ENGR-421 HW-7 REPORT FALL-2021

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

Initially, I have imported the necessary libraries (numpy, matplotlib.pyplot, and so on) . Then; by utilizing the np.genfromtxt function of the numpy library, I have read the "hw07_data_set.csv" file. While reading the data set; inside the np.genfromtxt function, I have set the delimiter to comma (delimiter= ","). Then, I have created the mean parameters, the covariance matrix parameters, and the cluster size parameters. By using the np.array function of the numpy library; I have concatenated the mean parameters, the cluster size parameters, and the covariance matrix parameters. After that, to update the memberships; I have defined a function called "update_mems". To update the centroids, I have defined a function called "update_cens". The update_cens function has a little difference compared to the one in the Lab11: Clustering. In this homework, we are expected to initialize the centroids to the given initial centroids data. So, to do this initialization, I have used the np.genfromtxt function of the numpy library, and set the delimiter "," inside the np.genfromtxt function (As the file name, I have passed the initial centroids csv file to the np.genfromtxt file. This hw07_initial_centroids.csv file is already given to us). Moreover; to plot the current state of the clusters; I have defined a function called "plot_cur_st". While implementing these functions; I have benefitted from the "Lab11: Clustering". Subsequently, I have implemented the k-means clustering algorithm. While implementing this k-means clustering algorithm, I have benefitted from "Lab11: Clustering".

Subsequently, I have implemented the Expectation-Maximization(EM) algorithm. EM algorithm involves E-STEP and M-STEP parts. While implementing the E-STEP algorithm and M-STEP algorithm, I have used the formulas given in the "Lecture 22-Clustering". Specifically, while implementing the E-STEP algorithm and M-STEP algorithm; I have benefitted from the 6th and 7th slides of the Lecture 22-Clustering. You can see the formulas I used while implementing the E-STEP algorithm in the Figure 3. Moreover, you can see the formulas I used while implementing the M-STEP algorithm in the Figure 4.

After 100 executions of my EM algorithm, I have found the mean vectors. To see the mean vectors that I have found after the 100 executions, you can see the Figure 1. After I found the mean vectors in Figure 1, I have plotted the original Gaussian densities with the dashed lines, and the Gaussian densities found by my EM algorithm with the solid lines. While drawing these lines, I have used the plt.contour function (from hw1) of the matplotlib.pyplot library of the python. You can see the plot showing the clusters after running the EM algorithm 100 times in Figure 2.

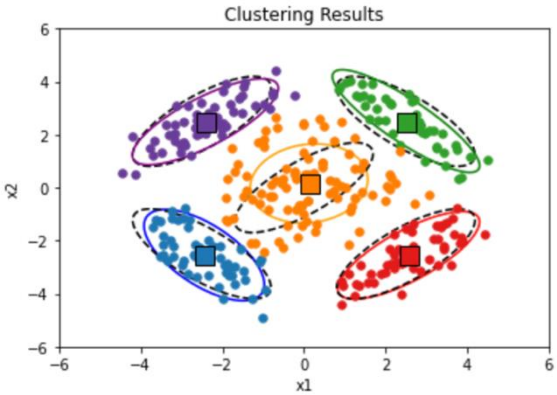

Figure 2: The plot showing the clusters after running EM algorithm 100 times

```
The Mean Vectors:

[[-2.44390007 -2.54539389]
 [ 2.5035433   2.51134852]
 [ 2.56726403 -2.55477256]
 [ 0.1279471   0.15595827]
 [-2.4146531   2.4855615 ]]
```

Figure 1: The mean vectors that I found after executing the EM algorithm 100 times

