ENGR 421 / DASC 521: Introduction to Machine Learning
Homework 09: Spectral Clustering

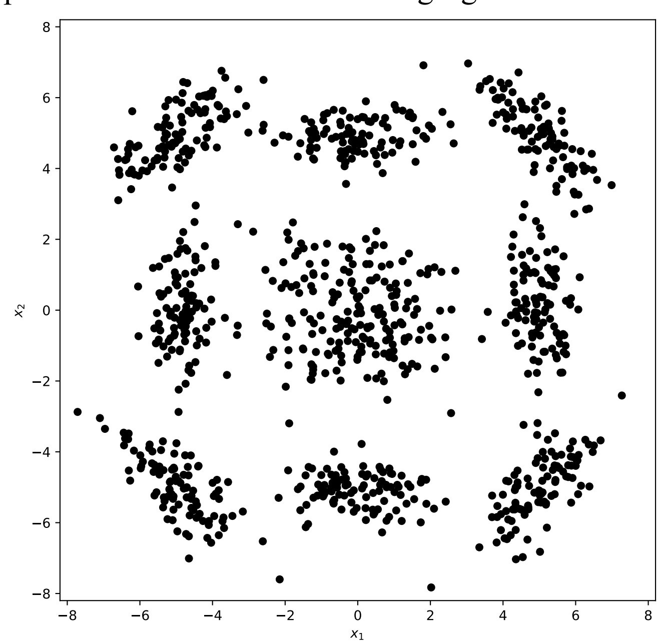
Deadline: May 28, 2022, 11:59 PM

In this homework, you will implement a spectral clustering algorithm in Python. Here are the steps you need to follow:

1. You are given a two-dimensional data set in the file named `hw09_data_set.csv`, which contains 1000 data points generated randomly from nine bivariate Gaussian densities with the following parameters.

$$\mu_1 = \begin{bmatrix} +5.0 \\ +5.0 \end{bmatrix}, \quad \Sigma_1 = \begin{bmatrix} +0.8 & -0.6 \\ -0.6 & +0.8 \end{bmatrix}, \quad N_1 = 100$$
$$\mu_2 = \begin{bmatrix} -5.0 \\ +5.0 \end{bmatrix}, \quad \Sigma_2 = \begin{bmatrix} +0.8 & +0.6 \\ +0.6 & +0.8 \end{bmatrix}, \quad N_2 = 100$$
$$\mu_3 = \begin{bmatrix} -5.0 \\ -5.0 \end{bmatrix}, \quad \Sigma_3 = \begin{bmatrix} +0.8 & -0.6 \\ -0.6 & +0.8 \end{bmatrix}, \quad N_3 = 100$$
$$\mu_4 = \begin{bmatrix} +5.0 \\ -5.0 \end{bmatrix}, \quad \Sigma_4 = \begin{bmatrix} +0.8 & +0.6 \\ +0.6 & +0.8 \end{bmatrix}, \quad N_4 = 100$$
$$\mu_5 = \begin{bmatrix} +5.0 \\ +0.0 \end{bmatrix}, \quad \Sigma_5 = \begin{bmatrix} +0.2 & +0.0 \\ +0.0 & +1.2 \end{bmatrix}, \quad N_5 = 100$$
$$\mu_6 = \begin{bmatrix} +0.0 \\ +5.0 \end{bmatrix}, \quad \Sigma_6 = \begin{bmatrix} +1.2 & +0.0 \\ +0.0 & +0.2 \end{bmatrix}, \quad N_6 = 100$$
$$\mu_7 = \begin{bmatrix} -5.0 \\ +0.0 \end{bmatrix}, \quad \Sigma_7 = \begin{bmatrix} +0.2 & +0.0 \\ +0.0 & +1.2 \end{bmatrix}, \quad N_7 = 100$$
$$\mu_8 = \begin{bmatrix} +0.0 \\ -5.0 \end{bmatrix}, \quad \Sigma_8 = \begin{bmatrix} +1.2 & +0.0 \\ +0.0 & +0.2 \end{bmatrix}, \quad N_8 = 100$$
$$\mu_9 = \begin{bmatrix} +0.0 \\ +0.0 \end{bmatrix}, \quad \Sigma_9 = \begin{bmatrix} +1.6 & +0.0 \\ +0.0 & +1.6 \end{bmatrix}, \quad N_9 = 200$$

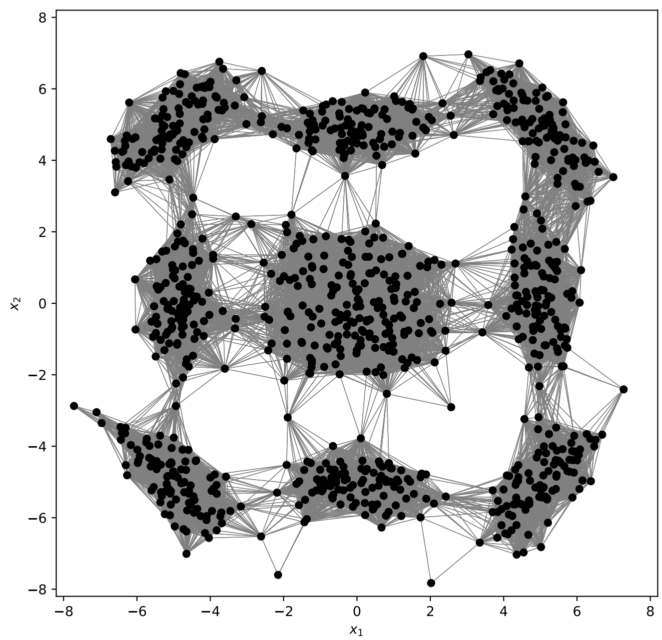
The given data points are shown in the following figure.



2. You should first calculate the Euclidean distances between the pairs of data points. The data point pairs with distance less than $\delta = 2.0$ are considered as connected. Construct the matrix $\mathbf{B}$ as follows:

$$b_{ij} = \begin{cases} 1, & \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|_2 < \delta \\ 0, & \text{otherwise.} \end{cases}$$
$$b_{ii} = 0$$

You should also visualize this connectivity matrix by drawing a line between two data points if they are connected. Your figure should be like the following figure. (20 points)



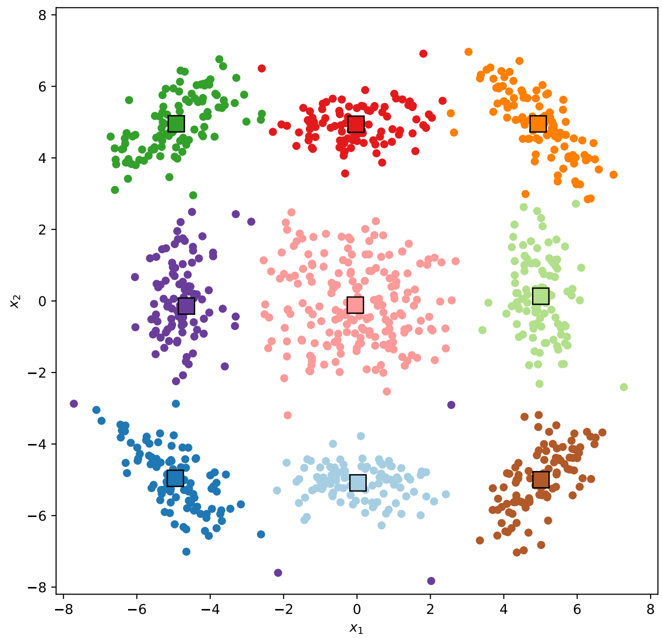
3. You should then calculate $\mathbf{D}$ and $\mathbf{L}$ matrices as described in the lecture notes. You should normalize the Laplacian matrix using the formula below. (20 points)

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

4. Find the eigenvectors of the normalized Laplacian matrix and pick $R = 5$ eigenvectors that corresponds to R smallest eigenvectors (eigenvectors that corresponds to 2$^{\text{nd}}$ smallest, 3$^{\text{rd}}$ smallest, 4$^{\text{th}}$ smallest, 5$^{\text{th}}$ smallest and 6$^{\text{th}}$ smallest eigenvalues since the smallest eigenvalue is 0). Using these eigenvectors construct the matrix $\mathbf{Z}$ as described in the lecture notes. Please note that the eigenvalues might not be returned in a decreasing or increasing order from the eig function. (20 points)

5. Run k-means clustering algorithm on $\mathbf{Z}$ matrix to find $K = 9$ clusters. When initializing your algorithm, use the following rows of $\mathbf{Z}$ matrix for initial centroids: 242, 528, 570, 590, 648, 667, 774, 891, and 955. (20 points)

6. Draw the clustering result obtained by your spectral clustering algorithm by coloring each cluster with a different color. Your figure should be like the following figure. (20 points)



What to submit: You need to submit your source code in a single file (.py file) named as ***STUDENTID.py***, where ***STUDENTID*** should be replaced with your 7-digit student number.

How to submit: Submit the file you created to Blackboard. Please follow the exact style mentioned and do not send a file named as ***STUDENTID.py***. Submissions that do not follow these guidelines will not be graded.

Late submission policy: Late submissions will not be graded.

Cheating policy: Very similar submissions will not be graded.