Pratishta Prakash Rao

Big Data Analytics -HW_06

1. What structure did your final decision tree classifier have? What was the if-else tree you got?

Solution:

The final if - else structure of the tree obtained is as follows:

```
if BrkDnkIsTea == 1:
        if PizTopIsPepperoni == 1:
                return 0
        else:
        if DinDnkIsPapaya == 1:
                return 0
        else:
                return 1
else:
        return 0
```

2. What was the accuracy of your resulting classifier, on the training data?

Solution:

After running the resulting classifier on the training data, the accuracy obtained wAS 93.8%.

3. What else did you learn along the way here?

Solution:

There were a few issues that encountered while doing this assignment.

- I was incorrectly calculating the purity of the leaf node as a result of which I ended up getting so many more attributes even after the max depth was reached. The tree didn't have leaf nodes.
- After rectifying my error, while splitting the attribute into two different data frames I noticed that a lot of my data frames where I was splitting the rows where the best attribute ==1 were empty. It took me a while to realize that due to the one hot encoding, if suppose the best attribute was 'BrkDnkIsTea', all the values other 'BrkDnk****' would be 0 which resulted in data frames being empty since these attributes cannot have a value of 1. In order to handle this case, I created a function to "drop_insignificant_cols" to drop such columns whose bhattacharyya co-efficients are zero in order to avoid getting empty data frames.

This involved a lot of thinking interms of the code structure which did take a significant amount of my time in identifying the problem and coming up with the solution.

- The other area which took time was to create the "emit_decision_tree" function as I had no idea recursively creating multiple decision stumps and the indentation.

4. What can you conclude?

Solution:

The following are the things that I have taken from this assignment:

- Proper implementation of the decision tree : including the need for stopping criteria and using decision stumps.
- Importance of bhattacharyya co-efficients in determining the similarity between the nodes
- Finding edge cases and creating a test suite.
- String manipulations in order to do indenting and writing it to a file.