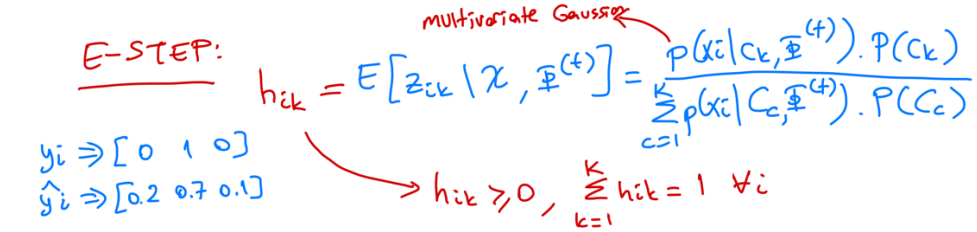

Figure 3: The formulas I used while implementing the E-STEP part of the EM algorithm

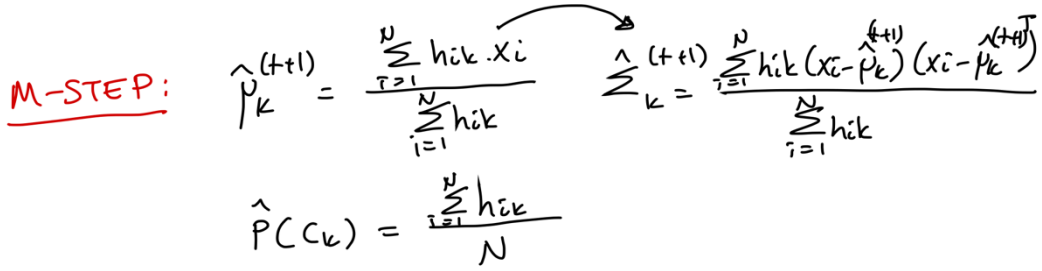

Figure 4: The formulas I used while implementing the M-STEP part of the EM algorithm (the formulas that I have used for updating the means, covariance matrices, and the prior probabilities in the M-STEP part of the EM algorithm)

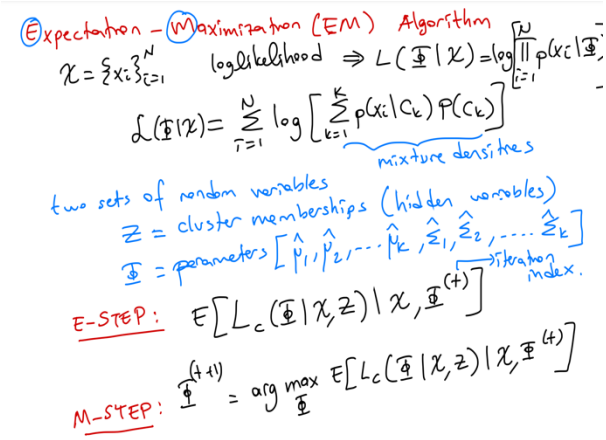

Figure 5: The general overview (including the formulas & brief definitions) of the E-STEP & M-STEP parts of the Expectation – Maximization(EM) Algorithm.

While implementing the M-STEP part of the EM algorithm, to update the means; first, I have calculated the summation of the prior probability*individual data points. Then, I have divided this summation with the summation of the cluster sizes. Furthermore; in the M-STEP algorithm, to update the covariance matrices; I have firstly found the summation of the prior probability*(individual data point-updated mean value)*(the transpose of the (individual data point-updated mean value)). Next, I have divided this summation by the summation of the prior probabilities, and calculated the update covariance matrix. In the M-STEP algorithm; to update the prior probabilities; I have firstly found the summation of the prior probabilities. Then; I have divided this summation by the summation of the cluster sizes, and calculated the updated prior probabilities (To see the formulas I used while implementing the M-STEP part, please see the Figure 4). In the E-STEP part of the EM algorithm, given the cluster memberships and the data points; I have found the likelihood of the parameters. Moreover, given the data points and the parameters (prior probabilities, means, and covariance matrices) , in the E-STEP, to update the prior probabilities in the E-STEP, I have calculated the expected value of this likelihood (To see the formulas I used while implementing the E-STEP part, please see the Figure 3). In addition to that, in the E-STEP part of my EM algorithm, I have selected the group which has the highest posterior probability (highest hik). (To see the formulas used in E-STEP, please see Figure 3).