Foundations of Machine Learning
AI2000 and AI5000

FoML-27
GMM

Dr. Konda Reddy Mopuri
Department of AI, IIT Hyderabad
July-Nov 2025

భారతీయ సాంకేతిక విజ్ఞాన సంస్థ హైదరాబాద్
भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Data-driven Intelligence
& Learning Lab

So far in FoML

- Intro to ML and Probability refresher
- MLE, MAP, and fully Bayesian treatment
- Supervised learning
    a. Linear Regression with basis functions
    b. Bias-Variance Decomposition
    c. Decision Theory - three broad classification strategies
    d. Neural Networks
- Unsupervised learning
    a. K-Means, Hierarchical clustering

For today

- GMM for clustering

Contents are taken from - Andrew Moore, CMU

Hard vs. Soft clustering

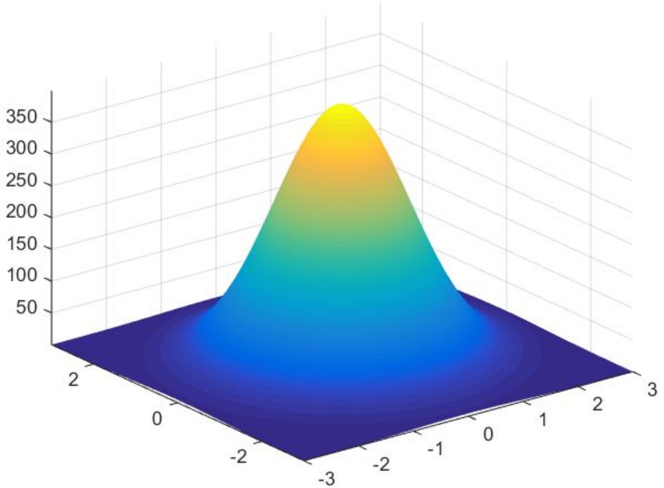
- Based on the overlap of clusters
  - Hard clustering - no overlap, complete/single assignment
  - Soft clustering - strength of association between element and cluster

Soft Clustering

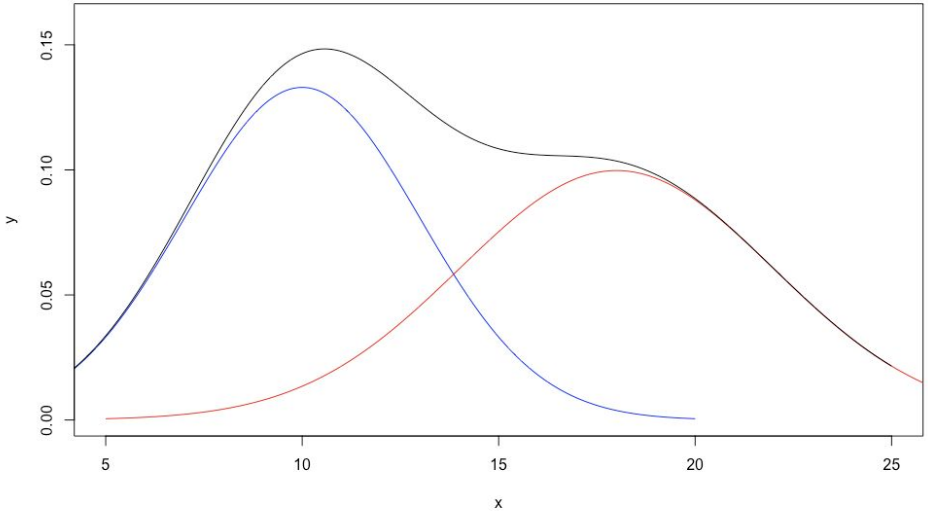
- Gives probabilities that an instance belongs to each of the cluster centers
- Each instance is assigned a probability distribution across a set of discovered categories/clusters

Gaussian Distribution

$$P(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\intercal \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$



