ENGR-421 HOMEWORK-8

Name-Surname: Barış KAPLAN
KU ID Number: 69054
KU Email Address: bkaplan18@ku.edu.tr

Initially, I have imported the necessary libraries. The libraries I imported are as follows:

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.spatial as spa
import math
```

After that, as we did in the previous labs and as I did in the previous homeworks; I have read the data by using the np.genfromtxt function of the numpy library.

Then, I have calculated the euclidean distances between the data points. While calculating the euclidean distances between the data points, I have used the formula in Figure 1.

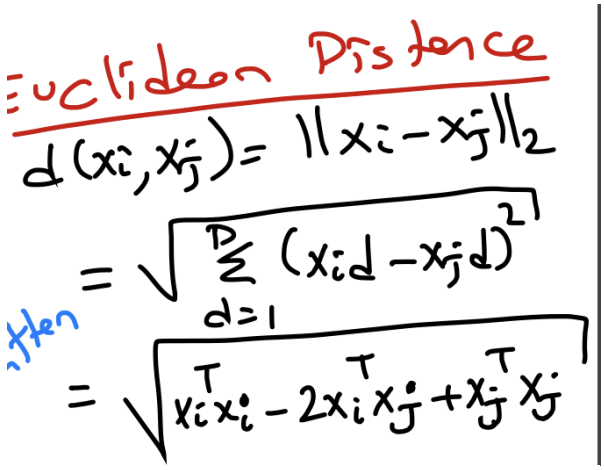

Figure 1: The formula I used to calculate the euclidean distances between the data points (from Lecture 23-Clustering)

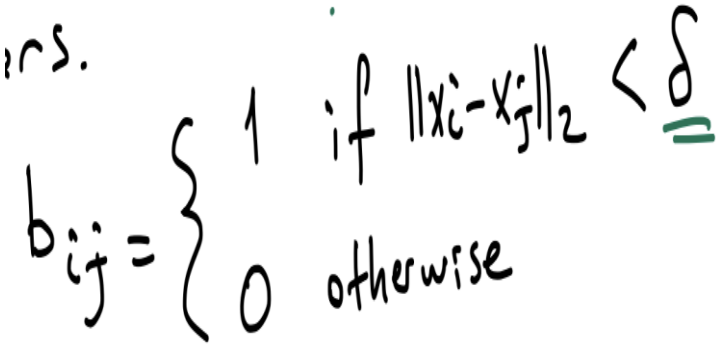

Figure 2: The rule I have used to construct the B matrix (from Lecture-23 Clustering)

After that, by using the rules in Figure 2 & Figure 3 ; I have constructed the B matrix. If the distance between the data points is smaller than the threshold value, I have assigned 1 to the entries of B matrix. Otherwise, I have assigned 0 to the entries of the B matrix. For the case where i is equal to j, I have assigned 0 to the Bij values for every i (assigning 0 to diagonals). Then, I have plotted the connectivity matrix by checking whether an entry of B matrix is equal to 1 (indicates connection between data points). You can see the connectivity matrix I obtained in Figure 5.

$$b_{ii} = 0 \quad \forall i$$

Figure 3: The second rule I used while constructing B matrix

Next, by using the rules in Figure 4, I constructed the D matrix. For constructing the D matrix, I have summed up the number of 1's in each row and assigned the resulting sum for each row to the diagonal entries in each row. I have assigned 0 to all other entries.

$$d_{ii} = \sum_{j \neq i} b_{ij} \quad \forall i$$

↳ # of neighbors of datapoint i

$$d_{ij} = 0 \quad \forall (i, j \neq i)$$

Figure 4: The rules I used while constructing the D matrix

