Sahil Gandhi
Problem 1

Part 1. E Step proof

Taking the partial derivative, of the objective function w.r.t π and equating it with 0, we get,

$$\sum_i \left|\left|X_i - \mu_k\right|\right|^2 = 0$$

What this equation signifies that, in the E-step, the update of the memberships is done such that the Euclidian distances between that of the point X and the chosen π is minimized.

Part 2. M Step proof

Similar to the 1st step, taking the derivative of the objective function w.r.t μ, we get,

$$2\sum_i \pi_{ik}(X_i - \mu_k) = 0 \Rightarrow \mu_k = \frac{\sum_i \pi_{ik} X_i}{\sum_i \pi_{ik}}$$

The above equation signifies that minimizing the objective function during the M step, the algorithm chooses a centroid that minimizes the average-distance of all the points in a cluster from that centroid.

Part 3. Explain why KMeans has to stop (converge), but not necessarily to the global minimum objective value.

I will answer this in 2 parts:
Part a. Explain why K-means converges?

- We can partition N data points into k clusters in k^Nways. Hence, however large, this is a finite space. Meaning, we can produce all possible cluster combinations and pick the one that has the lowest cost.
- From a K-Means algorithm point of view, we generate new clusters based on the previous clusters. From Part 1 and 2, we have proved that at each step, the overall objective function reduces or remains the same.
- The iterations can keep running till the cost no longer reduces. Once we reach this stage, the cost will no longer reduce as the chosen centroids are the most optimal, and the chosen memberships is also the most optimal for the given scenario.
- Hence, the algorithm converges when the cost does not improve.

Part b. ..not necessarily the global optimum

- As mentioned above, the number of clusterings is exponential in the size of the data.
- This way, it is improbable to work with a brute-force algorithm that looks through all the clusters and picks best one
- Also, K-Means is an EM-algorithm, meaning we make certain assumptions of properties in one step to maximize the parameters in the current step. Because of this, we begin our algorithm with a randomly chosen centroids.
- In doing this, we maximize the objective function at each step, and this usually leads us to a local optimum instead of a global optima.