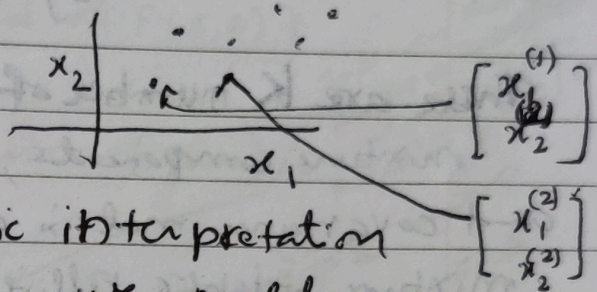
Gaussian Mixture Models (GMM)

We studied K-means clustering to group the given data $\underline{x} = \{ \underline{x}^{(1)}, \underline{x}^{(2)}, \ldots \underline{x}^{(m)} \}$, where $\underline{x}^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_n^{(i)} \end{bmatrix}$ (earlier we may have used $\underline{x}$ or $\underline{x}^{(i)}$) → Notation changed. Sorry

$\underline{x}^{(i)}$ is the i^{th} data vector or feature vector. Note that the entire data can be considered as → consisting of n dimensional rvs.

$$X_1 : x_1^{(1)} \; x_1^{(2)} \cdots x_1^{(m)}$$
$$X_2 : x_2^{(1)} \; x_2^{(2)} \cdots x_2^{(m)}$$
$$\vdots$$
$$X_n = x_n^{(1)} \; x_n^{(2)} \cdots x_n^{(m)}$$

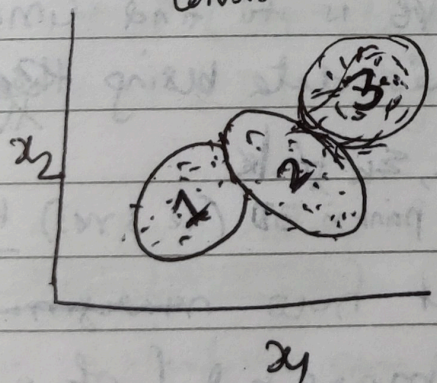
If we consider 2D then each data point can be represented in a 2D plot-



Can we have a probabilistic interpretation for clustering? i.e. how can we model $p(\underline{x}) = p_{x_1 \cdots x_n}(x_1, x_2 \cdots x_n)$ to reflect our intuition that the points (that we assign) remain closer to their cluster centres (centroids)

↓ joint density fn of $x_1 \cdots x_n$

Consider the plot below



We can see that the given points seem to form 3 clusters. We cannot model such a data by a simple distribution such as Gaussian (with some mean & some variance)

One can think of modeling each region with a distinct distribution. We may think of mixtures of Gaussians (GMMs) to model such a data.

GMM are ... in speech/speaker recognition.

How can we do it? We have to do it using unlabelled data x (unsupervised)

Since x is considered as mixture of Gaussians GMM has the following pdf for x

$$p(x) = p_{\underbrace{x_1 \dots x_n}_{x}}(x_1, x_2, \dots x_n) = \sum_{k=1}^{K} \omega_k \mathcal{N}(x / \mu_k, \Sigma_k)$$

If $K=3$ $\quad p(x) = \omega_1 \mathcal{N}(x/\mu_1, \Sigma_1) + \omega_2 \mathcal{N}(x/\mu_2, \Sigma_2) + \omega_3 \mathcal{N}(x/\mu_3, \Sigma_3)$

There are K number of Gaussian distributions called mixture components, μ_k and Σ_k represent mean and covariance matrix of k^{th} component ω_k represent mixture weights telling us about how much each component contribute to final GMM represented by $p(x)$

One has to have $\omega_k > 0$, $\forall k$ and $\sum_{k=1}^{K} \omega_k = 1$ to ensure that $p(x)$ is a valid pdf.

Now given x our objective is to find GMM parameters to represent this data using the parameters $\omega_k, \mu_k, \Sigma_k \; \forall k$

Example if $K=3$ & each data point is 3D (ie. 3 rvs) then we have to find $\omega_1, \omega_2, \omega_3$ and μ_1, μ_2, μ_3 all are 3×1 $\Sigma_1, \Sigma_2, \Sigma_3$ are 3×3.

