CMPE 481 Data Analysis and Visualization
Assignment 2 - Decision Trees
Zuhal Didem Aytaç – 2018400045

Step 0. Decision Tree Algorithm

The code processes as follows:

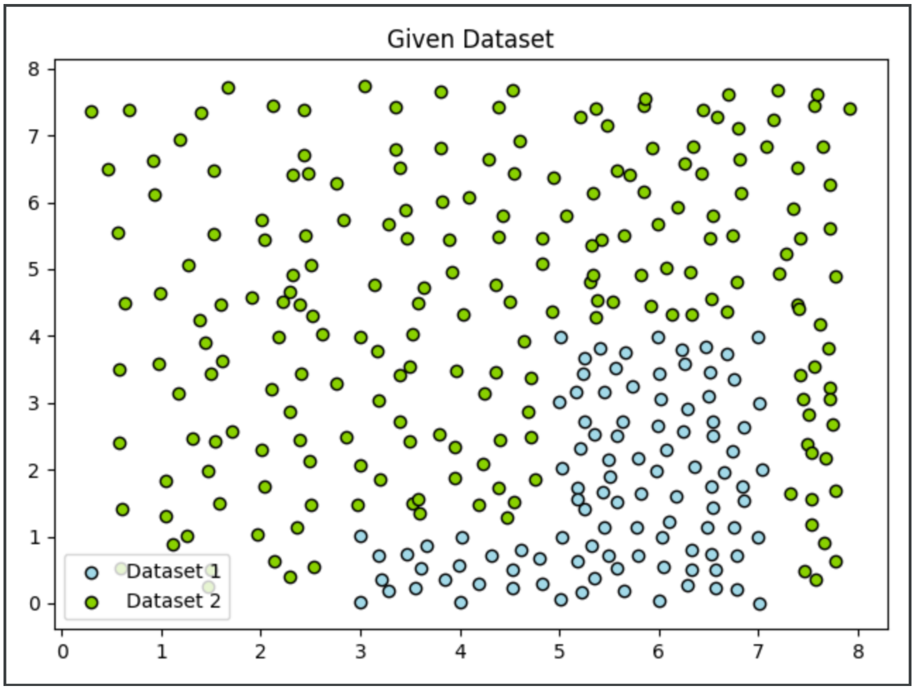
- It reads the given txt file and plots.
- For each row, it creates a Data object (with attributes x,y and group)
- In given data, it stores the minimum and maximum values for x and y.
- For each axis, it tries to split the range to 100.
- For each split trial, it calculates the information gain.
- For each axis, it finds and returns the best split point.
- It compares the splits and selects the one with higher information gain as the initial split.
- It reports the following values for initial split: split axis, split point, entropies of the resulting groups, weighted average entropy.
- After initial step is determined and reported, it continues with the second split.
- It decides which one of the groups to split further.
- It selects the group with higher entropy.
- It applies the same processes for the second split.
- It tries to split the group with higher entropy with respect to both x and y axis.
- It selects the split with higher entropy as the second split.
- It reports the following values for second split: split axis, split point, entropies of the resulting groups, weighted average entropy.
- It plots the results.
  - It plots the dataset
  - It draws the initial split with a thick line.
  - It draws the second split with a thin line.
  - It shows the boundary points in the legend.
- It then applies the Decision Tree Classifier of scikit-learn activity to the dataset.
- It uses the 'gini' criterion for the model.
- It extracts the split points and axis from the decision tree text.
- It plots the same results for the scikit learn splits.

