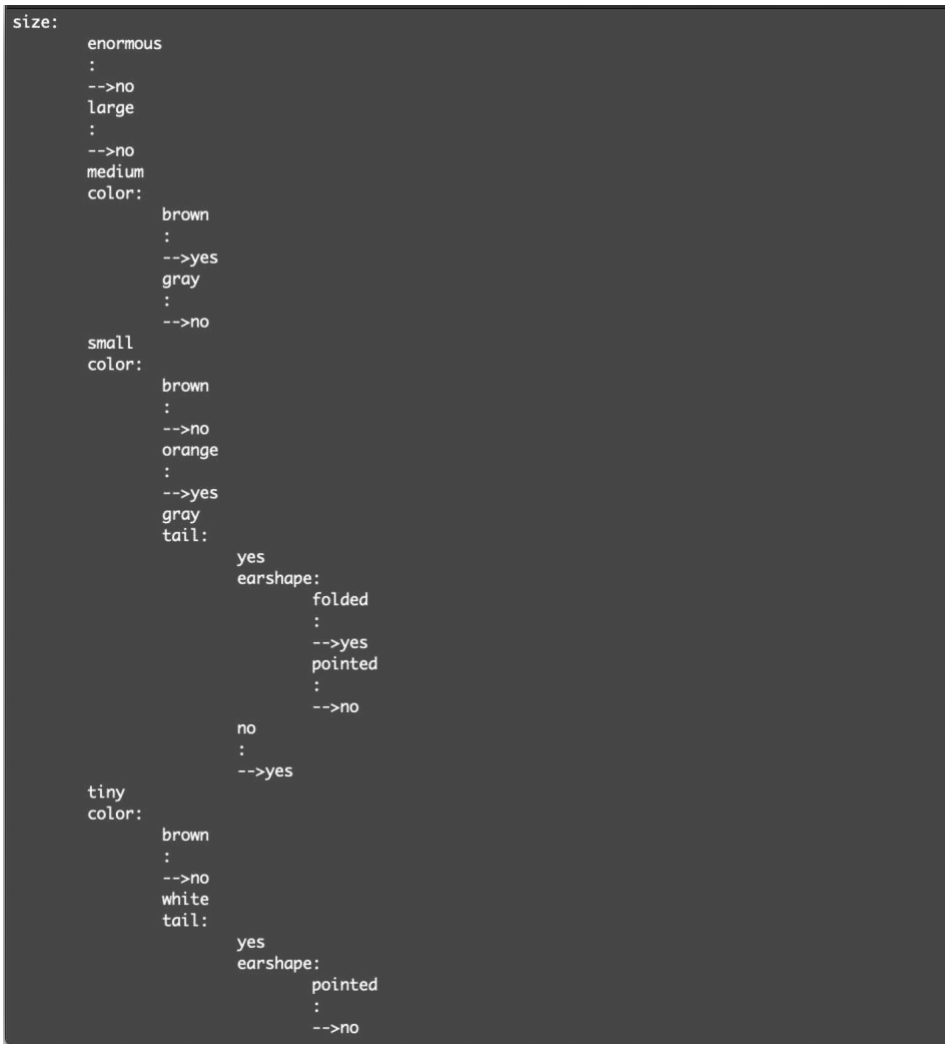
Muskan Uprety

Lab C Report

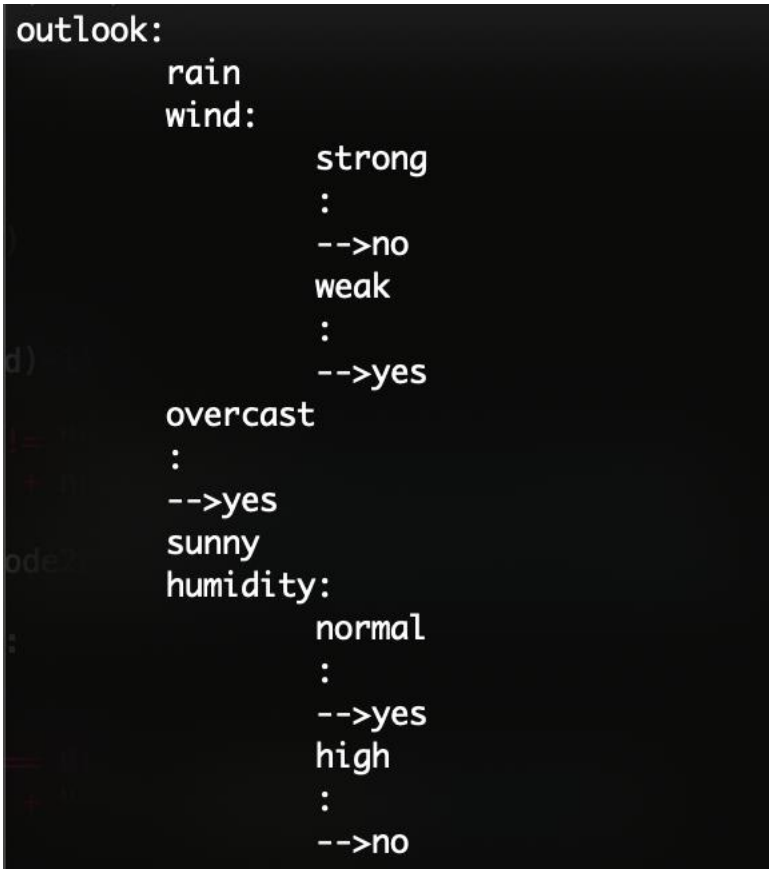
April 18, 2020

The objective of Lab C was to implement a decision tree that predicts binary classification for a set of given input variables. The program takes a file that is tab delimited and creates a 2D matrix, which the function uses to create a decision tree. The algorithm uses information gain from each input variable to decide which input to expand next by maximizing the gain. I tested the algorithm with the 3 files provided and the following diagrams represent the decision tree for each of them:

