04 Prove: Decision Tree

1. Please provide a link to your classifier in your public GitHub repo.

https://github.com/johnmwood/CS450/tree/master/week04

1. Briefly describe your overall approach to the task and highlight the most difficult part of this assignment.

Etienne Beaulac and I worked together on a majority of the assignment. We approached the decision tree by building out every part that we understood, working on the tree and then debugging the finer points which weren't working. The hardest part was getting the tree to build properly, and then getting the logic to fit for leaves when the testing data had a feature which wasn't in the training data.

1. Briefly describe how you handled numeric data.

We used the cut pandas function on the iris dataset to discretize the dataset into categories.

1. Briefly describe your you handled missing data.

Missing data was pretty easy to handle in the voting dataset as we just changed "?" values to "meh" ("meh" meaning the person voting couldn't decide on a "y" or "n"). This was conveniently explained in the dataset description so we were able to handle it accordingly.

1. Describe your results for the Iris data set. (e.g., What was the size of the tree? How did your implementation compare to existing implementations? How did your decision tree compare to your kNN classifier)

The tree for the iris dataset is 4 layers deep. Our implementation has about an average accuracy of 95% while the sklearn ID3 Decision Tree algorithm usually got about 93%. We were definitely surprised to see that it was generally performing better. Our decision tree performed a lot better than our kNN Classifier in general.

1. Include a textual representation of the tree your algorithm produced for the iris dataset.

'petal length (cm)': {
'(0.994, 2.967]': 0,
'(2.967, 4.933]':
'petal width (cm)':
'(0.0976, 0.9]': 1,
'(0.9, 1.7]': 1,
'(1.7, 2.5]':
'sepal width (cm)':
'(1.998, 2.8]': 2,
'(2.8, 3.6]':
'sepal length (cm)':
'(4.296, 5.5]': 2,
'(5.5, 6.7]': 2,
'(6.7, 7.9]': 2}},

'(3.6, 4.4]': 2}}}},
'(4.933, 6.9]': 2}}

1. Describe your results for the Lenses data set. (e.g., What was the size of the tree? How did your implementation compare to existing implementations?)

We didn't run the algorithm on the lenses dataset.

1. Include a textual representation of the tree your algorithm produced for the Lenses dataset.

N/A

1. Describe your results for the Voting data set. (e.g., What was the size of the tree? How did your implementation compare to existing implementations?)

The voting tree was about 6 layers deep with an average accuracy of 94% while the sklearn implementation was getting about the same or slightly better results. It was really cool seeing the tree build itself out like that.

1. Include ***a portion of*** the representation of the tree your algorithm produced for the Voting dataset.

'physi': {'meh': 'democrat',
'n': 'adop-budg-res': {'meh': 'democrat',
'n': 'religion': {'meh': 'democrat',
'n': 'exports': {'meh': 'democrat',

'n': 'republican',
'y': 'democrat'}},

1. If applicable, please describe anything you did to go above and beyond and the results you saw.

Above the expected one dataset, we used both Iris and the voting dataset for the decision tree. We also used pandas efficiently to reduce the size of the algorithm used in the book.

1. Please select the category you feel best describes your assignment:

E - Shows creativity and excels above and beyond requirements

1. Provide a brief justification (1-2 sentences) for selecting that category.

I believe the algorithm we implemented was well executed overall. We are really proud of the work we did and really enjoyed putting together the code for the assignment. We not only worked a lot of hours on the assignment, but made sure to test and fix our algorithm until it was refined and working well.