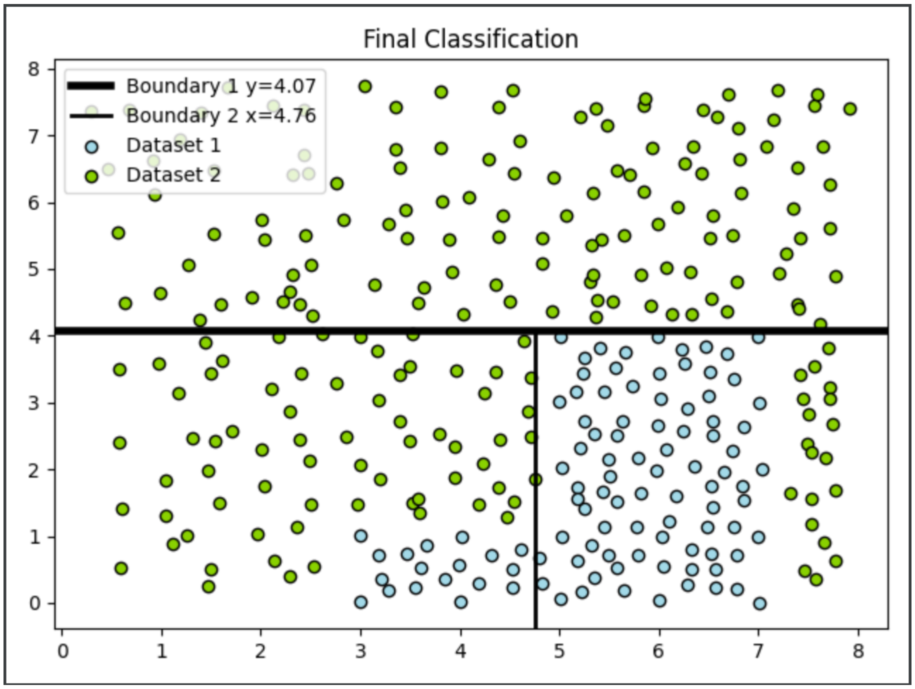
Running the Code (python version: Python 3.10.0)

```
pip3 install -r requirements.txt
python3 assignment2.py
```

Step 1. Decision Boundaries

The given dataset and the code's output are shown below.

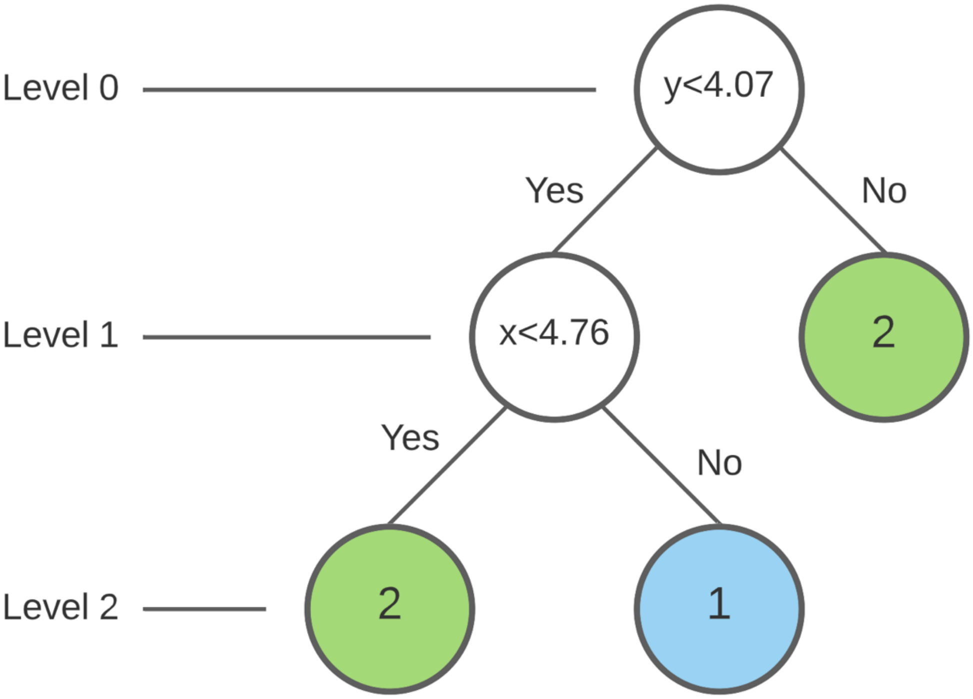




The code outputs the decisions as follows:

