Machine Learning
Fall 2022
HW2

Due: Sept. 13 (Tue), 11:59pm via Blackboard

Exploring K nearest neighbor (KNN) for classification (advanced): The following table shows the results of 100 students' scores in entrance exams 1 and 2, and their status of university admission. You can see the entire dataset in the file '**exam1and2.txt**'.

0: not admitted, 1: admitted

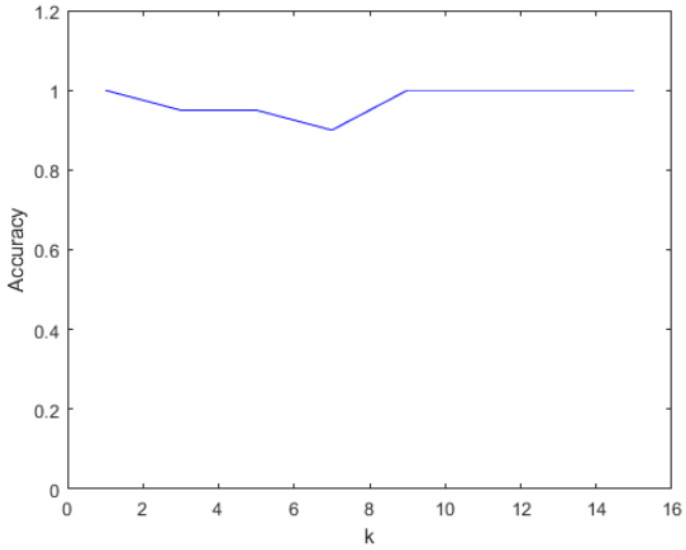
Student No.	Exam 1 (x)	Exam 2 (y)	Status (c)
1	34.62	78.02	0
2	30.29	43.89	0
3	35.85	72.9	0
4	60.18	86.31	1
5	79.03	75.34	1
.	.	.	.
.	.	.	.
.	.	.	.
96	83.49	48.38	1
97	42.26	87.10	1
98	99.32	68.78	1
99	55.34	64.93	1
100	74.78	89.53	1

Use the Matlab to solve for the following tasks:

1. Please divide the entire dataset into two subsets (training and test sets) in a 7:3 ratio, which means 70% of the dataset is the training set (70 samples) and the remaining 30% is the test set (30 samples). Based on the training set, you will classify 30 students in the test set to get their admission status. With K =3, please compare prediction results of the test set with actual results to calculate the classification accuracy on the test set.

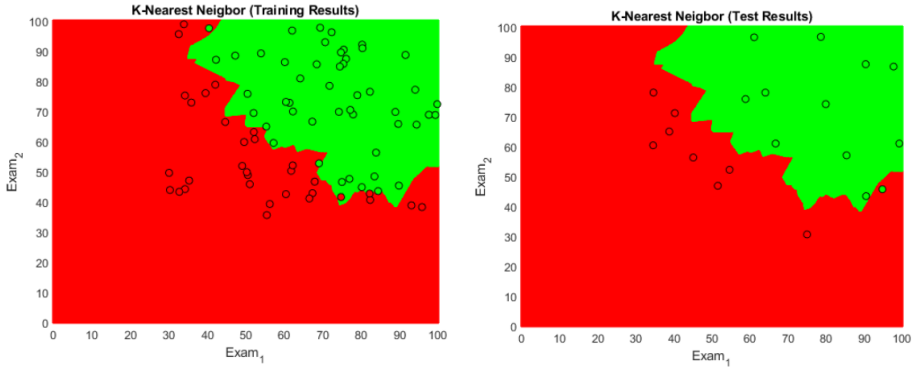
$$\text{Classification accuracy} = \frac{number\ of\ students\ in\ the\ test\ set\ who\ are\ classified\ into\ their\ actual\ \ status}{number\ of\ students\ in\ the\ test\ set}$$

2. Plot the relationship between K and classification accuracy of the test set. K ranges from 1 to 15 with step of 2.



Example figure

3. (**This problem is for graduate students only**) Draw the decision boundary with K = 3. Include the decision boundary and training set points in a figure, and, in another figure, include the decision boundary and test set points. (The form/format is similar to the example figures below. red circle: not admitted; green circle: admitted).