To do this let us look at how GMM can be expressed as the marginal (of a joint pdf)

We know that $p_{X,Z}(x, z) = p_Z(z)\, p_{X/Z}(x|z)$ $\quad$ x is a random vector & has multivariate Pdf

where Z is a discrete rv taking values $k = 1$ to K.

Denote $\omega_k = P_Z(Z=k)$

since we are dealing with GMM, assume conditional distributions as Gaussians

i.e. $p(x/z=k) = \mathcal{N}(x/\mu_k, \Sigma_k)$

We can then write

$$p(x) = \sum_{\forall z} p(z)\, p(x/z) \quad \text{(data point } x \in \text{to)}$$

$$= \sum_{k=1}^{K} \omega_k \mathcal{N}(x/\mu_k, \Sigma_k) \quad \text{which is GMM of } x$$

[See that $p(x)$ is marginal of $p(x,z)$]

As an example (from the previous figure)

Indicates that drawing a point from Gaussian with μ_1, Σ_1 has probability $p(z/z=1)$

$$p(x/z=1) = \mathcal{N}(x/\mu_1, \Sigma_1)$$

$$p(x/z=2) = \mathcal{N}(x/\mu_2, \Sigma_2)$$

$$p(x/z=3) = \mathcal{N}(x/\mu_3, \Sigma_3)$$

Then $p(x) = p(\text{region } 1)\, \mathcal{N}(x/\mu_1, \Sigma_1) + p(\text{region } 2)\, \mathcal{N}(x/\mu_2, \Sigma_2) + p(\text{region } 3)\, \mathcal{N}(x/\mu_3, \Sigma_3)$

Also note that one of the value of z in $z^{(n)}$ which can take for a

$z^{(1)} = 2$ means first data point $\in$ cluster 2

$z^{(n)} = 3$ means first data point $\in$ cluster 3

first data point is $x^{(1)}$ ↓ n) vector

Estimation of parameters of GMM $\theta = \{\omega_k, \mu_k, \Sigma_k\}_{k=1}^{K}$ with complete data.

Let us consider unrealistic case where we have labels z

Define $\mathbb{D} = \{x^{(n)}\}_{n=1}^{m}$ or $\mathbb{D}' = \{x^{(n)}, z^{(n)}\}_{n=1}^{m}$

$\mathbb{D}$ → incomplete data; $\mathbb{D}'$ → Complete data

$x^{(1)}$ represents first point

$x^{(2)}$ " 2nd point

$x^{(m)}$: mth data point.

$z^{(1)} = 1$ means first data pt $\in$ region (cluster)

If D' i.e., complete data is given ML estimation of θ can be obtained by maximizing the $P(x^{(n)}, z^{(n)})$ considering all data points

i.e. $\prod_{n=1}^{m} P(x^{(n)}, z^{(n)})$ or θ can be estimated by maximize

$$\max_{\theta} \log \left[\prod_{n=1}^{m} P(x^{(n)}, z^{(n)}) \right]$$

$$= \max_{\theta} \sum_{n=1}^{m} \log P(x^{(n)}, z^{(n)})$$

$$= \max_{\theta} \sum_{n=1}^{m} \log P(z^{(n)}) P(x^{(n)}/z^{(n)})$$

$$= \max_{\theta} \sum_{k=1}^{K} \sum_{n: z^{(n)}=k} \log P(z^{(n)}) P(x^{(n)}/z^{(n)}) \quad \text{---} \quad (2)$$

i.e. group the data by it $z^{(n)}$ values (Ex: If there are 30 data points and 10 each $\in$ to each cluster with number of clusters (regions) = K = 3, then $\log P(z^{(n)}) P(x^{(n)}/z^{(n)})$ appears 10 times (for 10 data points each). There are total of 30 terms.

Let $\gamma^{nk} \in \{0,1\}$ be a binary variable. If $z^{(n)} = k$ i.e., if n^{th} data point belongs to cluster k, $\gamma^{nk} = 1$ else 0. Note that $\gamma^{nk} = P(z^{(n)}=k)$ i.e., Chance of assigning n^{th} feature vector $x^{(n)}$ to a cluster

$\therefore$ We can write

$$\sum_{n} \log P(x^{(n)}, z^{(n)}) = \sum_{k} \sum_{n} \gamma^{nk} \log \{ P(z^{(n)}=k) \times P(x^{(n)}/z^{(n)}=k) \}$$

$$= \sum_k \sum_n \gamma^{nk} \left[\log \omega_k + \log \mathcal{N}(x^{(n)} / \mu_k, \Sigma_k) \right]$$
$$z^{(n)} = k$$

$$= \sum_{k=1}^{K} \sum_{n=1}^{m} \gamma^{nk} \log \omega_k + \sum_{k=1}^{K} \left\{ \sum_{n=1}^{m} \gamma^{nk} \log \mathcal{N}(x^{(n)} / \mu_k, \Sigma_k) \right\} \quad (1)$$

One can now write the pdf for $\mathcal{N}(x^{(n)} / \mu_k, \Sigma_k)$ and after differentiating (wrt parameters) and equating to 0, we get

[for maximizing $\prod_{n=1}^{m} P(x^{(n)}, z^{(n)})$ or for maximizing $\sum \log P(x^{(n)}, z^{(n)})$]

$$\omega_k = \frac{\sum_n \gamma^{nk}}{\sum_k \sum_n \gamma^{nk}} \quad \Rightarrow \quad \mu_k = \frac{\sum_n \gamma^{nk} x^{(n)}}{\sum_n \gamma^{nk}}$$

ω_k: scalar; this is known.

μ_k: a vector (n dimensional)

$$\Sigma_k = \frac{1}{\sum_n \gamma^{nk}} \sum_n \gamma^{nk} (x^{(n)} - \mu_k)(x^{(n)} - \mu_k)^T$$

Σ_k: $n \times n$ matrix

$(n \times 1)$ $(1 \times n)$ $\rightarrow n \times n$

$\omega_k = P(z^{(n)} = k)$ represents fraction of total number of data points which are assigned to each cluster k $\rightarrow$ ie probability of each cluster ($\omega_1 + \omega_2 + \dots \omega_K = 1$)

Note that $\sum_{k}^{K} \sum_{n=1}^{m} \gamma^{nk} = m$

$\omega_1 = \frac{m_1}{m}$, $\omega_2 = \frac{m_2}{m}$

$\omega_3 = \frac{m_3}{m}$ ...

So $\omega_1 + \omega_2 + \omega_3 + \dots = 1$

because $m_1 + m_2 + m_3 + \dots = m$

γ^{nk} is given the feature $x^{(n)}$. In this case γ^{nk} is assumed to be known ie

μ_k Vector represents mean of all data points which are assigned to cluster k (i.e. whose $z^{(n)} = k$)

Ex Suppose $K = 3$ & each data point is of dimension 2 (i.e., $n = 2$)

$X_1 = x_1^{(1)} \quad \ldots \quad x_1^{(m)}$

$X_2 = x_2^{(1)} \quad \ldots \quad x_2^{(m)}$

nth data point is $x^{(n)} = \begin{bmatrix} x_1^{(n)} \\ x_2^{(n)} \end{bmatrix}_{2\times 1}$

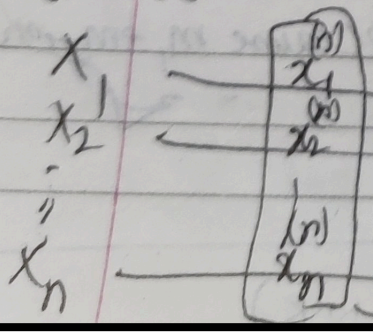
If $m = 30$ & if first 10 points of each r.v. (X_1 or X_2) $\in$ cluster 1

i.e., $\omega_1 = 10/30 = 1/3$, $M_k = \begin{bmatrix} m_{x1} \\ m_{x2} \end{bmatrix} = \begin{bmatrix} \text{average of 10} \\ \text{"} \end{bmatrix}$

Covariance matrix $\Sigma_1 = \Sigma_k$ will be of size 2×2 computed by using the first 10 points of X_1 & X_2

Important Point: The above clustering method is unrealistic since we have assumed which data points $\in$ to which cluster. For example for the above example we know that first 10 points (10 features) $\in$ cluster 1. In practice which data point $\in$ to which cluster is not known at all. So the problem of automatic clustering becomes quite difficult!

Let us now consider estimating θ when we do not know $z^{(n)}$ (i.e., which data pts $\in$ to which cluster). We are simply given the data



A method called EM (expectation maximization) is often used when we do not know $Z^{(n)}$ ie. when we are given incomplete data (D) & not D'

Parameter estimation for GMMs: Incomplete data (Use of EM algorithm)

When $Z^{(n)}$ is not given, one can guess γ^{nk} by using its posterior probability ie.

$$P(Z_k^{(n)}=1 / x^{(n)}) = \frac{P(x^{(n)} / z^{(n)}=k)\, P(z^{(n)}=k)}{P(x^{(n)})} = \gamma^{nk}$$

ie. if we do not know that nth data point belongs to (∈) cluster k, one can guess it by writing the above probability expression ie. the chance of assigning nth data point to kth cluster given that point is equal to what is written on the RHS of the above expression. In the previous case we assumed that this was known. That is if $\gamma^{nk}=1$ then $P(z^{(n)}=k / x^{(n)})=1$ else it is zero. Note that this is not practical since we do not know which data point belongs to which cluster. It may ∈ to any one of the K clusters. Hence obtaining the GMM parameters ie. $\omega_k, k=1\ldots K$ & μ_k & Σ_k, $k=1\ldots K$ is not easy. If it is unrealistic case where $z^{(n)}$ is known the problem can be solved easily (since we have the complete data). The solution is already obtained (see the expressions for $\omega_k, \mu_k, \Sigma_k$)

(4) continued

Let us now obtain the same solution with the incomplete data ($z^{(n)}$ not known) by guessing & arriving at solution using EM.

We know that

$$P(z^{(n)}=k / x^{(n)}) = \frac{P(x^{(n)}/z^{(n)}=k)\, P(z^{(n)}=k)}{\sum_{k'=1}^{K} P(x^{(n)}/z^{(n)}=k')\, P(z^{(n)}=k')} = \gamma^{nk}$$

Note that we cannot write $x^{(n)}=k$ since $x^{(n)}$ is a vector. We have to use another random variable say z to indicate that a particular data point belongs to a cluster k

This is $P(x^{(n)})$

Now, how do you compute the posterior? We need to know the parameters which are not known. Let start with some parameters so that we can guess the posterior.

We have $\gamma^{nk} = P(z^{(n)}=k / x^{(n)})$. We know that

$\gamma^{n\cdot 1} + \gamma^{n\cdot 2} + \gamma^{n\cdot 3} = P(z^{(n)}=1/x^{(n)}) + P(z^{(n)}=2/x^{(n)}) + P(z^{(n)}=3/x^{(n)})$

$= 1$ (Assuming $K=3$) for all $n=1$ to m

We will assume γ^{nk} for each data point (if there are say 30 data points, the total number of γ^{nk} values are $(30\times 3)=90$

With these values expectation of assigning n^{th} data point to each cluster can be written (average probability) as

$$P(z^{(n)}=1/x^{(n)}) \log P(x^{(n)}/z_0^{(n)}=1, \mu_1, \Sigma_1) + P(z^{(n)}=2/x^{(n)}) \log P(x^{(n)}/z_0^{(n)}=2, \mu_2, \Sigma_2) + P(z^{(n)}=3/x^{(n)}) \log P(x^{(n)}/z_0^{(n)}=3, \mu_3, \Sigma_3)$$

log is used here for convenience

i.e. it represents on an average what is the chance that it belongs to cluster 1, which itself depends on $\mu_1, \Sigma_1 \cdots \mu_3, \Sigma_3$.

