Textual Answers

a. Briefly describe what EM is and when using it is appropriate? Give an example use case?

Answer: The Expectation-Maximization algorithm is an iterative method which belongs to the class of latent variable models which are used to perform maximum likelihood/maximum a posteriori estimation of parameters with hidden/latent variables.

The EM algorithm is an iterative mechanism which continuously switches between E & M steps mentioned below:

i. **Initialization:** Acquire an initial estimate for the parameters.
ii. **E-Step:** Using the estimation of the parameters from previous step, compute the expected values of latent variables in the dataset.
iii. **M-Step:** Optimize/Maximize the parameters (that maximize a variant of the likelihood function) of the model (using previously estimated latent variables in the dataset).
iv. **Exit:** Recursively iterate E & M steps until the likelihood of the observations does not change significantly and then exit.

EM algorithm is efficient & specifically used to perform Density Estimation whenever the dataset is incomplete (contains hidden/latent variables). Classic Maximum Likelihood Estimation fails to derive this density estimate in the presence of latent variables.

Use case: EM algorithm is mostly used in Unsupervised Learning domain. The most famous variant of this algorithm is 'K-Means'. Thus, EM is widely used for Clustering (fitting a Gaussian Mixture model) and Density Estimation.

e. Does EM find the global optimum? Explain why/why not?

Answer: No! EM may perhaps not achieve globally optimal solution when working with Gaussian mixture models (which has large number of local optimas). It can converge to a local optima/saddle point and only guaranteed to converge to a point with zero gradient. Since, it converges to a point with zero gradient w.r.t the parameters, we might converge to a different local optima depending on the initial values. This is because, the likelihood (log) function is usually multi-modal and both E and M steps may <u>not</u> always maximize it. Thus, the best we can achieve is convergence to some local optimum at a constant ratio rate.

d. How does the algorithm perform? Briefly explain?

Observation: The likelihood values continuously increases until a convergence point. However, in different runs with other params, the likelihood values fluctuate a lot and the process till convergence is very slow.