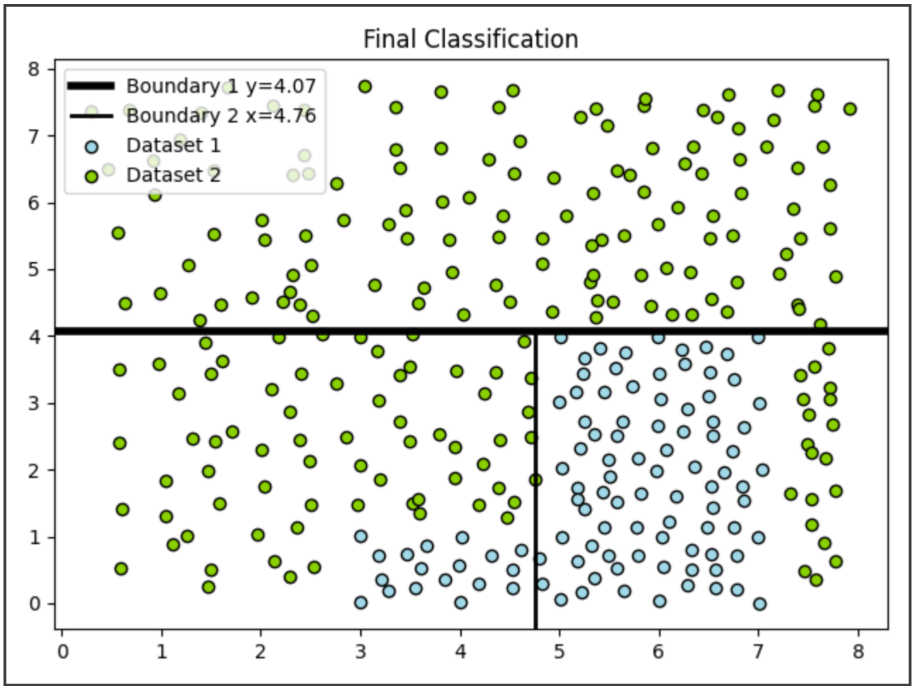
```
First split is made with respect to axis y at point 4.07020
Entropy bottom: 0.99545
Entropy top: 0.00000
Weighted average entropy: 0.60301
Second split is made with respect to axis x at point 4.76246
Entropy left: 0.74015
Entropy right: 0.68975
Weighted average entropy: 0.71268
```

