Gaussian Mixture Models and the EM Algorithm

Machine Learning (CE 40717) — Spring 2025

Ali Sharifi-Zarchi

CE Department
Sharif University of Technology

November 22, 2025

Motivation

- Many real-world densities are **multi-modal**. These arise from heterogeneous sources, sub-populations, or hidden structure.
- We want a **flexible probabilistic model** to:
  - model complex densities,
  - perform soft clustering,
  - generate new samples,
  - evaluate likelihood.
- Gaussian Mixture Models (GMMs) achieve this with a simple generative structure.

Outline (full version)

- Mixture models and Gaussian mixture definition
- Maximum Likelihood for incomplete-data likelihood
- EM Algorithm:
  - E-step: responsibility computation
  - M-step: closed-form updates
  - full derivations (means, covariances, mixing weights)
- Variational lower bound, KL divergence
- Convex vs. concave functions (full explanation)
- Full proof of EM monotonicity
- Examples, figures, EM iterations
- Practical notes and failure modes

Multivariate Gaussian Distribution

A **multivariate Gaussian** (normal) distribution over $\mathbf{x} \in \mathbb{R}^d$ is defined by a mean vector $\boldsymbol{\mu} \in \mathbb{R}^d$ and a positive definite covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$:

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}).$$

Density function:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right).$$

Key properties:

- Elliptical level sets defined by the quadratic form $(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$.
- $\boldsymbol{\Sigma}$ encodes variances and pairwise correlations.
- Fully characterized by first and second moments.

This distribution will serve as the building block for Gaussian mixture models.

Variance–Covariance Matrix

For a multivariate Gaussian $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the covariance matrix

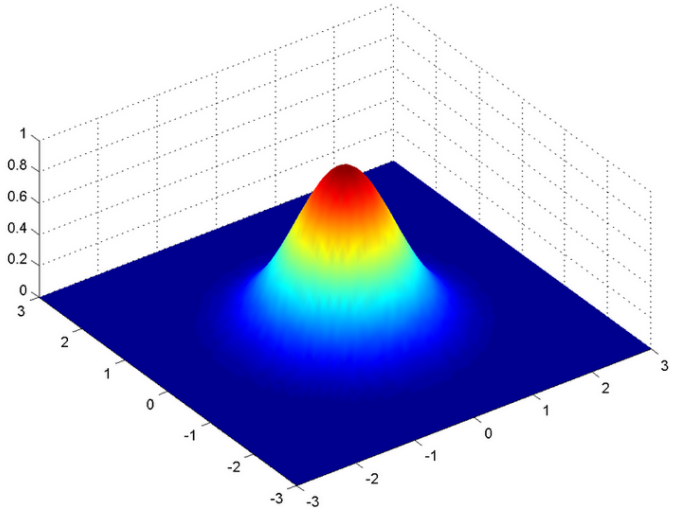
$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathrm{Var}(x_1) & \mathrm{Cov}(x_1, x_2) & \cdots \\ \mathrm{Cov}(x_2, x_1) & \mathrm{Var}(x_2) & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

encodes how each dimension varies and how pairs of dimensions interact.

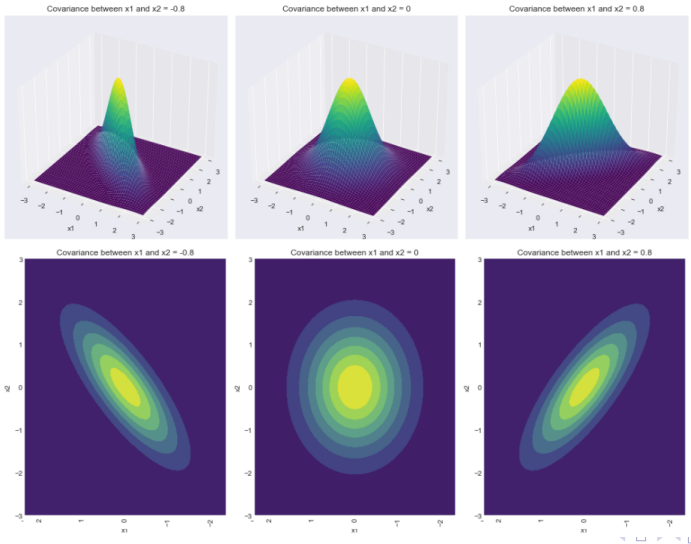
Interpretation of entries:

- Diagonal entries: $\Sigma_{ii} = \mathrm{Var}(x_i)$ — how spread out the distribution is along dimension i.
- Off-diagonal entries: $\Sigma_{ij} = \mathrm{Cov}(x_i, x_j)$ — how dimensions i and j move together.
- Positive covariance: x_i increases when x_j increases $\Rightarrow$ elongated ellipse along the diagonal direction.
- Negative covariance: x_i increases when x_j decreases $\Rightarrow$ ellipse tilted in the opposite direction.

