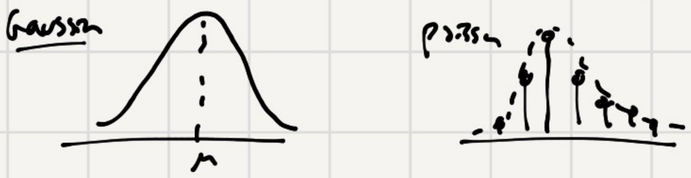
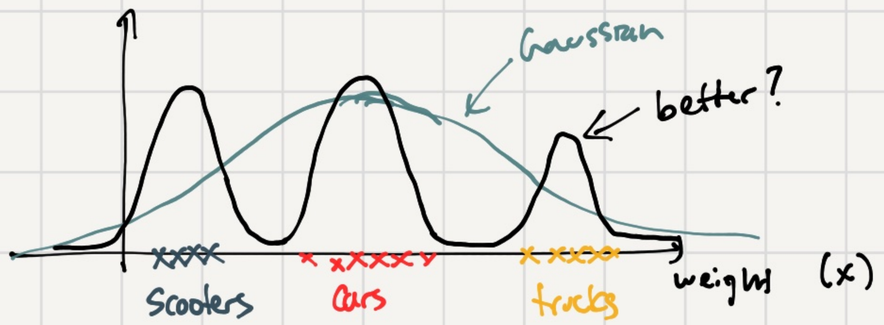
Lecture 4: Mixture Models & Clustering

CS5487 Lecture Notes (2024A)
Prof. Antoni B. Chan
Dept of Computer Science
City University of Hong Kong



So far, our probability models have one mode (peak).

e.g. Bridge sensor - measures weight of a vehicle.



If the data is more complicated, then a Gaussian can't tell the whole story.

Gaussian Mixture Model (GMM)

Two r.v.s = {

1) type of vehicle: $z \in \{\text{scooter}, \text{car}, \text{truck}\}$ = $\{1, 2, 3\}$ — "hidden state" z - we don't directly observe this

$p(z=j) = \pi_j$ (prior probability of a vehicle type crossing bridge)

$\sum_{j=1}^{3} \pi_j = 1$

2) weight of vehicle, conditioned on vehicle type — "observation" x

$p(x \mid z=j) = N(x \mid \mu_j, \sigma_j^2)$

each vehicle type has its own Gaussian (mean, variance).

}

Generative process:

1) Sample z (type of vehicle)
2) Sample $x \mid z$ (weight of specific type)

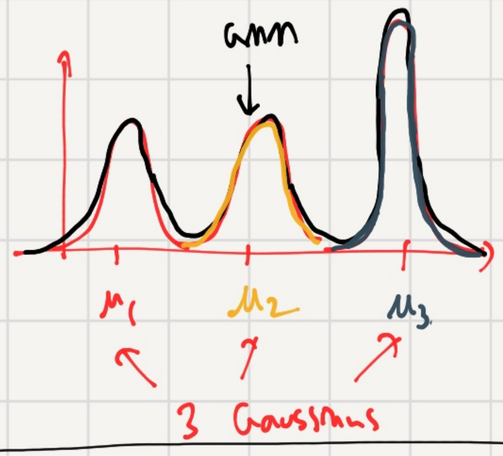
Note: we only observe x (weight). We don't see z.

Pdf of x

$$p(x) = \sum_z p(x, z) = \sum_{j=1}^{K} p(x \mid z=j)\, p(z=j)$$

$$\Rightarrow \boxed{p(x) = \sum_{j=1}^{K} \pi_j N(x \mid \mu_j, \sigma_j^2)} \quad \text{GMM}$$

π_j: component weight; $N(x \mid \mu_j, \sigma_j^2)$: mixture component (one peak)



"weighted sum of Gaussian pdfs"

(note: not the same as a weighted sum of Gaussian r.v.s)

Clustering w/ GMMs

Dataset $D = \{x_1, \ldots, x_n\}$. Assume K clusters

estimate a GMM from D:

1) Gaussian component → location μ_j, extent σ_j^2
2) component weight → probability of cluster π_j (# of samples)
3) Cluster assignment of each sample x_i (z_i)

Antoni's Hack

let: $z_i \in \{1, \dots, K\}$ = assignment of x_i to a cluster.