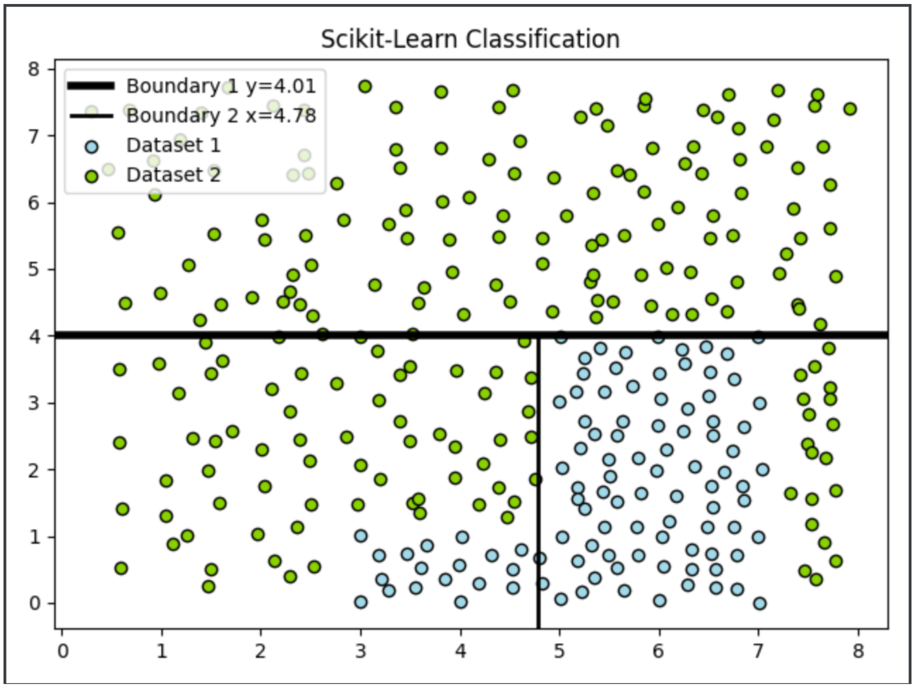
Step 2. Plot of the Decision Tree



Step 3. Comparison with Scikit-Learn

The code's output and scikit learn's output are given below to compare. As seen, the classification is similar with small differences in boundary points. The difference is at %0.01 level.

