

1. I implemented the ID3 decision tree algorithm using a separate Node class. From the fit function, I would create the root node. The root node would recursively create its child node in the create children function. Entropy is calculated in the calculateEntropy function, which return an array of entropy, one entropy for each column. Then the lowest entropy position would be marked, and the data would be split up to create more child nodes. And the child nodes would be stored in a childrenArray, which is a data attribute of each node class. Each node object as a most likely attribute, which is the most occurring classification from the labels.

	Cars Dataset	Voter Dataset
1	69.2	62.3
2	73.8	61.9
3	75.4	60.0
4	68.3	65.2
5	68.8	63.4
6	72.9	62.1
7	73.2	64.7
8	69.9	66.2
9	74.2	63.6
10	75.5	62.9
AVG	72.1	63.2

2. These accuracies are lower than expected because I think my base case of the recursively creating child node is too sensitive and will stop too soon thus leaving the tree not fully explored. Or I might be calculating the entropy incorrect, however going over it by hand on the lenses data set, I did not find it to be incorrect.
3. It seems like for the car dataset the most important attribute was the safety, it would split on that first and then the person carrying capacities. For the voter dataset, the most important attribute seem to be immigration.
4. I handled unknown attribute by analyzing the data and calculating the median of the data. Then I set all nan values in my dataset to that median. Thus, removing unknown attributes and making them into the most likely answer. I chose this approach because it was recommended to me by a fellow classmate who had near 100 percent accuracy on the car and the voter datasets.
5. As expected, the SKlearn decision tree model beat my model by about 20% more accurate on average, for both cars and voter. I ran the SKlearn decision tree on the magic.csv file. The hyperparameter that seems to affect overfitting the most was the min\_sample\_split and min\_sample\_leaf. Increase both of these parameters seems to reduce the chance of overfitting in general and increase accuracy.
6. I couldn't get the export\_graphviz function to work.