Gaussian Mixture Models

It is a model with a PDF of the form :

$$\sum_{j=1}^{J} \omega_j \mathcal{N}(\mu_j, \Sigma_j)$$

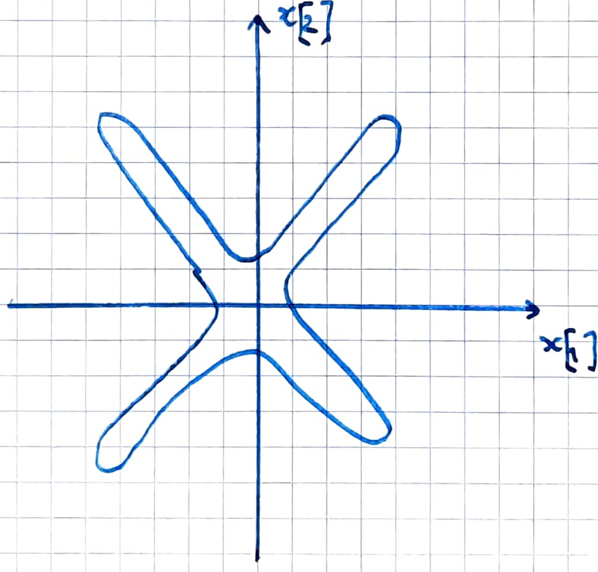
where $\mathcal{N}(x \mid \mu_j, \Sigma_j) = \sqrt{\frac{1}{(2\pi)^m \det \Sigma_j}} \exp\left(-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1} (x-\mu_j)\right)$

$x \in \mathbb{R}^m$

→ m is often 2 because we often work in $\mathbb{R}^2$ (2 Dimensions)

<u>First case : we want to build a model of this type</u>

Example : we want to create a density in $\mathbb{R}^2$ such that the iso-densities look like this :



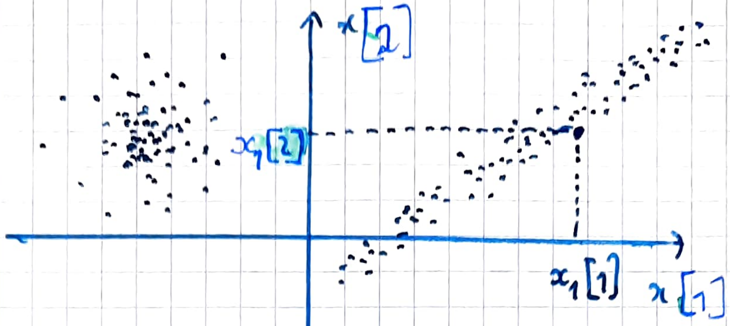
According to Gaussian-Multivariate-Covariance-Matrix / Gaussian.pdf we can define :

$$PDF(x) = \frac{1}{2}\mathcal{N}(0, \Sigma_1) + \frac{1}{2}\mathcal{N}(0, \Sigma_2)$$

with $\Sigma_1 = Q D_1 Q^T$ $\quad \Sigma_2 = Q D_2 Q^T$ $\quad Q = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$

$D_1 = \begin{pmatrix} 10 & 0 \\ 0 & 1 \end{pmatrix} \quad D_2 = \begin{pmatrix} 1 & 0 \\ 0 & 10 \end{pmatrix}$

Case two: we have a cloud of points and want to find the best GMM to model these points (real life)



These "observed" points are denoted by $x_1, x_2, \dots, x_n$. For each point x_i $(1 \leqslant i \leqslant n)$, $x_i[1]$ is his first coordinate and $x_i[2]$ the second.

Step 1: choose J

Algorithms won't do it for us

Now we have a model:

$$PDF(x) = \sum_{j=1}^{J} w_j \mathcal{N}(\mu_j, \Sigma_j)$$

We have $\underbrace{J}_{w_j} + \underbrace{mJ}_{\mu_j} + \underbrace{\frac{J(J-1)}{2}}_{\Sigma_j} = \frac{J^2}{2} + \left(m + \frac{1}{2}\right) J$ parameters to optimize

$m=2$ in the graphics above (because 2 Dimensional)

Step 2: Use EM algorithm:

At step t:

① Compute $\gamma_{ij}(t) = \dfrac{w_j^{(t)} \mathcal{N}(x_i \mid \mu_j^{(t)}, \Sigma_j^{(t)})}{\sum_{k=1}^{J} w_k^{(t)} \mathcal{N}(x_i \mid \mu_k^{(t)}, \Sigma_k^{(t)})}$

and $w_j^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} \gamma_{ij}^{(t)}$

② Compute $\mu_j^{(t+1)} = \dfrac{\sum_{i=1}^{n} \gamma_{ij}^{(t)} x_i}{\sum_{i=1}^{n} \gamma_{ij}^{(t)}}$ for all $j \in [\![1, J]\!]$

and $\Sigma_j^{(t+1)} = \dfrac{\sum_{i=1}^{n} \gamma_{ij}^{(t)} (x_i - \mu_j^{(t+1)})(x_i - \mu_j^{(t+1)})^T}{\sum_{i=1}^{n} \gamma_{ij}^{(t)}}$ for all $j \in [\![1, J]\!]$

This algorithm converges to a local maximum likelihood.

See the folder GMM - ~~EM~~ Proof because the proof is particularly interesting in itself, with the use of:

- Jensen inequality
- Deriving the Lagrangian
- Kullback-Leibler divergence
- Some nice differential calculations
- Conditional probabilities

Without proving anything, note that the EM algorithm is intuitive because:

- $$\gamma_{ij}^{(t)} = \frac{\mathbb{P}(Z_i = j)\,\mathcal{N}^{(t)}(X_i \mid Z_i = j)}{\mathcal{N}^{(t)}(X_i)} = \overset{\mathbb{P}}{f}(Z_i = j \mid X_i = x_i)$$

  is the probability of belonging to the j-th Gaussian given x_i

- This probability is used to weight each new average parameter

  $$\mu_j^{(t+1)} = \frac{\sum_i \gamma_{ij}^{(t)} x_i}{\sum_i \gamma_{ij}^{(t)}}$$

- Similarly for the expression of the covariance.