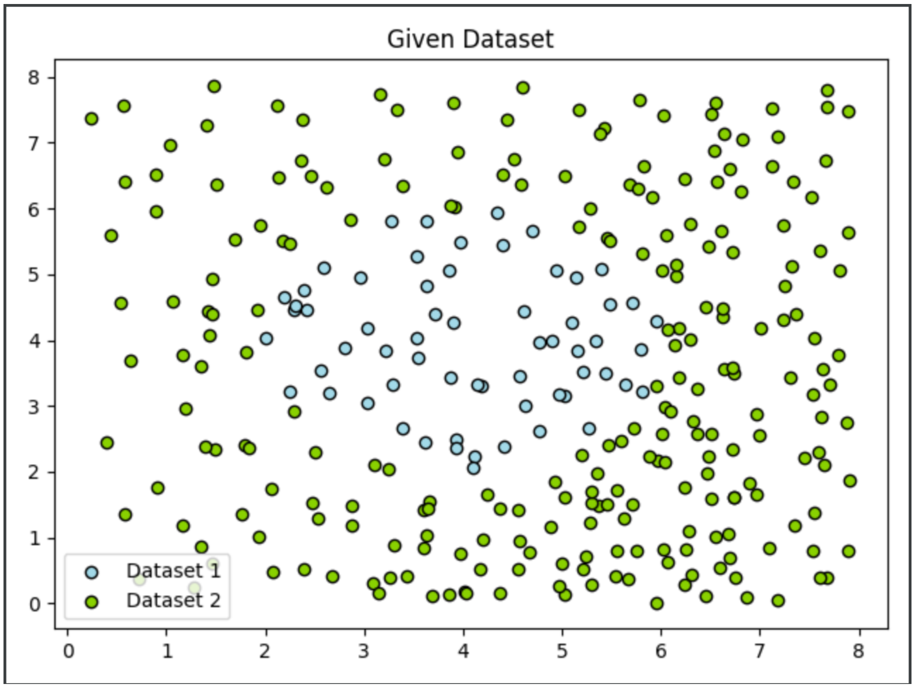


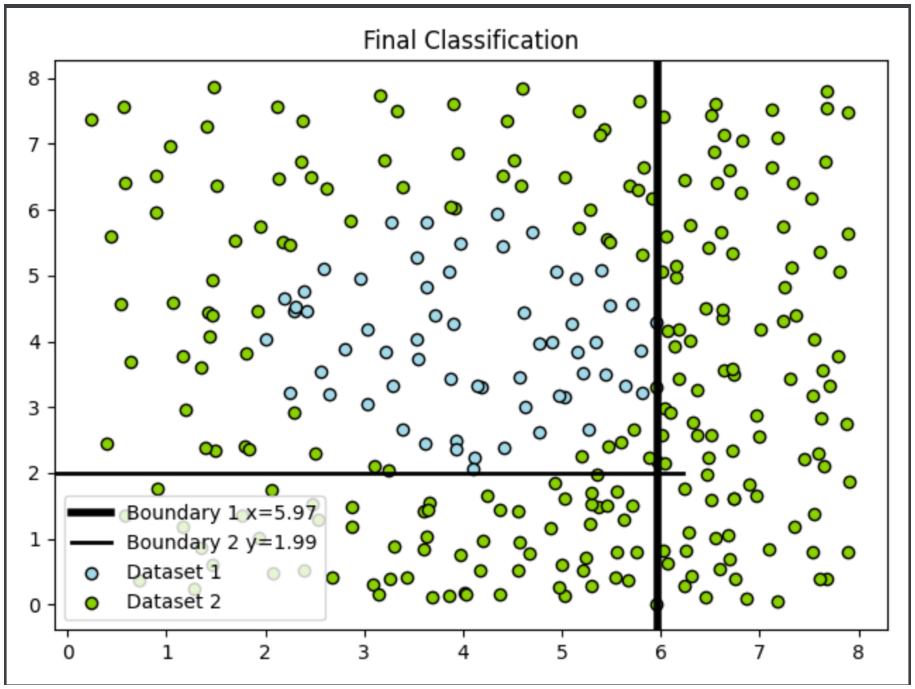


Step 4. My Own Dataset

Step 4.1. Decision Boundaries

I wanted to try a dataset where one group is centered on the middle. The outputs are given below.

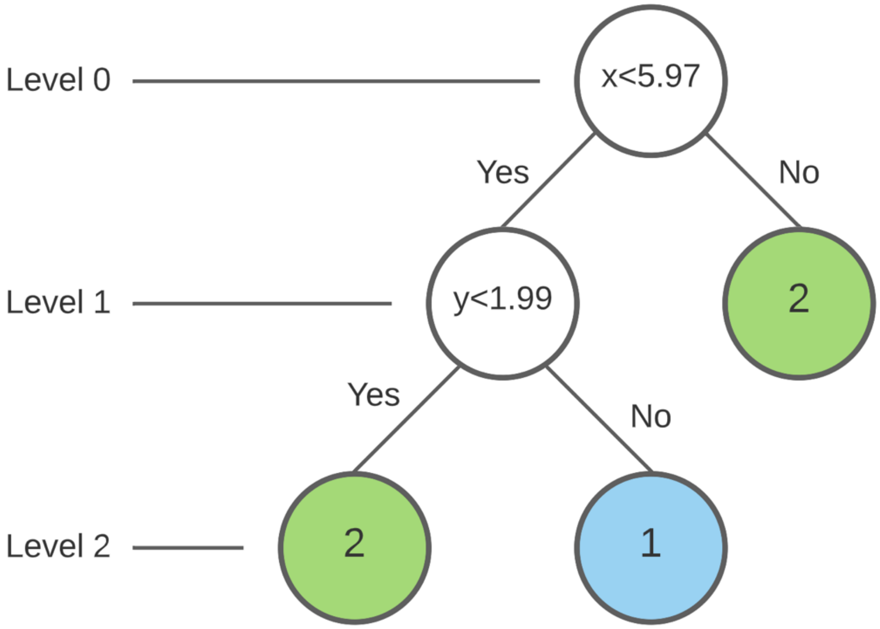




The code outputs the decisions as follows:

