**Explain the advantages and disadvantages of writing a program on your own vs using a pre-created suite such as WEKA.**

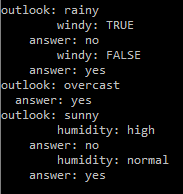
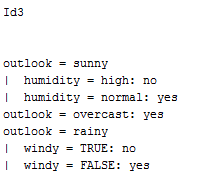
First of all, WEKA has a graphic interface that allows you to visualize the data and the tree in a more understandable way. The main disadvantage of writing a program first of all is the time it takes, it's longer to implement and if we wanted to visualize the tree in a graph we would have to develop a GUI. WEKA also offers a summary, a confusion matrix and a detailed accuracy by class. The main advantage of writing a program of our own and also the main disadvantage of the WEKA implementation is the flexibility this provides, while developing this we were able to print different stages of the tree and analyze them as we debugged the program.

**Explain what criteria you followed to choose the datasets for your tree and the WEKA tests.**

We tried all the test datasets given for the lab in order to test our program, that made us change the run.py file in order to ignore comments and accept upper and lower case letters as they were with slightly different formats. Next we used datasets that varied in size in order to test the performance of the program (it works fine for large data sets) and also the number of attributes in order to handle them.

**Include the graphics of the trees or part of the trees you generated in WEKA and your own program. Are they different, and if so, why?**

The only difference is the order in which the nodes are printed probably due to the tree run order. The next two images show the two trees being our program the one on the left and WEKA the one on the right.

**Based in what you have learned so far where would you use decision trees?**

We would use decision trees in scenarios like Francisco’s challenge that includes assigning students to different teams based on the size of the team and the amount of international students in them. A decision tree could help make a better distribution for this. Or for the Lego robots the robot could decide the next path based on different parameters it receives from the censors and comparing them to the training data set to take the best decision.

**Challenges during development**

We are not fans of working in a repository. That’s why even though we had been working on the program since last week we committed until today. Each commit shows which member(s) worked on which part. For example, Francisco created the skeletons of the files and the run file while Antonio and Angel developed the Node class and finally we all worked on the main event, the id3 implementation. As we are no experts in python, but we know our way around we used as reference for the structure a sample implementation that takes a train set and a data set in order to create the tree (<https://github.com/whatot/id3-py/blob/master/dtree.py>) this helped us to clarify a lot of our doubts on how to start implementing the algorithm, finally a classmate that already took the class helped us with the problem we had on calculating the gain and the entropy of the data set by recommending us to do it column by column instead of the data set all together.