$E(f(x)) = \sum g(x) \times P(x=k)$

$P(z/x)$ is a p.d.f. of $P(x/z\ldots)$

Note that (E) is same as the 2nd term of (1) & hence θ can be obtained using (1)

If we do this for all data points, we can write the total probability as

$$Q(\theta_{old}, \theta) = \sum_{k=1}^{K} \sum_{n=1}^{m} P(z^{(n)}=k / x^{(n)}, \theta_{old}) \log P(x^{(n)} / z^{(n)}, \theta) \quad \text{— (E)}$$

Here (E) Writing Q is called expectation step

where θ i.e. $\mu_1 \Sigma_1 \ldots \mu_K \Sigma_K$ are unknown

We can then maximize Q by choosing appropriate θ which is done iteratively Hence the name EM. (i.e., we arrive at the final θ by using expectation step and maximization step. That is we obtain the parameters by maximizing (over θ) the expectation of log(x/z) taken w.r.t likelihood $P(z/x)$

Page (5)

$E(Y) = \sum y_i P(y_i)$ — posterior — likelihood in EM.

Note that finally after convergence we arrive at the solution as (1) i.e. we succeed in making $\gamma^{nk} \in [0,1]$ (obtaining θ) i.e. by making γ^{nk} as 1 for those data points belong to clusters 1, 2, ... K iteratively.

(i.e. If in 30 data points first 10 belong to cluster 1 then $\gamma^{n,1} = 1$ for $n = 1, 2, \ldots 10$ and $\gamma^{n,1} = 0$ for $n = 11$ to 30, (11)ly for for other clusters)

How we perform EM? Once we assume γ^{nk} for all n, k we already know the expressions for getting w_k, μ_k, Σ_k for all k (See (1)). Use (1) to get the updated w_k, μ_k, Σ_k. Then use these to get updated γ^{nk} & form E step & maximize w.r.t θ. Repeat until we get convergence

~~EM to get $\theta = \omega_k, \theta_k = \omega_k$~~

EM to get $\omega_k, \Sigma_k, \mu_k$ iteratively

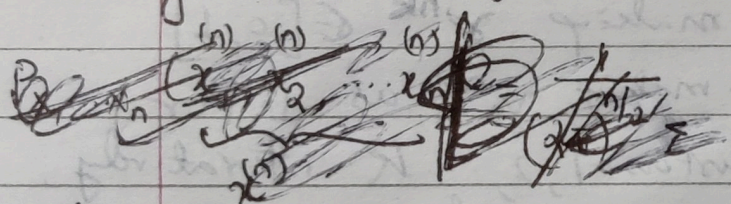
E Step. (Expectation step)

Write E step as (E) given in previous page. Here $\gamma^{nk} = P(z^{(n)}=k / x^{(n)}, \theta_{old})$ are assumed. θ ~~correspond to~~ are unknown.

M Step (Maximization step)

Maximize (E) for all θ. This leads to expressions given in (1) on page (3). Using these expressions we get updated parameters $\theta = \omega_k, \mu_k, \Sigma_k$

Go back to E step after computing updated γ^{nk} using expression given in page (4). Use



Here $\mu_k = \begin{bmatrix} \mu_k^{(1)} \\ \mu_k^{(2)} \end{bmatrix}_{2\times 1}$

$$P(x^{(i)} / z^{(i)}=k) = P_{x_1, x_2 \ldots x_n}(x_1^{(i)}, x_2^{(i)} \ldots x_n^{(i)} / z^{(i)}=k)$$

$$= \frac{1}{(2\pi)^{n/2} |\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2} \underset{1\times n}{(x^{(i)} - \mu_k)^T} \underset{n\times n}{[\Sigma_k]^{-1}} \underset{n\times 1}{(x^{(i)} - \mu_k)} \right)$$

If $n=2$, (two r.v.s x_1 & x_2 i.e., 2D feature vectors)

$$\rightarrow P(x^{(i)} / z^{(i)}=k) = P_{x_1 x_2}(x_1^{(i)}, x_2^{(i)} / z^{(i)}=k)$$

$$= \frac{1}{(2\pi) |\Sigma_k|^{1/2}} \times \exp\left(-\frac{1}{2} \underset{1\times 2}{([x_1^{(i)} - \mu_k^{(1)}] [x_2^{(i)} - \mu_k^{(2)}]} \underset{2\times 2}{\Sigma_k^{-1}} \right.$$