- treat z_i's as a parameter (optimize them)
- objective: maximize the joint LL of x, z

$$\hat{\theta} = \underset{\theta, z}{\text{argmax}} \sum_{i=1}^{n} \log p(x_i, z_i) = \underset{\theta, z}{\text{argmax}} \sum_i \log p(x_i | z_i) + \log p(z_i)$$

$$= \underset{\theta, z}{\text{argmax}} \sum_i \log N(x_i | \mu_{z_i}, \sigma^2_{z_i}) + \log \pi_{z_i}$$

Indicator variable trick

let $z_{ij} = \begin{cases} 1, & z_i = j \quad (x_i \text{ is assigned to cluster } j) \\ 0, & \text{otherwise} \end{cases}$

$$p(z_i) = \prod_{j=1}^{K} \pi_j^{z_{ij}} \Rightarrow \log p(z_i) = \sum_{j=1}^{K} z_{ij} \log \pi_j$$

$\pi_j^{z_{ij}}$: selects π_j when $z_{ij}=1$

$z_{ij} \log \pi_j$: select $\log \pi_j$ when $z_{ij}=1$

$$p(x_i | z_i) = \prod_{j=1}^{K} N(x_i | \mu_j, \sigma_j^2)^{z_{ij}} \Rightarrow \log p(x_i | z_i) = \sum_{j=1}^{K} z_{ij} \log N(x_i | \mu_j, \sigma_j^2)$$

$N(x_i | \mu_j, \sigma_j^2)^{z_{ij}}$: select the j^{th} Gaussian

$$\Rightarrow \underset{\theta, z}{\text{argmax}} \sum_{i=1}^{N} \sum_{j=1}^{K} z_{ij} \left(\log \pi_j + \log N(x_i | \mu_j, \sigma_j^2) \right)$$

θ: $\{\mu_j, \sigma_j^2, \pi_j\}$; z: z_i; objective J.

- variables depend on each other: alternating maximization
  1) optimize z with θ fixed.
  2) optimize θ with z fixed.

1) Find z_i w/ $\theta = \{\mu_j, \sigma_j^2, \pi_j\}$ fixed.

$$J = \sum_{i=1}^{N} \sum_{j=1}^{K} z_{ij} \left(\log \pi_j + \log N(x_i | \mu_j, \sigma_j^2) \right)$$

(i: sum over samples; j: z_{ij} selects one term)

each z_i is independent of other z_i's:

$$\underset{z_{ij}}{\text{argmax}} \sum_{j=1}^{K} z_{ij} \left(\log \pi_j + \log N(x_i | \mu_j, \sigma_j^2) \right)$$

Recall $z_{ij} \in \{0, 1\}$, only one $z_{ij} = 1$ for a given i.

⇒ since only one term in the sum can be selected w/ $z_{ij}=1$, then just pick the max term.

$$z_i = \underset{j}{\text{argmax}} \log \pi_j + \log N(x_i | \mu_j, \sigma_j^2)$$

select j w/ largest joint log-likelihood

2) Find $\{\mu, \sigma^2, \pi\}$ given z fixed.

$$\hat{\mu}_j = \underset{\mu_j}{\text{argmax}} \sum_{i=1}^{N} \sum_{j=1}^{K} z_{ij} \left(\log \pi_j + \log N(x_i | \mu_j, \sigma_j^2) \right)$$

$\log N(x_i | \mu_j, \sigma_j^2) = -\frac{1}{2\sigma_j^2}(x_i - \mu_j)^2 + \dots$

$$= \underset{\mu_j}{\text{argmax}} \sum_{i=1}^{n} z_{ij} \left(-\frac{1}{2\sigma_j^2} (x_i - \mu_j)^2 \right)$$

$$\frac{\partial}{\partial \mu_j} = \sum_i z_{ij} \left(-\frac{1}{2\sigma_j^2} \cdot 2 (x_i - \mu_j)(-1) \right) = 0$$

$$\sum_i z_{ij} x_i - \left[\sum_i z_{ij} \right] \mu_j = 0$$

$$\Rightarrow \hat{\mu}_j = \frac{1}{\sum_{i=1}^{N} z_{ij}} \sum_{i=1}^{N} z_{ij} x_i$$

$\sum_{i=1}^{N} z_{ij} x_i$: sum of samples assigned to j

$\sum_{i=1}^{N} z_{ij}$: # samples assigned to j

mean of samples assigned to j.

Similarly

$$\Rightarrow \hat{\sigma}_j^2 = \frac{\sum_{i=1}^{N} z_{ij}(x_i - \hat{\mu}_j)^2}{\sum_{i=1}^{N} z_{ij}}$$] variance of samples assigned to j

$$\Rightarrow \hat{\pi}_j = \frac{1}{N}\sum_{i=1}^{N} z_{ij}$$] fraction of samples assigned to j.

3) iterate (1) & (2) until convergence.

Notes:

- this 2-step iteration always maximizes our original objective $J \rightarrow$ converge (to a local max)
  - if $\sigma_j^2 = c$, $\pi_j = \frac{1}{K}$, then this is the K-means algorithm (Lloyd's Algorithm)
    1) $z_i = \arg\min_j (x_i - \mu_j^{(o)})^2$ ⇐
    2) $\mu_j = \frac{1}{\sum_i z_{ij}} \sum_i z_{ij} x_i$ ⇐
- need some initial values of z or θ. (solution depends on initialization)
- Not maximizing the actual data log-likelihood $\log p(D) = \sum_{i=1}^{N} \log p(x_i)$.

Expectation-Maximization (EM) Algorithm (Dempster, Laird, Rubin) 1977
73,000 citations.

MLE when there are hidden variables

X = observation r.v.

Z = hidden r.v.

joint LL: $p(X, Z) = p(X|Z)p(Z)$. data LL: $p(X) = \sum_Z p(x|Z)p(Z)$

Goal:

$$\hat{\theta} = \arg\max_\theta \log p(X) = \arg\max_\theta \log \sum_Z p(X|Z)p(Z)$$

Key observation: if we have both X, Z, then it's easy.

$$\arg\max \log p(X, Z) = \arg\max \log p(X|Z) + \log p(Z)$$

(same as step 2 in my algorithm)

- guess the values of Z in probabilistic way.
  1) use expected value of Z given our current model & data $\Rightarrow \hat{Z}$
  2) maximize $\log p(X, \hat{Z})$ to get a new model.
  3) repeat.

Formally

0) Select initial model $\hat{\theta}^{(old)}$

1) E-step: $Q(\theta; \hat{\theta}^{(old)}) = E_{Z|X,\hat{\theta}^{(old)}}[\log p(X, Z|\theta)]$

   - "Q function"
   - θ = param
   - $\hat{\theta}^{(old)}$ = current model (fixed)
   - $Z|X, \hat{\theta}^{(old)}$ = conditional expectation using current model.
   - $\log p(X, Z|\theta)$ = joint LL using param θ.

2) M-step: $\hat{\theta}^{(new)} = \arg\max_\theta Q(\theta; \hat{\theta}^{(old)})$

3) $\hat{\theta}^{(old)} \leftarrow \hat{\theta}^{(new)}$, iterate (1) & (2)

EM for GMMs

joint LL:

$$\log p(X, z \mid \theta) = \sum_{i=1}^{N} \sum_{j=1}^{K} z_{ij} \left(\log \pi_j + \log N(x_i \mid \mu_j, \sigma_j^2) \right)$$

1) **E-step:**

$$Q(\theta; \hat{\theta}^{(old)}) = E_{z \mid X, \hat{\theta}^{(old)}} \left[\log p(X, z \mid \theta) \right]$$

$$= \sum_{i=1}^{N} \sum_{j=1}^{K} \underbrace{E_{z \mid X, \hat{\theta}^{(old)}} \left[z_{ij} \right]}_{\hat{z}_{ij}} \left(\log \pi_j + \log N(x_i \mid \mu_j, \sigma_j^2) \right)$$

$$= \sum_{i} \sum_{j} \hat{z}_{ij} \left(\log \pi_j + \log N(x_i \mid \mu_j, \sigma_j^2) \right)$$

↑ similar to joint LL, but with $\hat{z}_{ij}$ instead of z_{ij}

$$\hat{z}_{ij} = E_{z \mid X, \hat{\theta}^{(old)}} \left[z_{ij} \right]$$

↓ expectation of an indicator variable (PS1-5)

$$= p(z_{ij} = 1 \mid X, \hat{\theta}^{(old)})$$

↓ Bayes' Rule

$$= \frac{p(X \mid z_{ij} = 1) \, p(z_{ij} = 1)}{p(X)}$$

↓ x_i's are independent, $p(X) = p(x_i) \, p(X_{-i})$

$$= \frac{p(x_i \mid z_{ij} = 1) \, p(X_{-i}) \, p(z_{ij} = 1)}{p(x_i) \, p(X_{-i})}$$

$$\hat{z}_{ij} = \frac{\hat{\pi}_j \, N(x_i \mid \hat{\mu}_j, \hat{\sigma}_j^2)}{\sum_k \hat{\pi}_k \, N(x_i \mid \hat{\mu}_k, \hat{\sigma}_k^2)}$$

