I first created the dataset according to given numbers in the pdf and stored them in X matrix then in order to show the data I plotted generated data using matplotlib. Then in order to find initial centroids and memberships to run EM algorithm I used K-means algorithm for two iteration it gives more reasonable initial points than randomly selecting. (Of course, in K-means initial points are randomly selected). I took k-means algorithm from the lab 11. I deleted some unnecessary parts and fixed the iteration size to 2. Then by using centroids and memberships matrixes I got from k-means I calculated initial priors, covariances and means (which are simply centroids) of classes.

Text, whiteboard

Description automatically generatedE-step) By using the formula below, I calculated the probability of points belonging to each class so, I obtained [300 x 5] matrix, then in order to normalize it I divided this matrix to the class probability summation matrix [300 x 1] for each data. As result I got normalized [300, 5]

Text, letter

Description automatically generatedM-step) By using the formula below, I calculated class priors, class means and class covariances by using the matrix I got in E-step.

I repeated E-step and M-step 100 iterations respectively.

After running EM algorithm 100 iterations I got my final class covariance, mean and prior matrixes. By using matplotlib I showed each class clustering in different color, by using meshgrid and contour method, I drawed solid ellipse for the points which have 0.05 probability of each class with the matrixes I calculated, also I drawed dashed ellipse for the points which have 0.05 probability of each class with the matrixes described in pdf in the same way.