Pets.txt:

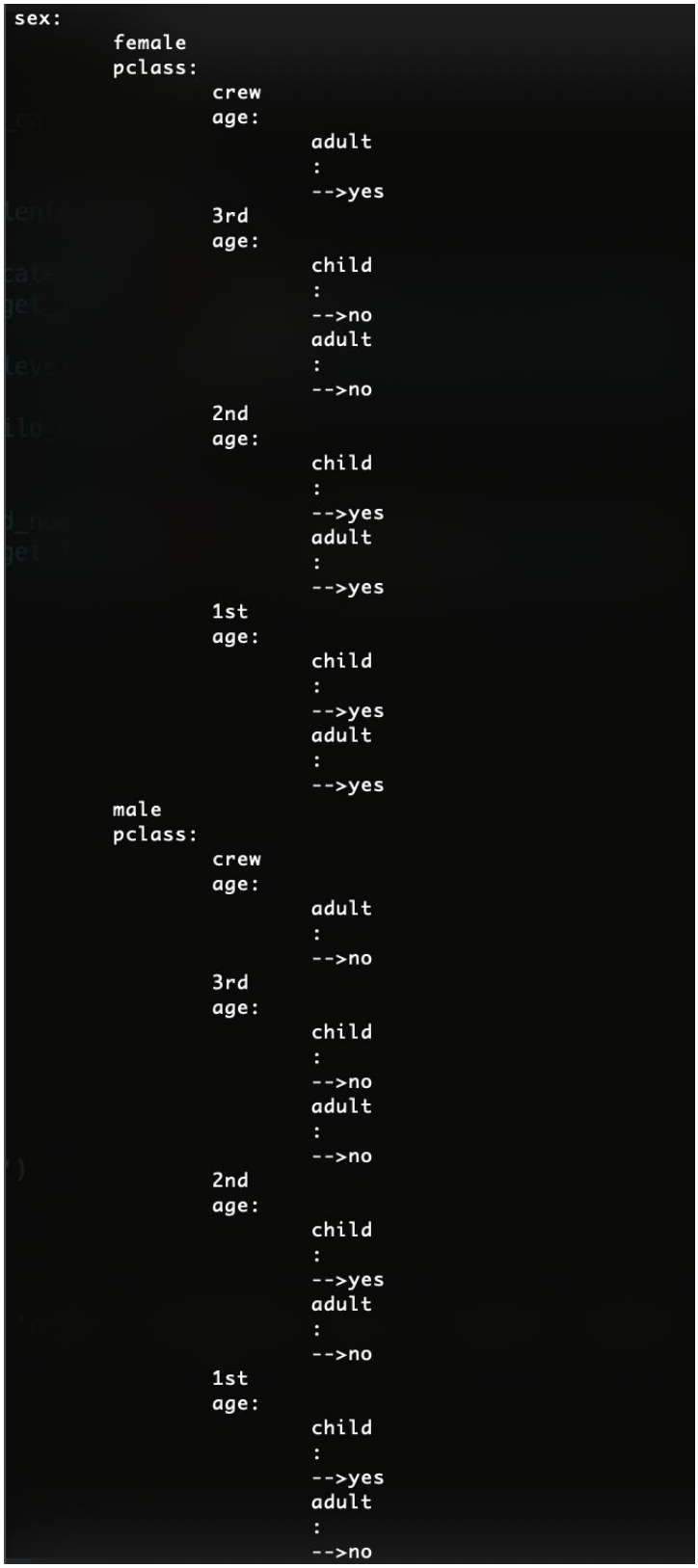


Tennis.txt:



Every tab we go inside shows a deeper level of the tree. For instance, in tennis.txt, calculating the gain for each column, the algorithm finds that outlook has the highest gain. So level 0 is "outlook" variable. How you would read the tree above would be : if outlook is rainy and wind is strong, the classification for "play tennis" is no, but if outlook is rainy and wind is weak, classification is yes; if outlook is overcast, classification is a yes without having to look at any other input variables and so on. The symbol '→" preceded final answer to avoid confusion on which the final classification is.

Another tree for titanic2.txt:



The tree for pets had 19 nodes, tennis had 8 nodes in the tree and titanic had 25 nodes in total.

My algorithm produced the following level of accuracy with 2 different types of test:

```
(base) Muskans-MacBook-Pro:AI_LabC muskanuprety$ python3 decision.py pets.txt
pets.txt
train set validation =  0.8666666666666667
test set validation =  0.3333333333333333
(base) Muskans-MacBook-Pro:AI_LabC muskanuprety$ python3 decision.py tennis.txt
tennis.txt
train set validation =  1.0
test set validation =  0.9285714285714286
(base) Muskans-MacBook-Pro:AI_LabC muskanuprety$ python3 decision.py titanic2.txt
titanic2.txt
train set validation =  0.7905497501135847
test set validation =  0.7900954111767379
(base) Muskans-MacBook-Pro:AI_LabC muskanuprety$
```