← calculated with $\hat{\theta}^{(old)} = \{ \hat{\mu}_j, \hat{\sigma}_j^2, \hat{\pi}_j \}$

← "soft assignment" of x_i to cluster j.

$$\hat{z}_{ij} = p(z_i = j \mid x_i, \hat{\theta}^{(old)})$$

← posterior prob $[0, 1]$

2) **M-step:** same as step 2 but with $z_{ij} \to \hat{z}_{ij}$:

$$\hat{\mu}_j = \frac{1}{N_j} \sum_{i=1}^{N} \hat{z}_{ij} x_i$$

⇐ Sample mean w/ weights based on soft assignments.

$$N_j = \sum_{i=1}^{N} \hat{z}_{ij}$$

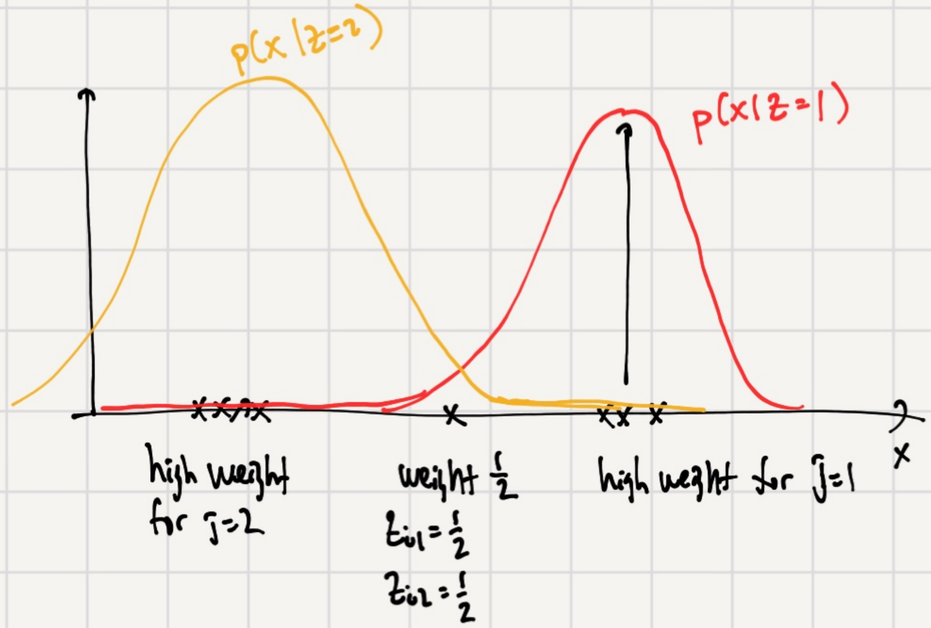
⇐ total weight of samples assigned to j.

$$\hat{\sigma}_j^2 = \frac{1}{N_j} \sum_{i=1}^{N} \hat{z}_{ij} (x_i - \hat{\mu}_j)^2$$

⇐ same

$$\hat{\pi}_j = \frac{N_j}{N}$$

⇐ fraction of weight assigned to cluster j.



Notes:

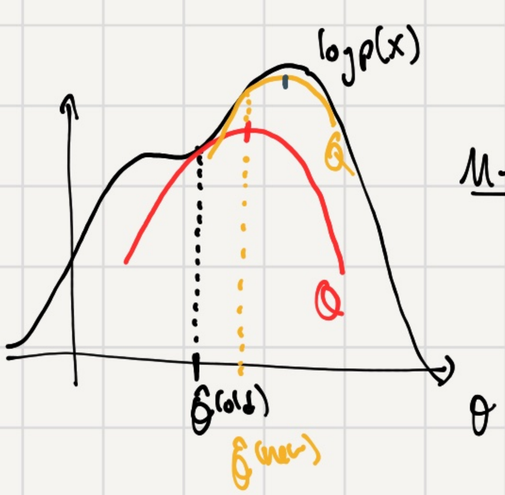
1) General: EM is a general framework for MLE
   on models w/ hidden variables

2) convergence: each iteration increases data LL ($p(x)$)
   → converges to a local max.
   (but it could be slow)

3) initialization: different init → different $\hat{\theta}$
   pick $\hat{\theta}$ w/ largest data LL.

4) interpretation:

   E-step: construct a lower bound function
   to $\log p(x) \geq Q(\theta; \hat{\theta}^{(old)}) + \text{const}$



   M-step: maximize the LB function

   (see handout)