After that, by using the formula in figure 7; I have normalized the L matrix that I have calculated. Subsequently, I have obtained the eigenvalues and eigenvectors of the Lsymmetric matrix by using the np.linalg.eig function of the numpy library. Then, I have sorted the eigenvalues according to their indices by using np.argsort function of the numpy library. Moreover, I have obtained the 2nd, 3rd, 4th, 5th, and 6th eigenvalues. I have constructed the Z matrix by firstly making an array (by using np.array function of the numpy library of Python) with 2nd, 3rd, 4th, 5th, and 6th eigenvalues; and then by taking the transpose of that array (by using ".T" abbreviation) . Next, I have obtained initial centroids by vertically stacking (with np.vstack function of the numpy library) of the desired rows of the Z matrix. Then, I have implemented the K-means clustering algorithm, and executed the K-means clustering algorithm on the Z matrix. While implementing the K-means clustering algorithm, I have benefitted from "Lab11-Clustering". While initializing the centroids in the update_cens function, I have assigned the centroids to the initial centroids which are obtained from the desired rows of the Z matrix. The logic of plotting the clustering results is same with "Lab11-Clustering". While plotting the clustering results, I have used the plt.plot, plt.xlabel, plt.ylabel, plt.title functions of the Python's **matplotlib.pyplot** library. In Figure 6, you can see the plot which displays the clustering results.

CONNECTIVITY MATRIX :

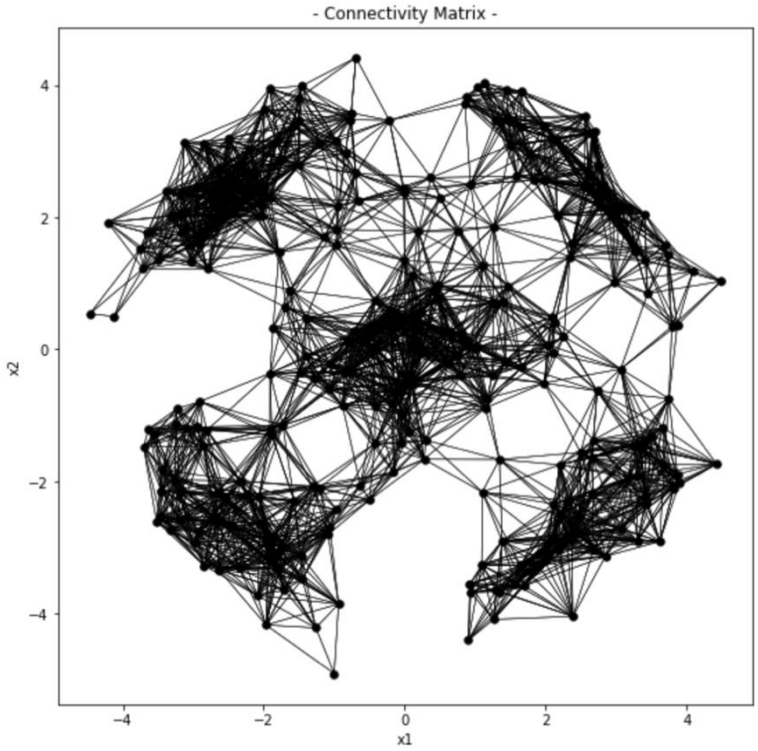

Figure 5: The connectivity matrix that I found (It is same with the expected resullt given in pdf)

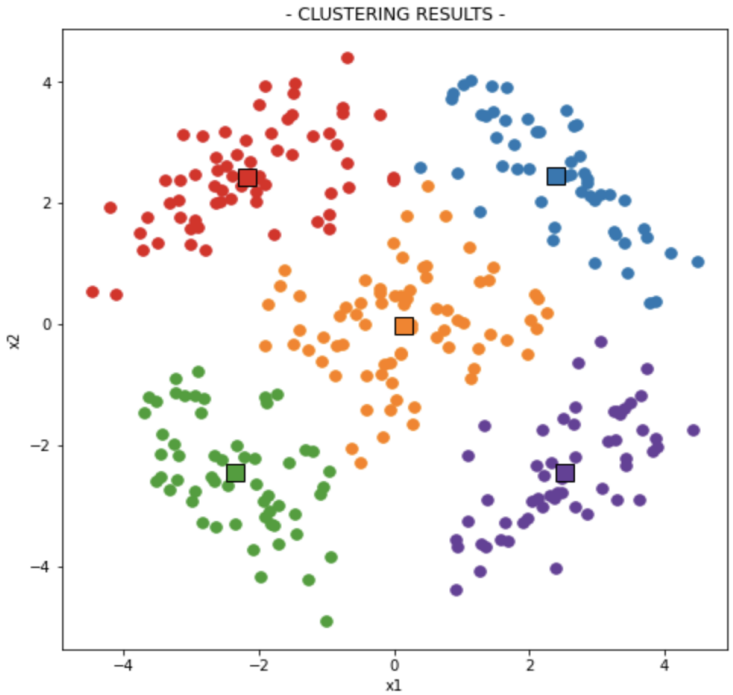

Figure 6: The plot which displays the clustering results (It is the same with the expected result given in pdf)

Please see the 1st and 2nd slides (the Spectral Clustering part) of the "Lecture-23-Clustering" to see the formulas I used in this homework.

$$\mathbf{L}_{symmetric} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{B}\mathbf{D}^{-1/2}$$

Figure 7: The formula I used to normalize the L matrix