Multivariate Gaussian Distribution



Multivariate Gaussian Distribution



What is a Mixture Model?

A mixture model represents a probability density as a convex combination of simpler component densities:

$$\mathbb{P}(\mathbf{x} \mid \boldsymbol{\theta}) = \sum_{j=1}^{K} \pi_j \mathbb{P}(\mathbf{x} \mid z = j; \boldsymbol{\theta}_j)$$

where:

- $\mathbf{x} \in \mathbb{R}^d$ is a **feature vector**,
- z is a latent discrete variable,
- $\pi_j \geq 0, \sum_j \pi_j = 1$.

The latent variable z indicates which component generated $\mathbf{x}$.

Gaussian Mixture Model (GMM)

$$\mathbb{P}(\mathbf{x}) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x} \mid \mu_j, \Sigma_j)$$

The multivariate Gaussian density:

$$\mathcal{N}(\mathbf{x} \mid \mu, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^\top \Sigma^{-1}(\mathbf{x}-\mu)\right).$$

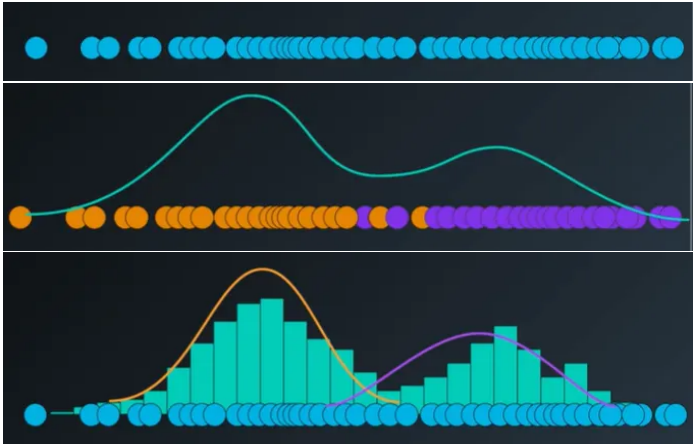
Parameters:

$$\boldsymbol{\theta} = \{\pi_j, \mu_j, \Sigma_j\}_{j=1}^{K}.$$

Notation (vector-corrected)

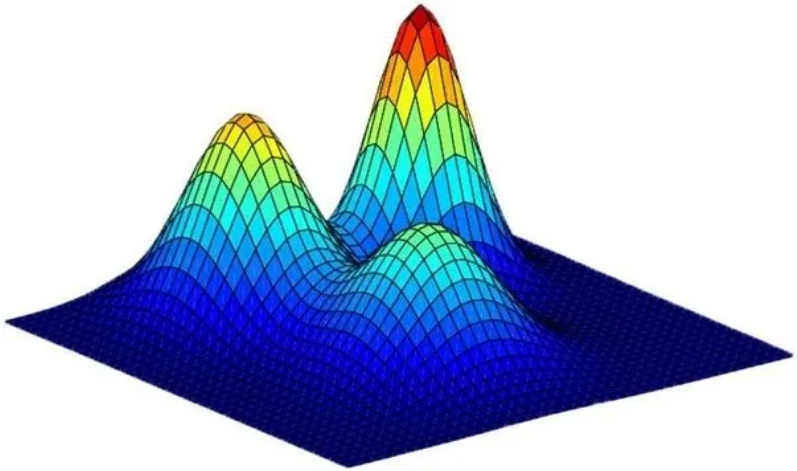
- $\mathbf{x}^{(i)}$: feature vector for sample i.
- $x_t^{(i)}$: scalar component t of $\mathbf{x}^{(i)}$.
- $z^{(i)}$: latent component label (categorical). One-hot form: $z_j^{(i)} \in \{0, 1\}$.
- π_j: mixing weight (scalar).
- μ_j: mean vector of component j.
- Σ_j: covariance matrix of component j.
- N: number of samples; d: dimension.

