutes with that problem. Otherwise it was simple to find things in the slides etc.
- I liked everything about this. I feel that the material made more sense by doing.

Repository Link: https://github.com/JoshuaFoldes2/CAP5610-Homework