Mixture of Gaussians

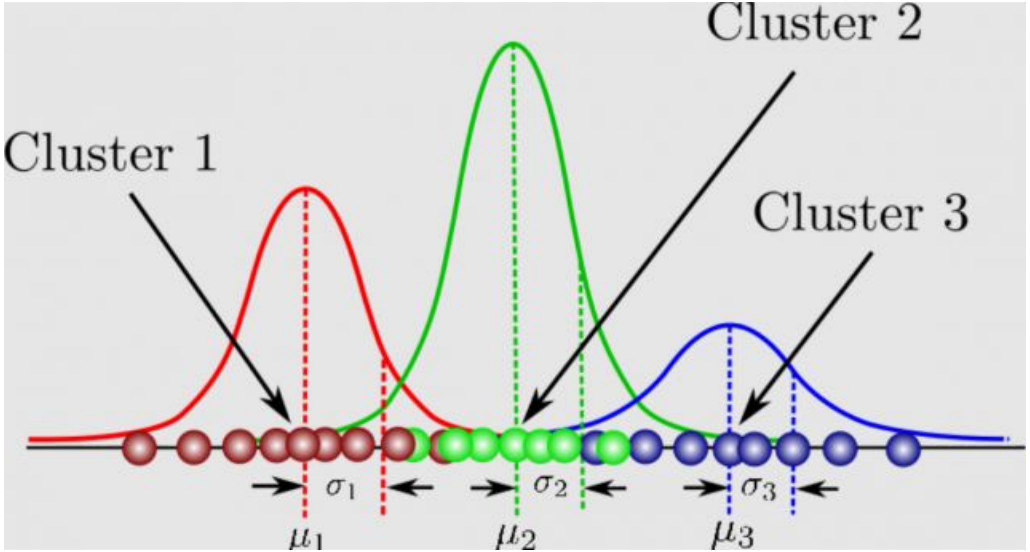


Credits - Yufeng

The GMM setting

- There are K components - w_i
- Each has associated mean and covariance - μ_i and Σ_i

Mixture of Gaussians



The GMM setting

- Each instance is assumed to be generated as follows
  - Pick a Gaussian component at random - with a probability $p(w_i)$
  - Sample from it - $N(\mu_i , \Sigma_i)$

Mixture Model

- Weighted sum of a number of pdfs where the weights are determined by a distribution $\boldsymbol{\pi}$

$$p(x) = \pi_0 f_0(x) + \pi_1 f_1(x) + \pi_2 f_2(x) + \ldots + \pi_k f_k(x)$$

$$\text{where } \sum_{i=0}^{k} \pi_i = 1$$

to satisfy unit area

$$\boxed{p(x) = \sum_{i=0}^{k} \pi_i f_i(x)}$$

GMM

- Weighted sum of a number of Gaussians where the weights are determined by a distribution $\boldsymbol{\pi}$

$$p(x) = \pi_0 N(x|\mu_0, \Sigma_0) + \pi_1 N(x|\mu_1, \Sigma_1) + \ldots + \pi_k N(x|\mu_k, \Sigma_k)$$

$$\text{where } \sum_{i=0}^{k} \pi_i = 1$$

$$\boxed{p(x) = \sum_{i=0}^{k} \pi_i N(x|\mu_k, \Sigma_k)}$$

total params

K Scalars π_i

$2K$ vectors $\underline{\mu_i}$

$2K$ matrices Σ_i

Expectation Maximization for GMMs

- Iterate until convergence
  - E-step
  - M-step

Expectation Maximization for GMMs

- The E-step

compute the expected classes|clusters for all the data instances

$$p(w_i | x_k, \lambda_t) = \frac{p(x_k | w_i, \lambda_t) \cdot p(w_i | \lambda_t)}{p(x_k | \lambda_t)}$$

K

$\{ \mu_i, \Sigma_i \}$

$\lambda_t = (\mu, \Sigma)^t$

$\pi_t = p(w_i | \lambda_t)$

Expectation Maximization for GMMs

- The E-step

compute the expected classes|clusters for all the data instances

$i = 1$ to K

$$P(w_i \mid x_k, \lambda_t) = \frac{P(x_k \mid w_i, \lambda_t) \cdot P(w_i \mid \lambda_t)}{P(x_k \mid \lambda_t)}$$

$$= \frac{P(x_k \mid w_i, \mu_i(t), \Sigma_i(t)) \cdot \pi_i(t)}{\sum_{j=1}^{c} P(x_k \mid w_j, \mu_j(t), \Sigma_j(t)) \cdot \pi_j(t)}$$

$\mathcal{N}(\mu_k, \Sigma_k)$

Expectation Maximization for GMMs

- The E-step

$$= \frac{\overbrace{P(x_k \mid w_i, \mu_i(t), \Sigma_i(t)) \cdot \pi_i(t)}}{\sum_{j=1}^{c} \underbrace{P(x_k / w_j, \mu_j(t), \Sigma_j(t)) \cdot \pi_j(t)}}$$

Evaluating the likelihood of sample x_k with one Gaussian

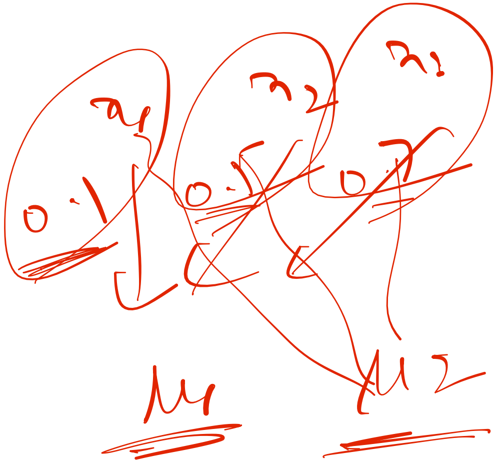
Expectation Maximization for GMMs

- The M-step

compute the maximum likeli μ_s and Σ_s given our assignments (membership distributions)