GMM



Source: Tilak Mudgal, "Gaussian Mixture Modeling (GMM)," Medium.

Multivariate GMM



Source: Tilak Mudgal, "Gaussian Mixture Modeling (GMM)," Medium.

Why GMMs?

- Can approximate arbitrary continuous densities (universal approximators).
- Soft clustering via posterior probabilities.
- Generative: can sample new data.
- More expressive than k-means due to covariance matrices.

Incomplete-data Log-Likelihood

Given data $\mathscr{X} = \{\mathbf{x}^{(i)}\}_{i=1}^{N}$, the likelihood is:

$$\mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} \sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}^{(i)} \mid \mu_j, \Sigma_j).$$

Hence the incomplete-data log-likelihood is:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{N} \ln \left(\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}^{(i)} \mid \mu_j, \Sigma_j) \right).$$

- The main difficulty: $\ln(\sum_j \cdot)$ makes direct maximization impossible.
- EM solves this by introducing latent variables.

Latent Variables and Complete-data Likelihood

Let $z_j^{(i)} \in \{0, 1\}$ indicate which component generated $\mathbf{x}^{(i)}$:

$$\sum_{j=1}^{K} z_j^{(i)} = 1.$$

The complete-data likelihood factorizes:

$$\mathbb{P}(\mathscr{X}, Z \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} \prod_{j=1}^{K} \left[\pi_j \mathcal{N}(\mathbf{x}^{(i)} \mid \mu_j, \Sigma_j) \right]^{z_j^{(i)}}.$$

Complete-data log-likelihood:

$$\ln \mathbb{P}(\mathscr{X}, Z \mid \boldsymbol{\theta}) = \sum_{i=1}^{N} \sum_{j=1}^{K} z_j^{(i)} \left(\ln \pi_j + \ln \mathcal{N}(\mathbf{x}^{(i)} \mid \mu_j, \Sigma_j) \right).$$

E-step: Responsibilities

In E-step we compute the posterior:

$$\gamma_j^{(i)} = \mathbb{P}(z_j^{(i)} = 1 \mid \mathbf{x}^{(i)}, \boldsymbol{\theta}^{(t)}) = \frac{\pi_j^{(t)} \mathcal{N}(\mathbf{x}^{(i)} \mid \mu_j^{(t)}, \Sigma_j^{(t)})}{\sum_{k=1}^{K} \pi_k^{(t)} \mathcal{N}(\mathbf{x}^{(i)} \mid \mu_k^{(t)}, \Sigma_k^{(t)})}.$$

Define:

$$N_j = \sum_{i=1}^{N} \gamma_j^{(i)}.$$

This represents the soft count for component j.

Q-function (Expectation of Complete-data LL)

$$Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) = \mathbb{E}_{Z|\mathcal{X},\boldsymbol{\theta}^{(t)}} \left[\ln \mathbb{P}(\mathcal{X}, Z \mid \boldsymbol{\theta}) \right]$$

$$= \sum_{i=1}^{N} \sum_{j=1}^{K} \gamma_j^{(i)} \left(\ln \pi_j + \ln \mathcal{N}(\mathbf{x}^{(i)} \mid \mu_j, \Sigma_j) \right).$$

Maximizing Q w.r.t. $\boldsymbol{\theta}$ gives M-step updates.

M-step: Update for μ_j (Full Derivation)

We maximize the part of Q involving μ_j:

$$Q_{\mu_j} = \sum_{i=1}^{N} \gamma_j^{(i)} \left(-\frac{1}{2} (\mathbf{x}^{(i)} - \mu_j)^\top \Sigma_j^{-1} (\mathbf{x}^{(i)} - \mu_j) \right) + \text{const.}$$

Take derivative:

$$\frac{\partial Q_{\mu_j}}{\partial \mu_j} = \sum_{i=1}^{N} \gamma_j^{(i)} \Sigma_j^{-1} (\mathbf{x}^{(i)} - \mu_j).$$

Set to zero:

$$\sum_{i=1}^{N} \gamma_j^{(i)} \mathbf{x}^{(i)} = \left(\sum_{i=1}^{N} \gamma_j^{(i)} \right) \mu_j.$$

Thus:

$$\boxed{\mu_j^{\text{new}} = \frac{1}{N_j} \sum_{i=1}^{N} \gamma_j^{(i)} \mathbf{x}^{(i)}}$$

