Decision Tree

Machine Learning

Aiden Koknat

Data Representation:

The points were stored in a multidimensional array, with each initial index containing a tuple that contains the original data and the discretized version. The last index in each data point is the class label (0 or 1).

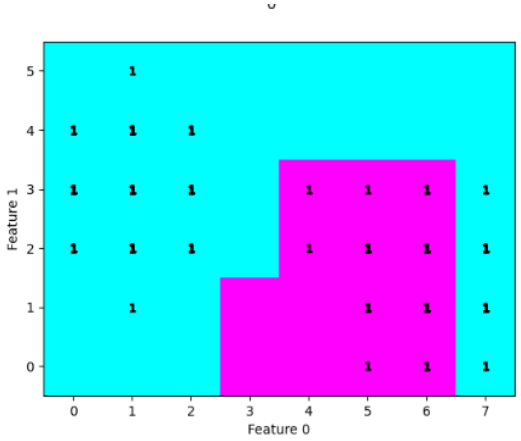
Working with Synthetic Data:
Many of my initial issues came from deciding how to properly organize my code. I began with doing a function-less coding that would read, discretize, and split data of the first synthetic data. This is because I started without a thorough understanding of the decision tree intricacies. However, once I brushed up and learned how the tree worked, I was able to create functions for each part of the data sorting process and use it for the rest of the synthetic datasets.

Working with Pokemon Data:
Since my functions worked universally, I merely had to alter the pokemon code to be in a similar format as the synthetic data sets. This involved removing the names on the stats, and changing the True rows on the pokemonLegendary file to 1 and False rows to 0, which I then appended to the pokemon stat file.

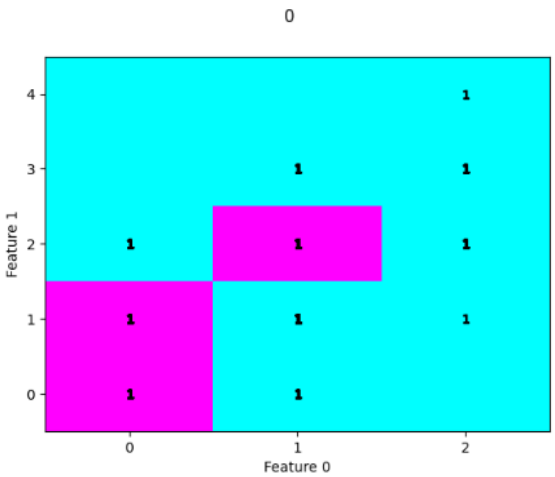
Predicting Data:
In order to predict what potential points' class value would be, I iterated through the tree that the point's reference data was a part of, and then checked to see if the prediction that was stored inside the tree matched what the class value of the actual point was. Since my data was represented in a fashion that kept the raw and discretized data next to each other, it was easy to quickly reference both values when needed.
If the classValue of the data point matched the predicted value of the tree, it would add a point to the accuracy. After iterating through all the points, it would divide how many points were successful by the total amount of points, before displaying the accuracy.
I did not have enough time to bug fix the prediction/accuracy of the program, but the essence of the code is there. The accuracy of the first synthetic dataset I got was 68.5%, for example. With more time, I would've workshopped the discretization to increase the accuracy.

Visualizing Data

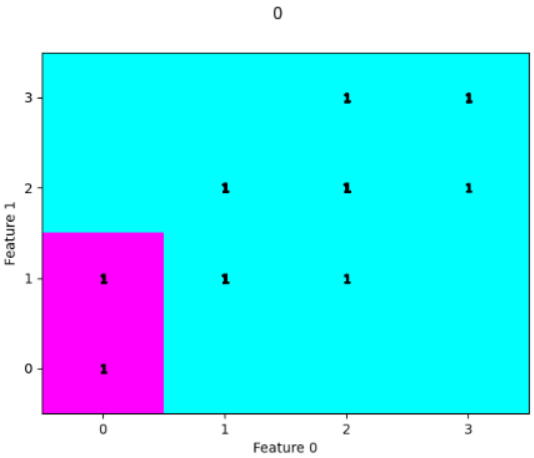
Synthetic 1:



Synthetic 2:



Synthetic 3:



Synthetic 4: