ENGR 421 / DASC 521: Introduction to Machine Learning
Homework 08: Expectation-Maximization Clustering

Deadline: May 20, 2022, 11:59 PM

In this homework, you will implement an expectation-maximization (EM) 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 `hw08_data_set.csv`, which contains 1000 data points generated randomly from nine bivariate Gaussian densities with the following parameters.

$$\mu_1 = \begin{bmatrix} +5.0 \\ +5.0 \end{bmatrix}, \qquad \Sigma_1 = \begin{bmatrix} +0.8 & -0.6 \\ -0.6 & +0.8 \end{bmatrix}, \qquad N_1 = 100$$

$$\mu_2 = \begin{bmatrix} -5.0 \\ +5.0 \end{bmatrix}, \qquad \Sigma_2 = \begin{bmatrix} +0.8 & +0.6 \\ +0.6 & +0.8 \end{bmatrix}, \qquad N_2 = 100$$

$$\mu_3 = \begin{bmatrix} -5.0 \\ -5.0 \end{bmatrix}, \qquad \Sigma_3 = \begin{bmatrix} +0.8 & -0.6 \\ -0.6 & +0.8 \end{bmatrix}, \qquad N_3 = 100$$

$$\mu_4 = \begin{bmatrix} +5.0 \\ -5.0 \end{bmatrix}, \qquad \Sigma_4 = \begin{bmatrix} +0.8 & +0.6 \\ +0.6 & +0.8 \end{bmatrix}, \qquad N_4 = 100$$

$$\mu_5 = \begin{bmatrix} +5.0 \\ +0.0 \end{bmatrix}, \qquad \Sigma_5 = \begin{bmatrix} +0.2 & +0.0 \\ +0.0 & +1.2 \end{bmatrix}, \qquad N_5 = 100$$

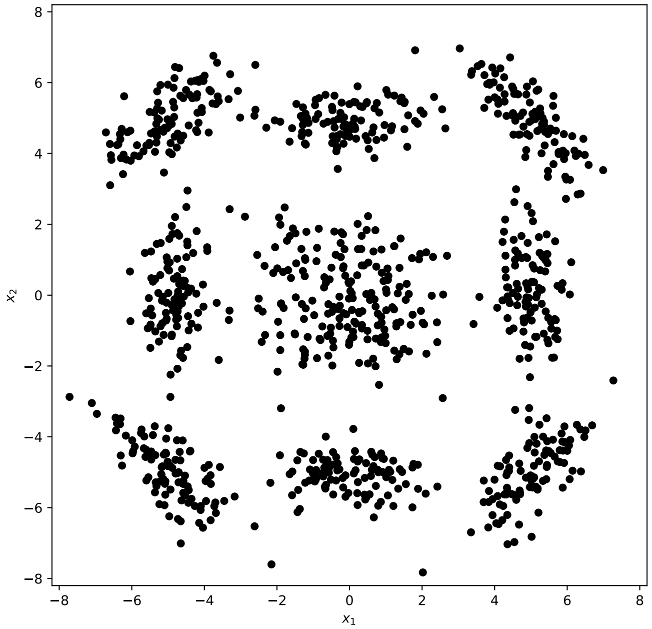
$$\mu_6 = \begin{bmatrix} +0.0 \\ +5.0 \end{bmatrix}, \qquad \Sigma_6 = \begin{bmatrix} +1.2 & +0.0 \\ +0.0 & +0.2 \end{bmatrix}, \qquad N_6 = 100$$

$$\mu_7 = \begin{bmatrix} -5.0 \\ +0.0 \end{bmatrix}, \qquad \Sigma_7 = \begin{bmatrix} +0.2 & +0.0 \\ +0.0 & +1.2 \end{bmatrix}, \qquad N_7 = 100$$

$$\mu_8 = \begin{bmatrix} +0.0 \\ -5.0 \end{bmatrix}, \qquad \Sigma_8 = \begin{bmatrix} +1.2 & +0.0 \\ +0.0 & +0.2 \end{bmatrix}, \qquad N_8 = 100$$

$$\mu_9 = \begin{bmatrix} +0.0 \\ +0.0 \end{bmatrix}, \qquad \Sigma_9 = \begin{bmatrix} +1.6 & +0.0 \\ +0.0 & +1.6 \end{bmatrix}, \qquad N_9 = 200$$

The given data points are shown in the following figure.



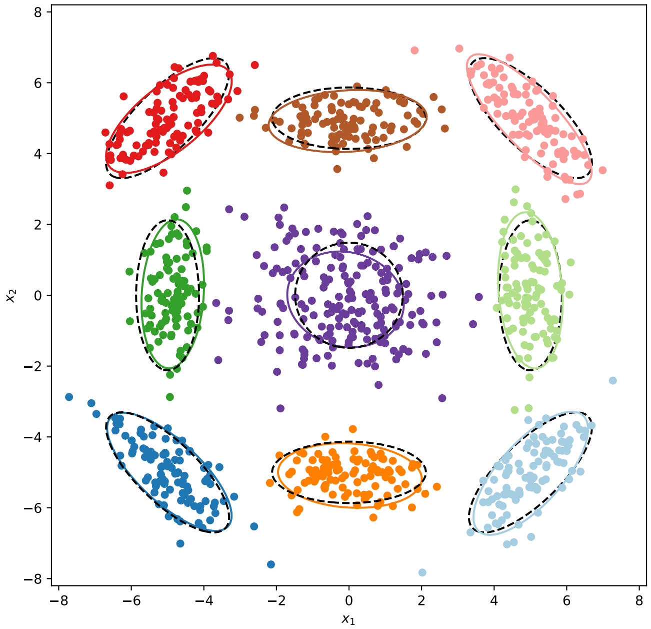
2. To initialize your EM algorithm, you should take the centroids given in the file named `hw08_initial_centroids.csv` as the initial values for the mean vectors. By assigning the data points to the nearest center, estimate the initial covariance matrices and prior probabilities in your EM algorithm. (20 points)

3. After the initialization step, run your EM algorithm for 100 iterations. Report the mean vectors your EM algorithm finds. Your results should be like the following matrix. (50 points)

```
print(means)

[[-4.9508988  -4.98464367]
 [-4.85629614  0.0404331 ]
 [-4.96379877  4.984647  ]
 [ 0.02477868 -5.09014979]
 [-0.09548618 -0.116943  ]
 [-0.03701877  4.91812108]
 [ 5.00933942 -5.02595861]
 [ 4.99839618  0.13777844]
 [ 4.96705774  4.97185503]]
```

4. Draw the clustering result obtained by your EM algorithm by coloring each cluster with a different color. You should also draw the original Gaussian densities you use to generate data points and the Gaussian densities your EM algorithm finds with dashed and solid lines, respectively. Draw these Gaussian densities where their values are equal to 0.05. Your figure should be like the following figure. (30 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.