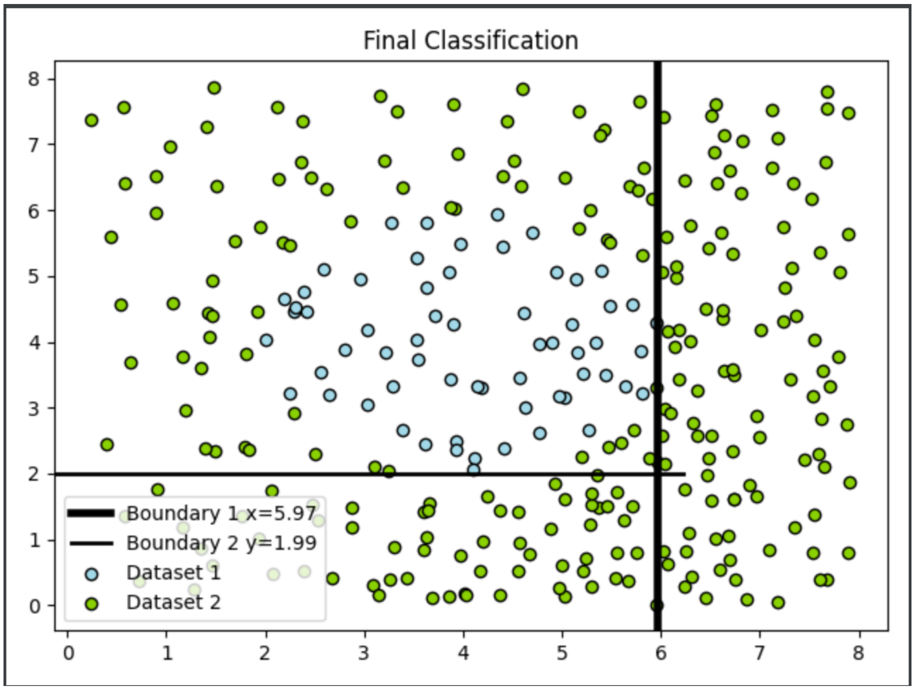
```
First split is made with respect to axis x at point 5.97249
Entropy left: 0.88960
Entropy right: 0.00000
Weighted average entropy: 0.59701
Second split is made with respect to axis y at point 1.99407
Entropy bottom: 0.00000
Entropy top: 0.99256
Weighted average entropy: 0.67809
```

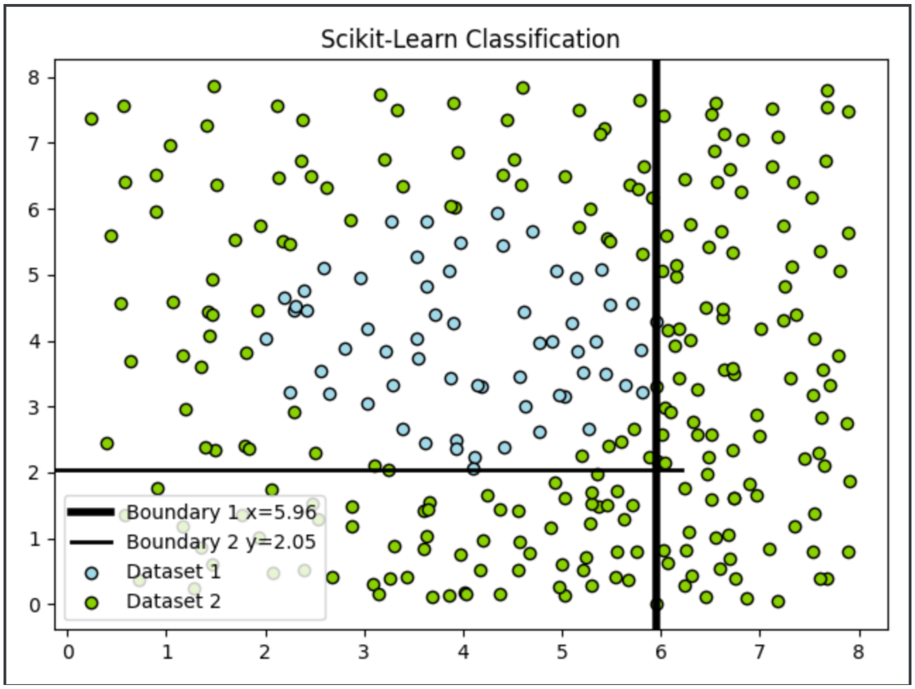
Step 4.2. Plot of the Decision Tree



Step 4.3. Comparison with Scikit-Learn

The code’s output and scikit learn’s output are given below to compare. As seen, the classification is similar with small differences in boundary points.


Final Classification
Boundary 1 x=5.97
Boundary 2 y=1.99
Dataset 1
Dataset 2


Scikit-Learn Classification
Boundary 1 x=5.96
Boundary 2 y=2.05
Dataset 1
Dataset 2