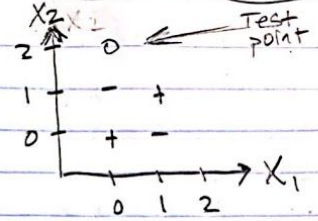
Atanas Delevski
ECE 407 HW6
4/9/2020

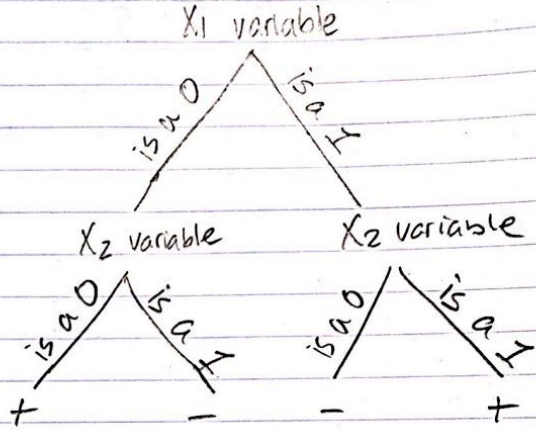
Question 1: (I did on paper)

ECE 407 HW #6 ATANAS DELEVSKI 6735411753 4/8/2020

Q1)

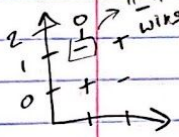


a) Decision Tree



b) We cannot classify $\begin{bmatrix}0\\2\end{bmatrix}$ using the decision tree, as there is no branch associated with picking a "2".

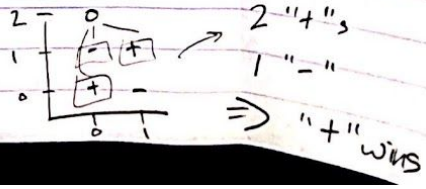
c) If we used 1-NN algorithm, we look at just the closest neighbor to the test point. In this case it is a "−"



↳ The point $\begin{bmatrix}0\\2\end{bmatrix}$ would be a "−"

d) If we used the 3-NN algorithm, we would look at the 3 closest points and pick which class has the most votes.

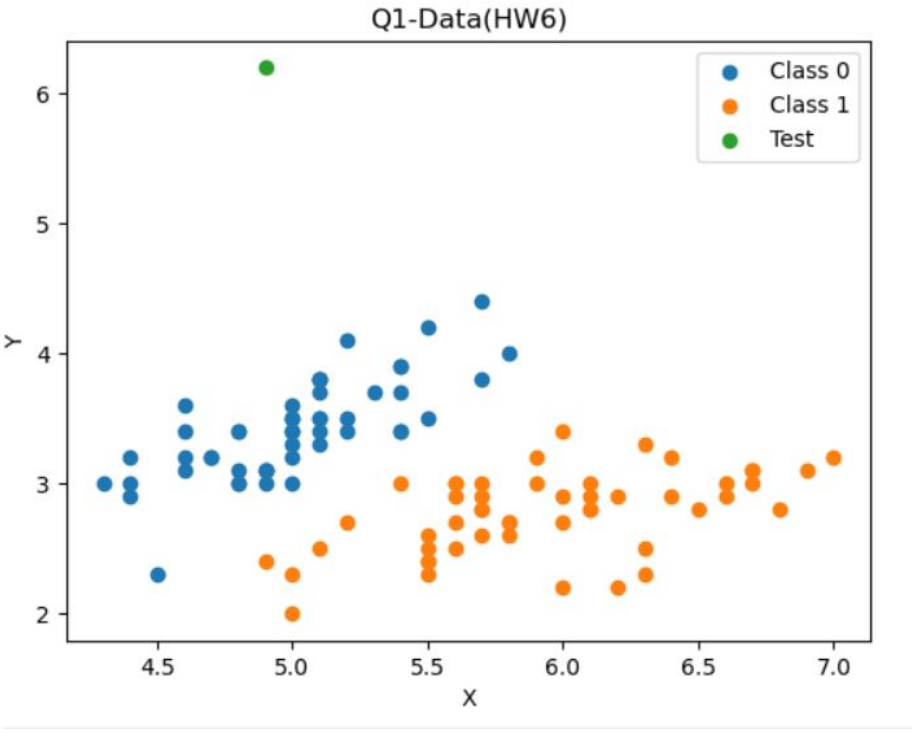
↳ $\begin{bmatrix}0\\2\end{bmatrix}$ would be a "+"



Question 2:

For this question, I used Python.

After opening the excel file in Google Drive to inspect the data, I realized I would need the ***matplotlib*** library to plot and visualize my data. I also realized I would need the ***Pandas*** library to work with the data.



After plotting the data, (and consulting Piazza), I was able to come to the conclusion that the vector [4.9, 6.2] would be classified as Class 0 whether it was 1-NN, or 3-NN, or 5-NN, or even 30-NN as it was so far above Class 0 and away from Class 1, that almost the entire set of Class 0 is closer to the test point than any of Class 1.

Therefore:

Part A) Class 0
Part B) Class 0

My Code is on the following Page.

My Code:

```
import pandas as pd
import matplotlib.pyplot as plt

data = pd.read_excel('Q1-Data(HW6)(1).xlsx', index_col='Id')
df = pd.DataFrame(data)

ds1 = df[:50] # Class 0
ds1x = ds1['x']
ds1y = ds1['y']

ds2 = df[50:] # Class 1
ds2x = ds2['x']
ds2y = ds2['y']

test_x = 4.9
test_y = 6.2

plt.scatter(ds1x, ds1y, label='Class 0')
plt.scatter(ds2x, ds2y, label='Class 1')
plt.scatter(test_x, test_y, label='Test')
plt.title('Q1-Data(HW6)')
plt.xlabel('X')
plt.ylabel('Y')
plt.legend()
plt.show()
```