ENGR421 – HW7 Report

I first initialized the Gaussian variables; means, covariances and class prior probabilities (which I named as frequencies in the code) and updated the initial memberships with the code provided in lab 11. To calculate the class covariances I first grouped the datapoints by these initialized memberships and calculated the covariances with np.cov.

Then to calculate the E step of the expectation maximixation algorithm that is given with formula,

$$h_{ik} = \frac{p(x_i \mid C_k, \Phi^{(t)}) \cdot P(C_k)}{\sum_{c=1}^{K} p(x_i \mid C_c, \Phi^{(t)}) \cdot P(C_c)}$$

(i denotes the datapoint index, k denotes the class number)

I used the multivariate gaussian distribution for class densities that is given by formula,

$$\frac{1}{\sqrt{(2\pi)^D |\Sigma_c|}} \cdot \exp\left[-\frac{1}{2} (x-\mu_c)^T \underline{\underline{\Sigma_c}}^{-1} \cdot (x-\mu_c) \right]$$

I calculated it for each point in the e step for a given datapoint and a given class and multiplied it with the class frequency. During the EM iterations I stored the values (as a denoted these as hik in the first equation) in a list as individual arrays for storing hik values for every datapoint and appended (1,300) hik calculations in a list making the total H list in dimensions (5,300) after being turned into a numpy array. Then as stated in its equations I summed the hik values of each datapoint for a given class and divided for the whole H array during the 100 EM iterations.

Then I wrote an m step method which takes the Hik values for a given class and the corresponding class label index. Here is where I update the class means by multiplying hik values with the corresponding datapoints for every datapoint and dividing it with the class hik values' summation. I also updated the class covariances by matrix multiplying the differences between datapoints and the current class means which is also multiplied with the hik values for all datapoints. This is also divided by the summation of hik values. The class prior probabilities are also updated by summing the hik values for the class divided by the number of datapoints. These all are done abiding the calculations we did in the lecture as follows,

$$\hat{\mu}_k^{(t+1)} = \frac{\sum_{i=1}^{N} hik \cdot xi}{\sum_{i=1}^{N} hik} \curvearrowright \hat{\Sigma}_k^{(t+1)} = \frac{\sum_{i=1}^{N} hik\,(xi - \hat{\mu}_k^{(t+1)})(xi - \hat{\mu}_k^{(t+1)})^T}{\sum_{i=1}^{N} hik}$$

$$\hat{P}(C_k) = \frac{\sum_{i=1}^{N} h_{ik}}{N}$$

To generate the 100 EM iterations, for each iteration I generated a clean H list to store all my hk values and called the e step for each class and after the H matrix is formed I called the m step for each class to update the class means, covarriances and frequencies using the H matrix.

Finally I updated the datapoint memberships with the provided method in Lab 11, and plotted the predicted memberships of the datapoints.