Harsh Patel

CS 3600 – Introduction to AI

Project 4a

Question 6: (Performance)

The testDummySet1 had a tree size of three and an average classification rate of 1.0. Although testDummySet1 only had 20 examples in the training set (with an equal split of positive and negative examples), it was able to get a very high classification rate due to it being able to learn that only one attribute (attribute #5) was the deciding factor between the positive (if attribute #5 == 0) and negative examples (if attribute #5 == 1). Due to only one attribute completely splitting the examples into two distinct groups (completely reducing the entropy to zero), there was no further need to split and the tree size was able to remain relatively small. The testDummySet2 had a tree size of eleven and an average classification rate of 0.65. Unlike testDummySet1, testDummySet2 was not able to find a clear split between the positive and negative examples so the tree size was a bit bigger. Also, due to the small number of examples, it was not able to tease apart the differences between some of the examples and therefore had some incorrect classifications, explaining the lower classification rate. The Connect4 dataSet had a tree size of 41521 and an average classification rate of 0.7579. The extremely large tree size comes from the large number of distinct examples (which is reasonable as there lots of winning game states in a Connect 4 game) and the usage of each spot on the game board as an attribute (instead of using patterns as attributes). The average classification is also reasonably high once again due to the vast number of examples (as lots of winning state space is being covered aka example being tested was in training set). The Cars dataSet had a tree size of 408 and an average classification rate of 0.943 over all runs. The tree size of this dataset was reasonably small compared to the number of examples provided (1728 examples) which bodes well for the overall learning phase of the algorithm. There were enough examples to differentiate the differences between the attributes and the use of descriptive attributes (instead of possible piece positions like that in the Connect4 dataset) allowed better grouping of examples. As a result, a higher classification rate was achieved.

Question 7: (Applications)

The Cars dataset (or similar dataset with maybe a few more attributes) and the resulting decision tree could be used in conjunction with a user-facing GUI as part of a car dealership's "new car" selection process. The dataset could have examples with attributes such as price range, number of seats, safety rating, MPG, trunk size, number of cylinders, etc. The customer can then interact with the GUI and select attributes they desire and the decision tree would display similarly grouped cars for the user to explore further. The dealership can also use the decision tree to group new cars that the automobile company produces with the cars it already has in the dealership. The Connect4 dataset can be used in conjunction with a search algorithm like A* when creating a better Connect4 bot. Basically, we can add another term to the A* heuristic

calculation that takes into account the average win percentage of all examples that are most similar to the possible game states the agent has to pick from as it traverses the search tree. In other words, as the agent is traversing the search tree, we can get the child nodes of the current node we are at, use the decision tree to classify those game states, and then take classification (projected win/loss) into account along with the heuristics to make an “informed” decision.