1. As explicitly as possible, explain:

a. the meaning of $z_{i,j}$

   If x_i belongs to distribution with Θ_j then $z_{ij} = 1$ else it is 0, this is assuming that Θ is known.

b. which quantities need to be computed in the E-step and how they should be computed

   Assuming Θ(mean, stddev) of each distribution is known

   - Compute $f(x_i | \Theta_j)$ for each x_i and Θ_j combination using formula $\frac{1}{\sqrt{2\pi * 1 * 1}} * e^{-(x_i-\mu_j)^2/2}$
   - Using the computed $f(x_i | \Theta_j)$ for each x_i and Θ_j, compute expected value of each z_{ij} using the formula $E[z_{ij}] = f(x_i | \Theta_j) / \sum_{j=1 \text{ to } k} f(x_i | \Theta_j)$ where k is the number of clusters

c. which quantities needs to be computed in the M-step and how.

   Based on the computed z_{ij} value in the E-step, for each of the k distribution need to compute the new mean using the formula :

$$\mu_j = \frac{\sum_{i=1}^{n} x_i * E[z_{ij}]}{\sum_{i=1}^{n} E[z_{ij}]}$$

d. Additionally, on my M-step slide (the slide numbered 61), I omitted the steps needed to derive the boxed formula for μj from the E[\log L( )] expression two lines above. You should include the corresponding derivation to support your calculation of μj.

   Attached in zip scanned hand written copy- Derivation.pdf

3. Clearly state which methods you have chosen for both initialization and termination, and give a brief justification of why you think they are sensible choices.

Initialization : I am choosing k random integers between the minimum and maximum data point of the input data set, where k is the number of distributions. This helps in

better convergence of the data and does not make any assumption on the distribution of the input data.

Termination : Terminate when the difference between the log likelihood computed after the current E-step and log likelihood computed after the previous run E-step is less than 0.001. Another termination method I tried was stopping based on a fixed number of iteration(went up to 100). Noticed that using the epsilon difference as termination criteria helped to stop on convergence faster.

5. Deliverable: To tie all of this together, give us a written pseudocode description of the overall algorithm for *k*-component Gaussian clustering, plus selection of $1 \le k \le 5$ optimizing the BIC score, showing how the various quantities mentioned above are computed/used.

1. For each cluster $1 \le k \le 5$

a. Initialize mean array of size k with k random numbers between the minimum and maximum data point in the input set.

b. do {

b.1 if newLogLike != null => oldLogLik = newLogLik

b.1 E-step

b.1.1 for each x_i (where *i* is from 1 to n)and f_j (where j is from 1 to k) initialize $\texttt{normalDensityMatrix}[x_i][f_j] = \frac{1}{\sqrt{2\pi * 1 * 1}} * e^{-(x_i - \mu_j)^2/2}$

b.1.2 f or each x_i and f_j initialize $\texttt{expectedMatrix}[x_i][f_j] = \frac{\text{normalDensityMatrix[xi][fj]}}{\sum_{j=1}^{k} \text{normalDensityMatrix[xi][fj]}}$

b.2. newLogLike = Compute Log Likelihood using $\sum_{i=1}^{n} log\left(\frac{\sum_{j=1}^{k} \text{normalDensityMatrix[xi][fj]}}{k}\right)$

b.3. Compute BIC using 2 * ln *L(x | ϑ-hat) - r* ln *n*

b.4 M-step

b.4.1 update mean array such that for each meanArray[j] where j is from 1 to k :

$$\text{meanArray[j]} = \frac{\sum_{i=1}^{n} x_i * \text{expectedMatrix[xi][fj]}}{\sum_{i=1}^{n} \text{expectedMatrix[xi][fj]}}$$

} while (newLogLike – oldLogLik > epsilon)

6,7. **Deliverable** : Attached in the zip file.

8. Deciding whether there are, say, "three clusters" in the data is a difficult problem.

a. **Deliverable**: However, assuming there are exactly three, is the assumption that all three have a variance of 1 reasonable for that data? Why or why not?

Assume 3 clusters and variance 1, we'll get the points appropriately clustered by k-means for the data set above i.e. (9,10,11) (20,21,22) (46,49,55,57). But the final model is not very representative for the third cluster because only point 49 is reasonably close to the mean 51.75. The probability of seeing the other three points in the distribution is very less. The best option would be to consider variance as a parameter as well and estimate it for this dataset. However, if the number of clusters is assumed to be 4, having unit variance will give us a model which closely represents the data, but may be a case of over-fitting. It's very difficult to extrapolate beyond this based on the small sample.

b. **Deliverable**: Without assuming that you know the number of clusters, on the small sample above and the "Final" test data set, do the BIC scores for the various models seem to provide useful guidance in selecting a reasonable number of clusters?

Based on observing the small sample dataset and the final dataset, the BIC score is very low when the number of clusters is 1. With increasing k, the BIC scores increases until the most optimal number of clusters is hit, and starts to go down. It alternates up and down for different number of clusters, but there is a maximum point. So, it's a very good indicator to help us choose k.