M-step: Update for Σ_j (Full Derivation)

We maximize:

$$Q_{\Sigma_j} = \sum_{i=1}^{N} \gamma_j^{(i)} \left[-\frac{1}{2} \ln |\Sigma_j| - \frac{1}{2} (\mathbf{x}^{(i)} - \mu_j)^\top \Sigma_j^{-1} (\mathbf{x}^{(i)} - \mu_j) \right].$$

Derivative identity:

$$\frac{\partial}{\partial \Sigma} \ln |\Sigma| = (\Sigma^{-1})^\top, \qquad \frac{\partial}{\partial \Sigma} \mathbf{a}^\top \Sigma^{-1} \mathbf{a} = -\Sigma^{-1} \mathbf{a}\mathbf{a}^\top \Sigma^{-1}.$$

Setting derivative to zero yields:

$$\boxed{\Sigma_j^{\text{new}} = \frac{1}{N_j} \sum_{i=1}^{N} \gamma_j^{(i)} (\mathbf{x}^{(i)} - \mu_j^{\text{new}})(\mathbf{x}^{(i)} - \mu_j^{\text{new}})^\top}.$$

For numerical stability, in practice:

$$\Sigma_j \leftarrow \Sigma_j + \epsilon I.$$

M-step: Update for π_j - Part 1

We maximize:

$$Q_\pi = \sum_{j=1}^{K} N_j \ln \pi_j.$$

Subject to:

$$\sum_{j=1}^{K} \pi_j = 1.$$

Using a Lagrange multiplier:

$$\mathscr{L} = \sum_j N_j \ln \pi_j + \lambda \left(\sum_j \pi_j - 1 \right).$$

M-step: Update for π_j - Part 2

Derivative:

$$\frac{\partial \mathscr{L}}{\partial \pi_j} = \frac{N_j}{\pi_j} + \lambda = 0.$$

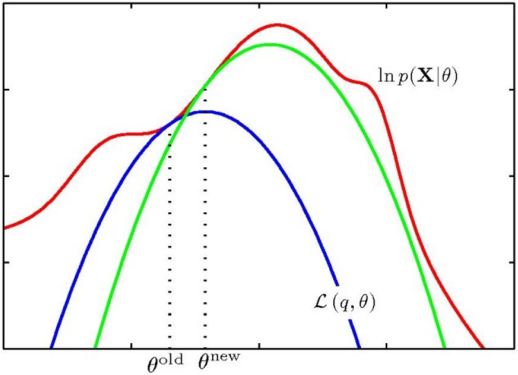
Solve:

$$\pi_j = -\frac{N_j}{\lambda}.$$

Apply constraint $\rightarrow$ final update:

$$\boxed{\pi_j^{\text{new}} = \frac{N_j}{N}}$$

EM Algorithm Illustration



- EM alternates between computing posteriors (E-step)
- and maximizing expected complete-data LL (M-step)
- until convergence.

Convex vs. Concave Functions (Full Explanation)

A function $f : \mathbb{R}^d \to \mathbb{R}$ is:

- **Convex** if
$$f(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y}), \quad 0 \leq \alpha \leq 1.$$
Its epigraph forms a convex set.
- **Concave** if
$$f(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) \geq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y}).$$
- **Strictly convex** if inequality is strict.

Key fact: $-\ln x$ is convex; $\ln x$ is concave.
This property is used to derive Jensen's inequality:

$$\ln \mathbb{E}[X] \geq \mathbb{E}[\ln X].$$

Variational Lower Bound via Jensen's Inequality

We start with:

$$\ln \mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}) = \ln \sum_{Z} \mathbb{P}(\mathscr{X}, Z \mid \boldsymbol{\theta}).$$

Introduce any distribution $Q(Z)$:

$$= \ln \sum_{Z} Q(Z) \frac{\mathbb{P}(\mathscr{X}, Z \mid \boldsymbol{\theta})}{Q(Z)}.$$

Apply Jensen:

$$\ln \mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}) \geq \sum_{Z} Q(Z) \ln \frac{\mathbb{P}(\mathscr{X}, Z \mid \boldsymbol{\theta})}{Q(Z)}$$

Define:

$$F(\boldsymbol{\theta}, Q) = \sum_{Z} Q(Z) \ln \frac{\mathbb{P}(\mathscr{X}, Z \mid \boldsymbol{\theta})}{Q(Z)}.$$