Expectation Maximization for GMMs

- The M-step



compute the maximum likeli μs and Σs given our assignments (membership distributions)

$$\mu_i(t+1) = \frac{\sum_k P(w_i / x_k, \lambda_t) \cdot x_k}{\sum_k P(w_i / x_k, \lambda_t)}$$

→ equivalent of sample mean, but with partial/soft assignments (weighted mean)

Expectation Maximization for GMMs

- The M-step

compute the maximum likeli μs and Σs given our assignments (membership distributions)

$$\mu_i(t+1) = \frac{\sum_k P(w_i | x_k, \lambda_t) \cdot x_k}{\sum_k P(w_i | x_k, \lambda_t)}$$

$$\Sigma_i(t+1) = \frac{\sum_k P(w_i | x_k, \lambda_t) \cdot [x_k - \mu_i(t+1)][x_k - \mu_i(t+1)]^T}{\sum_k P(w_i | x_k, \lambda_t)}$$

x_1

x_N

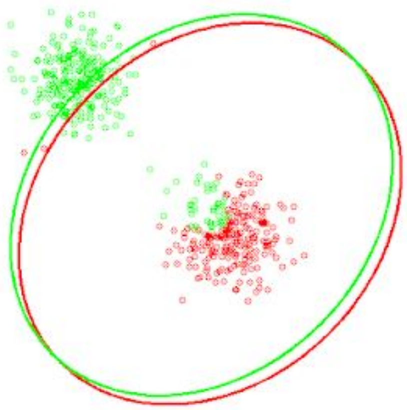
Expectation Maximization for GMMs

- The M-step

$$\pi_i(t+1) = \frac{\sum_k p(\omega_i \mid x_k, \lambda_t)}{\underbrace{N}_{\text{No. of observations (or, dataset size)}}}$$

GMM clustering - Example

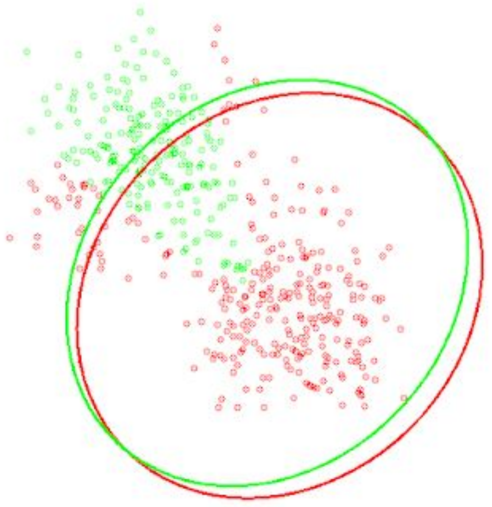
Clear separation between clusters



Source - seevisionc.blogspot.com

GMM clustering - Example

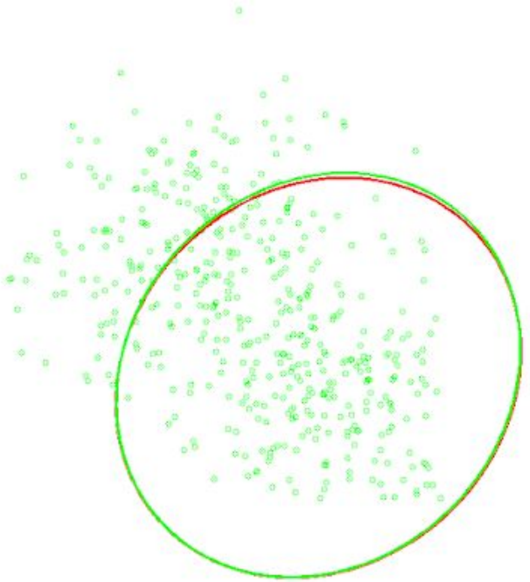
Increased variance - ambiguous boundary



Source - seevisionc.blogspot.com

GMM clustering - Example

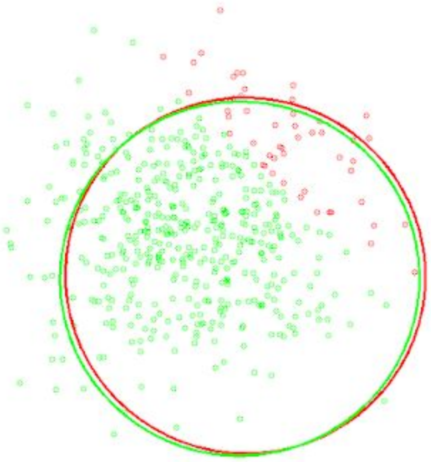
Further increasing the variance - blurs the boundary



Source - seevisionc.blogspot.com

GMM clustering - Example

When no clusters are present



Source - seevisionc.blogspot.com

Next

- UnSupervised Learning - PCA

Rough work