Example figure

Hints:

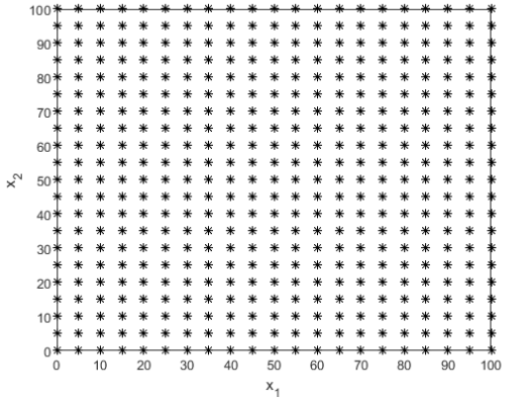
1. Use the built-in function **'load'** to load the dataset into the matlab.
2. Use the built-in function **'cvpartition'** to split dataset into training and test sets. Use the built-in function **'training'** and **'test'** to obtain the indexes of training and test set.
   For example,

   cv = cvpartition (100, 'holdout',0.2)
   data_train = data(training(cv))
   data_test = data(test(cv))
   Then you can training set 'data_train' and test set 'data_test'.

For the convenience, you can separate the score and the status.

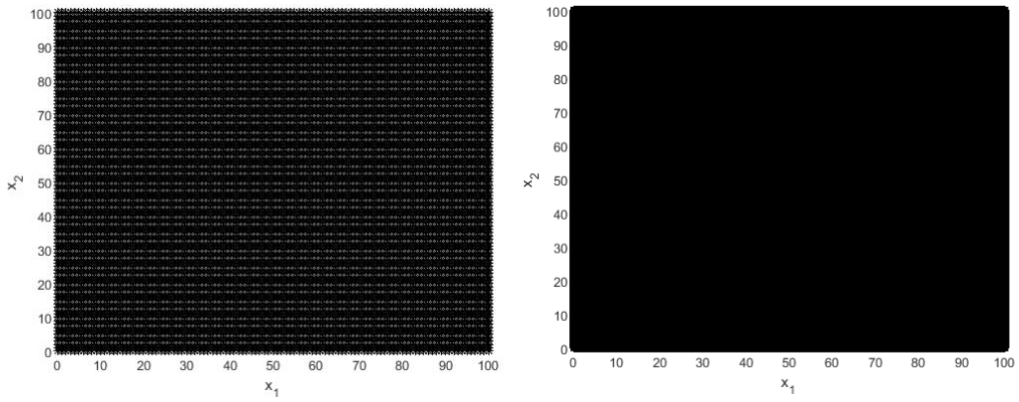
```
X1_train= data_train(:,1)
X2_train= data_train(:,2)
Y_train= data_train(:,3)
X1_test= data_test (:,1)
X2_test = data_test (:,2)
Y_test = data_test (:,3)
```

3. Use the built-in function **'meshgrid'** ,**'gscatter'**, and other useful functions to draw the decision boundary. The idea of drawing the decision boundary is to generate a square dot matrix. For example, if we generate x_1 and x_2 with step value of 5 (x_1=0:5:100, x_2=0:5:100), we get the matrix below.

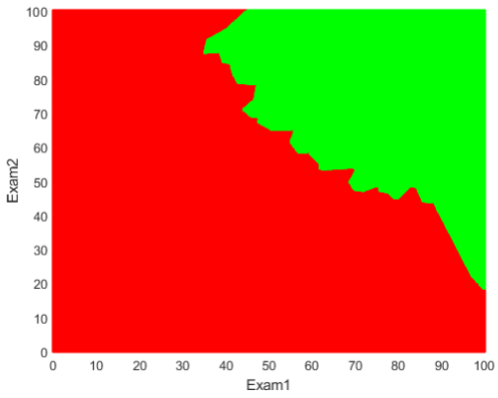


Hence if we generate the matrix with step value of 1 or less, what will happen?

step 1 step 0.1



We will achieve denser matrixes if a small step is used. Next, with the training set, we use KNN to predict every point in the dot matrix . If the prediction of one point is 1, we record 1 in the prediction vector, otherwise we record 0. Then we use the **'gscatter'** function to draw the decision boundary. You can plot the training set and test set on this figure.



4. When it comes to calculating the accuracy, you may use the operator '==’.

```
>> a=[1 2 3 4 2]

a =

     1     2     3     4     2

>> b=[1 5 3 4 9]

b =

     1     5     3     4     9

>> c=(a==b)

c =

  1×5 logical array

   1   0   1   1   0
```

5. The programming method is not unique, and you can try any approach to achieve the objectives.

6. This website https://www.mathworks.com/ is used to search matlab built-in functions. You can type the name of the built-in function in the searching column and learn how to use the built-in function.