And:

$$\ln \mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}) = F(\boldsymbol{\theta}, Q) + \mathrm{KL}\big(Q(Z) \,\|\, \mathbb{P}(Z \mid \mathscr{X}, \boldsymbol{\theta})\big).$$

Why EM Increases Likelihood (Full Proof) - Part 1

At iteration t:

$$Q^{(t)}(Z) = \mathbb{P}(Z \mid \mathscr{X}, \boldsymbol{\theta}^{(t)}).$$

Then:

$$\ln \mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}^{(t)}) = F(\boldsymbol{\theta}^{(t)}, Q^{(t)}).$$

M-step chooses:

$$\boldsymbol{\theta}^{(t+1)} = \operatorname*{argmax}_{\boldsymbol{\theta}} F(\boldsymbol{\theta}, Q^{(t)}).$$

Thus:

$$F(\boldsymbol{\theta}^{(t+1)}, Q^{(t)}) \geq F(\boldsymbol{\theta}^{(t)}, Q^{(t)}).$$

Why EM Increases Likelihood (Full Proof) - Part 2

Since KL divergence is always non-negative:

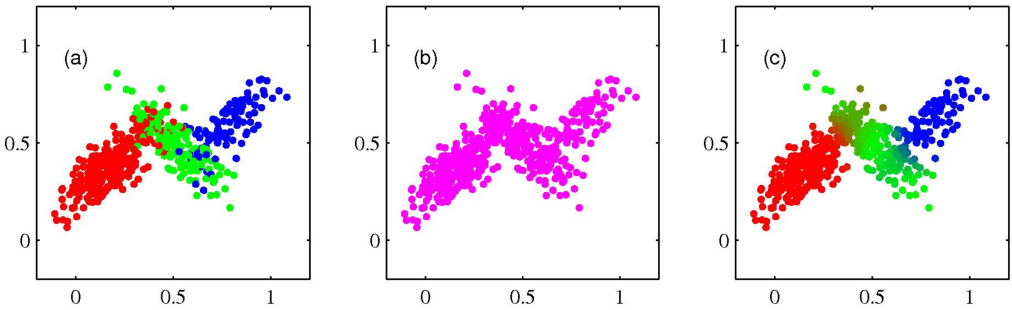
$$\ln \mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}^{(t+1)}) \geq F(\boldsymbol{\theta}^{(t+1)}, Q^{(t)}).$$

Combine them:

$$\boxed{\ln \mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}^{(t+1)}) \geq \ln \mathbb{P}(\mathscr{X} \mid \boldsymbol{\theta}^{(t)})}$$

$\rightarrow$ EM **always increases likelihood**.

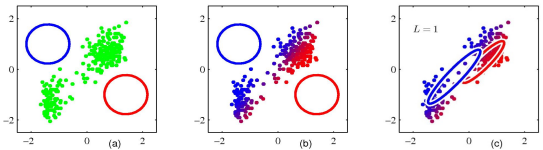
EM & GMM Example



(a) (b) (c)

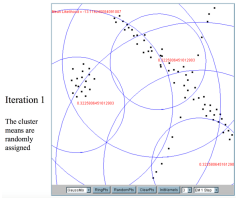
- Dataset contains two overlapping Gaussian components.
- Colors illustrate **soft assignments** (responsibilities).
- Means gradually move toward cluster centers as EM iterates.

EM & GMM Example

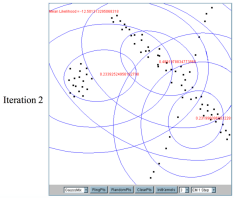


- Six-step visualization of EM evolution in 2D.
- Components reshape (covariance) according to local density.
- Shows **different initializations → different convergence paths**.

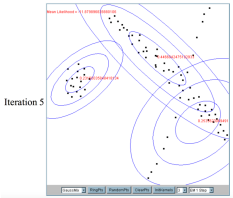
EM Iterations — Evolution of Parameters



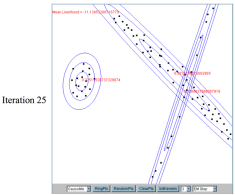
Iter 1



Iter 2



Iter 5



Iter 25

- Early steps: drastic changes in means/covariances.
- Later steps: parameters stabilize and likelihood increases slowly.
- EM often converges smoothly after a few iterations.

