Estimation - Recap of MLE + Bayesian estimation, Gaussian Mixture Model - EM algorithm.

Gaussian Mixture Model (GMM) - Full Theory and Code

1. Maximum Likelihood Estimation (MLE) & Bayesian Estimation

- MLE: Choose parameters that maximize likelihood.
- Bayesian Estimation: Incorporates prior knowledge with likelihood.

2. Gaussian Mixture Model (GMM)

- Data is assumed to come from a mixture of several Gaussian distributions.
- Each cluster has: mean (μ), covariance (Σ), mixing coefficient (π).
- EM algorithm is used for estimation.

3. EM Algorithm Intuition

- Start with guesses for parameters.
- E-step: Calculate probability of each point belonging to each cluster (soft assignment).
- M-step: Update cluster parameters using these probabilities.
- Repeat until convergence.

4. Hard Assignment vs Soft Assignment

- Hard assignment: Each point belongs to exactly one cluster (like K-means).
- Soft assignment: Each point belongs to all clusters with probabilities (like GMM).

5. Choosing K (Number of Clusters)

- Likelihood always increases with K (risk of overfitting).
- Use penalized likelihood criteria:

* AIC = 2p - 2ln(L)
* BIC = p ln(n) - 2ln(L)

- Select K that minimizes AIC or BIC.

6. GMM Algorithm Implementation

- From Scratch: Implemented using E-step, M-step, log-likelihood computation.
- With sklearn: GaussianMixture class handles EM algorithm.

* gmm.fit(X)
* gmm.predict(X) → hard assignments
* gmm.predict_proba(X) → soft assignments

7. Visualization

- EM iterations move Gaussian ellipses to align with data distribution.

- Convergence when parameters stabilize.

Python Code Examples:

- Custom implementation of GMM with EM (NumPy + SciPy).

- Sklearn GaussianMixture example with visualization and printing of weights, means, and covariances.