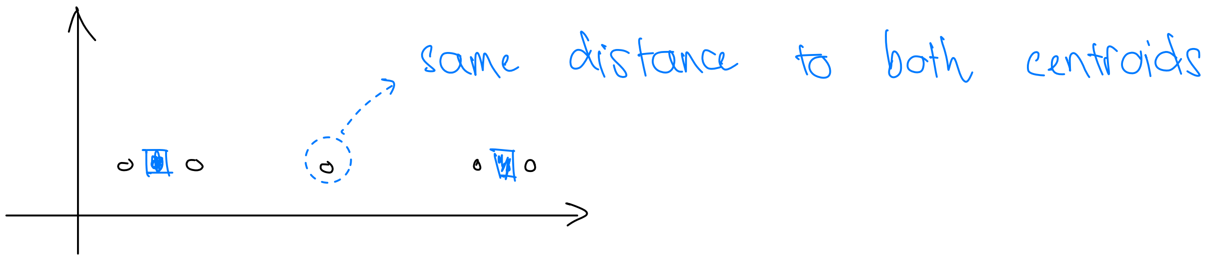
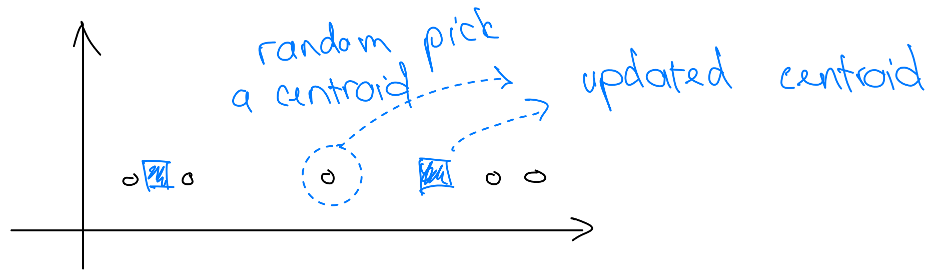
EXPECTATION MAXIMIZATION

Main idea:

- Expectation Minimization (EM) a more generalized version of K-means
- Problem with K-means:
  - Iteration t :



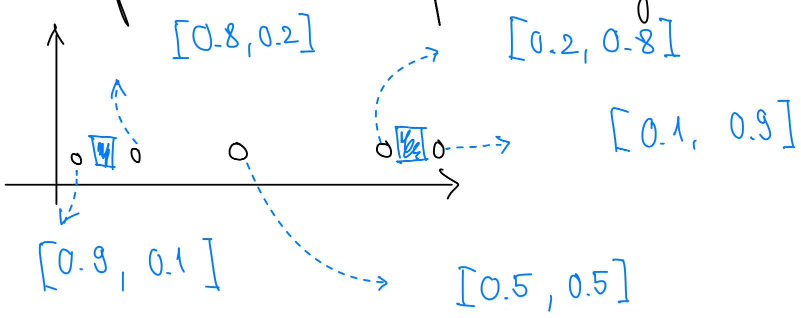
  - Iteration t+1 :



$\Rightarrow$ Not ideal. The centroids are not symmetric even though the data is symmetric

- How EM solve that problem:
  - In EM, each data point is a probability

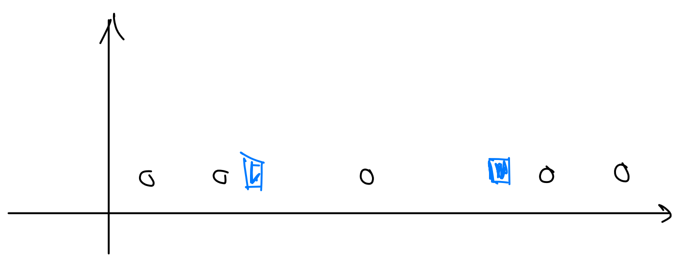
Iteration t :



where: $[\underbrace{0.9}_{P(z_i=1)}, \underbrace{0.1}_{P(z_i=2)}]$

  - Since z_i is a probability, μ will be more reprent the data

$$
\begin{cases}
\mu_1 = \dfrac{0.9x_1 + 0.8x_2 + 0.5x_3 + 0.2x_4 + 0.1x_5}{0.9 + 0.8 + 0.5 + 0.2 + 0.1} \\[2ex]
\mu_2 = \dfrac{0.1x_1 + 0.2x_2 + 0.5x_3 + 0.8x_4 + 0.9x_5}{0.1 + 0.2 + 0.5 + 0.8 + 0.9}
\end{cases}
$$



$\Rightarrow$ Look much better! (more symmetrical)

Expectation Maximization vs K-means

K-means (Deterministic approach)

- Assignment step:

For each data point x_i:

$$z_i = \underset{k=1,\dots,K}{\operatorname{argmin}} \|x_i - \mu_k\|^2$$

- Centroids step:

For each centroid:

$$\mu_k = \frac{\sum_{i \in S_k} x_i}{|S_k|} \quad (S_k = \{i : z_i = k\})$$

$$= \frac{\sum^n I(z_i = k)\, x_i}{\sum^n I(z_i = k)}$$

EM (Probabilistic approach)

- E-step (Evaluation)

For each data point x_i:

$$\begin{cases} p(z_i = 1) \\ \vdots \\ p(z_i = K) \end{cases}$$

$\Rightarrow \quad p(z_i = k)$ for each k

Example: sigmoid func distance

$$p(z_i = k) = \frac{\exp\left(-\frac{1}{\lambda}\|x_i - \mu_k\|^2\right)}{\sum_{k=1}^{K}\left(-\frac{1}{\lambda}\|x_i - \mu_k\|^2\right)}$$

- M-step: (Maximization)

For each centroid:

$$\mu_k = \frac{\sum^n p(z_i = k)\, x_i}{\sum^n p(z_i = k)}$$

How to choose correct λ ?

To find θ, we use this procedure (Probabilistic Approach to Clustering)

Probabilistic Approach to Clustering

- Assume some "hidden" joint distribution $p(x, z \mid \theta)$ that generates the data x and the labels z.
- The goal is to find that distribution by estimating θ
- To estimate θ, there are 2 scenarios:

  i) x and z are known (complete information). Estimate θ by maximizing the joint probability

  $$\max_\theta \sum_{i=1}^{n} \log p(x_i, z_i \mid \theta)$$

  ii) Only x are known (incomplete information). Estimate θ by maximizing the marginal probability

  $$\max_\theta \sum_{i=1}^{n} \log p(x \mid \theta)$$

next

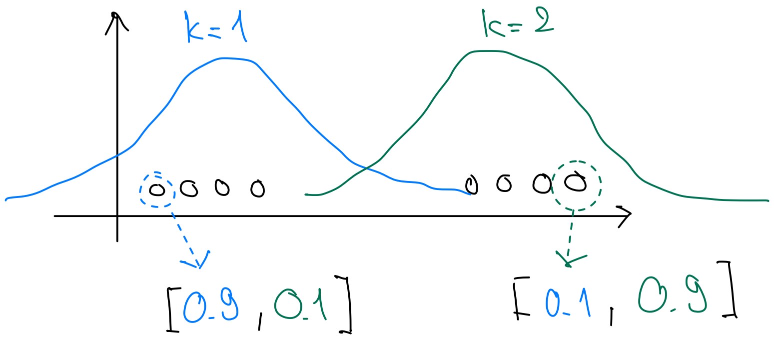
How does this help in clustering?

- Remember that the goal of clustering is to find the best possible assignment / label for each data point.
- Once we know the joint distribution $p(x, z \mid \theta)$ by estimating θ we can infer the label of each data point.

$$z \sim p(z \mid \underbrace{x}_{\text{observed}}, \underbrace{\theta}_{\text{estimated earlier}})$$

Example with Gaussian Distribution

- Visual:



- Write down the joint distribution $p(x, z)$

$$\underbrace{p(x, z)}_{\text{joint distribution}} = \underbrace{p(z)}_{\text{marginal distribution}} \cdot \underbrace{p(x \mid z)}_{\text{conditional distribution}}$$

For a specific k and data point x_i:

$$\underbrace{p(x_i, z = k)}_{\text{joint probability data and } z=k} = p(z=k) \cdot p(x_i \mid z = k)$$

$$= \underbrace{\pi_k}_{\text{probability } z=k} \cdot \underbrace{N(x_i \mid \mu_k, \sigma_k^2)}_{\text{probability seeing } x \text{ given } z=k \text{ (likelihood)}}$$

Trick:

Distribution is big if-else where each statement is a probability

The goal is to estimate θ, so that we can infer the labels

$$\boxed{\theta = \{\pi_k, \mu_k, \sigma_k^2\}_{k=1}^{K}}$$

The nitty griddy of how to estimate θ

Scenario 1: Complete Information

- We observed both data and labels (complete information)
- Estimate θ by maximizing joint probability:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{n} \log p(x, z \mid \theta)$$

- Detail calculations:

i) Write down log-joint probability

For a specific data point and specific k, the joint probability is:

$$p(x_i, z_i = k) = \pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2)$$

Then for all data points and all values of k:

$$\sum_{i=1}^{n} \sum_{k=1}^{K} I(z_i = k)\, p(x_i, z_i = k)$$
$$= \sum_{i=1}^{n} \sum_{k=1}^{K} I(z_i = k) \left[\pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2) \right]$$

Finally, arrive at log-joint probability:

$$\sum_{i=1}^{n} \sum_{k=1}^{K} I(z_i = k) \log\left[\pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2) \right]$$

ii) Estimate θ

- $\Theta = \{\pi_k, \mu_k, \sigma_k^2\}_{k=1}^{K}$
- Taking derivative w.r.t π_k, μ_k, σ_k, we arrive at these results:

$$\begin{cases} \pi_k = \dfrac{\sum_{i=1}^{n} I(z_i = k)}{n} \\[2ex] \mu_k = \dfrac{\sum_{i=1}^{n} I(z_i = k)\, x_i}{\sum_{i=1}^{n} I(z_i = k)} \\[2ex] \sigma_k = \operatorname{var}(\{x_i \mid z_i = k\}) \end{cases}$$

variance of all data points where $z = k$

Obviously, in practice we won't always have labels, that is what we will analyze next

Scenario 2: Incomplete information

- Can only observe data points, labels are hidden
- Estimating θ by maximizing marginal probability

$$\hat{\theta} = \arg\max_{\theta} \sum_{i=1}^{n} \log p(x_i \mid \theta)$$

- Detail calculations:

i) <u>Write down log-marginal probability</u>

For a specific data point:

$$p(x_i \mid \theta) = \sum_{k=1}^{K} p(x_i, z_i = k \mid \theta)$$

$$= \sum_{k=1}^{K} \pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2)$$

For all data points:

$$\sum_{i=1}^{n} p(x_i \mid \theta) = \sum_{i=1}^{n} \sum_{k=1}^{K} \pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2)$$

Add the log:

$$\sum_{i=1}^{n} \log p(x_i \mid \theta) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2) \right)$$

ii) <u>Estimate θ:</u>

- $\theta = \{ \pi_k, \mu_k, \sigma_k^2 \}$
- There is no closed form solution to maximizing the log marginal probability

Why? Lets analyze the marginal probability

$$\sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2) \right)$$

$$= \sum_{i=1}^{n} \log \Big(\underbrace{\sum_{k=1}^{K} f(\pi_k, x_i, \mu_k, \sigma_k^2)}_{\text{many distinct funtions}} \Big)$$

$\Leftrightarrow$ many mountains

- log (moutain) $\Leftrightarrow$ steps to that mountain
- log (moutains) $\Leftrightarrow$
  - steps to mountain 1
  - Steps to mountain 2
  - ...

- So no closed form method, luckily we can still use Expectation Maximization (EM) to estimate θ.

Expectation Maximization Algorithm

- Pseudo code:
  - Initially θ
  - For $t = 1, 2, 3, \ldots$

i) E - step (Evaluation):

Fill the hidden value z_i by drawn i.i.d. from $p(z \mid x, \theta)$

$$z_i \overset{i.i.d}{\sim} p(z \mid x, \theta)$$

For specific data point x_i and value k:

$$p(z_i = k \mid x_i, \theta^t) = \frac{p(x_i, z_i = k \mid \theta^t)}{\sum_{k=1}^{K} p(x_i, z_i = k \mid \theta^t)}$$

(posterior) conditional probability label of data point x_i is k

prior . likelihood $= p(z_i = k) \cdot p(x_i \mid z_i = k)$

marginal $p(x) = \sum^{K} p(x, z)$

ii) M - step (Maximization):

- Update θ by maximizing the expected log joint probability

For specific data point and value k:

$$\log p(x_i, z_i = k \mid \theta^t)$$

For all data points and all values of k:

$$\sum_{i=1}^{n} \sum_{k=1}^{K} p(z_i = k \mid x_i, \theta^t) \log p(x_i, z_i = k \mid \theta^t)$$

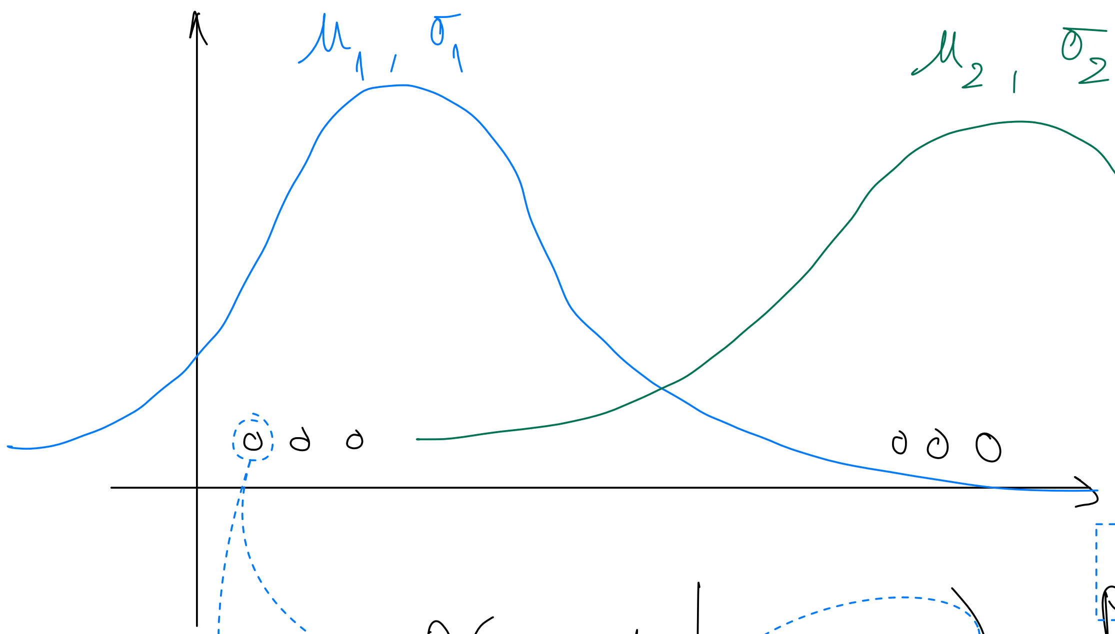
$I(z_i = k)$ in complete information case

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} p(z_i = k \mid x_i, \theta^t) \log \left[\pi_k \cdot N(x_i \mid \mu_k, \sigma_k^2) \right]$$

- Taking derivative w.r.t. π_k, μ_k, σ_k, we arrive at these results:

$$\begin{cases} \pi_k^{t+1} = \dfrac{\sum_{i=1}^{n} p(z_i = k \mid x_i, \theta^t)}{n} \\[2ex] \mu_k^{t+1} = \dfrac{\sum_{i=1}^{n} p(z_i = k \mid x_i, \theta^t) \cdot x_i}{\sum_{i=1}^{n} p(z_i = k \mid x_i, \theta^t)} \\[2ex] \sigma_k^{t+1} = \mathrm{Var}(\{x_i \mid z_i = k\}) \end{cases}$$

variance of data points with same label



Assignments:

$$\underbrace{P(z=1 \mid x_1, \mu_1, \sigma_1)}_{\gamma_{11}} = \frac{p(x_1, z=1 \mid \mu_1, \sigma_1)}{\sum^{k} p(x_1, z=k \mid \mu_1, \sigma_1)}$$

joint $p(x, z)$

marginal $p(x)$

observed

$$\underbrace{P(z=2 \mid x_1, \mu_2, \sigma_2)}_{\gamma_{12}}$$

$\vdots$

Do the same for other data points, then:
we have the distribution of z_i

$$z_i \sim p(z \mid x, \mu, \sigma)$$

joint liklihood

Centroids (μ, σ):

$$\{\mu, \sigma\}_{new} = \arg\max \sum E_{z_i \sim p(z \mid x, \mu, \sigma)} \left[\log p(x_i, z_i \mid \mu_1, \sigma_1) \right]$$

randomly drawn from