GMM 1-D Example — Detailed

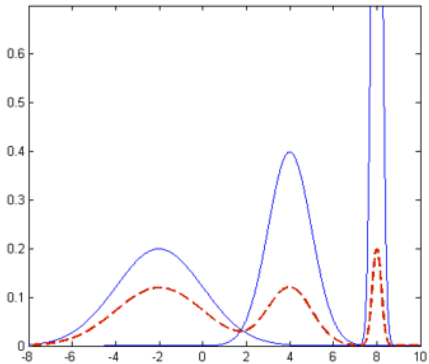
- True mixture:

$$\pi_1 = 0.6, \quad \pi_2 = 0.4$$

$$\mu_1 = 0, \quad \mu_2 = 3$$

$$\sigma_1^2 = 0.5, \quad \sigma_2^2 = 1.0$$

- EM initialized with random means and equal variances.
- After convergence, EM recovers mixture parameters accurately.
- Plot shows true density vs. EM estimate.

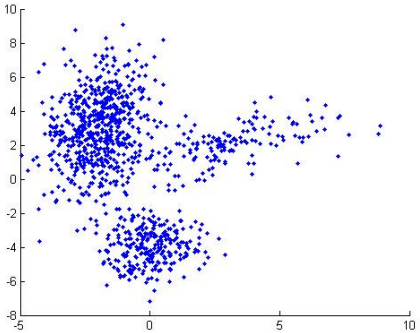


GMM 2-D Example — Detailed

- Synthetic dataset with $K = 3$ components.
- Components have **anisotropic covariances**:

$$\Sigma_j \text{ not diagonal}$$

- EM visually fits ellipses corresponding to covariance contours.
- Soft assignments shown by color gradient.



Convergence: Monotonicity and Local Optima

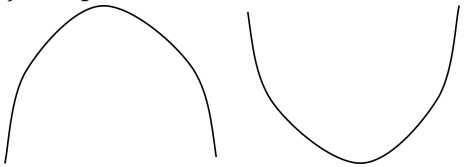
- EM **always increases** the log-likelihood:

$$\ell(\boldsymbol{\theta}^{(t+1)}) \geq \ell(\boldsymbol{\theta}^{(t)}).$$

- EM converges to a stationary point:

$$\nabla \ell(\boldsymbol{\theta}) = 0.$$

- This stationary point may be a:
  - local maximum,
  - saddle point.
- Initialization has major impact on final solution.



Concave LL Convex LL

Practical Tips for Using EM in GMMs

- **Initialization**:
  - k-means
  - k-means++ for more robustness
  - multiple random restarts
- **Covariance choices**:

  full, diagonal, tied, spherical
- **Stopping criteria**:
  - $\Delta \ell < \epsilon$
  - small parameter change
  - max iterations
- **Regularization**:

$$\Sigma_j \leftarrow \Sigma_j + \epsilon I \qquad (\epsilon \approx 10^{-5})$$

Failure Modes and Solutions

- **Component collapse**:

$$\Sigma_j \to 0$$

  Solution: add regularization, reinitialize collapsed component.
- **Singular covariance**: happens when one component takes a single point.
- **Slow convergence**: Solutions:
  - better initialization
  - annealing/tempering EM
  - variational EM
  - stochastic EM
- **Overfitting**: use MAP estimation or Bayesian GMM.

Comparison: EM for GMM vs. k-means

- k-means:
  - hard assignments
  - spherical clusters
  - minimizes within-cluster square distances
- EM + GMM:
  - soft assignments ($\gamma_j^{(i)}$)
  - full covariance modeling
  - probabilistic interpretation (likelihood)
- EM is more flexible—but requires more parameters and computation.

Summary

- GMMs provide flexible density estimation with multiple Gaussian components.
- EM algorithm:
  - E-step computes responsibilities.
  - M-step updates μ_j, Σ_j, π_j.
- EM maximizes a lower bound and increases likelihood monotonically.
- Initialization and regularization are critical to good performance.
- GMMs outperform k-means when clusters are anisotropic or overlapping.

References

- C. M. Bishop, *Pattern Recognition and Machine Learning*, Chapter 9.
- Original slides from:

  Hamid R. Rabiee & Zahra Dehghanian (Spring 2025)

- Additional illustrations from course-provided materials.

Acknowledgement

Special Thanks

Soheil Sayah Varg

for contributions to preparing, organizing, and refining this slide deck.

Thank You!

Questions?