EM Motivation

- want to do MLE of mixture of Gaussian parameters
- But this is hard, because of the summation in the mixture of Gaussian equation (can't take the log of a sum).
- If we knew which point contribute to which Gaussian component, the problem would be a lot easier (we can rewrite so that the summation goes away)
- So... let's guess which point goes with which component, and proceed with the estimation.
- We were unlikely to guess right the first time, but based on our initial estimation of parameters, we can now make a better guess at pairing points with components.
- Iterate
- This is the basic idea underlying the EM algorithm.

EM Algorithm

What makes this estimation problem hard?
1) It is a mixture, so log-likelihood is messy
   \[ \ln p(X|\theta, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left( \sum_{k=1}^{K} \tau_k \phi(X_n; \mu_k, \Sigma_k) \right) \]
2) We don't directly see what the underlying process is

Suppose some oracle told us which point comes from which Gaussian.
How? By providing a "latent" variable \( z_{nk} \) which is 1 if point \( n \) comes from the \( k \)th component Gaussian, and 0 otherwise (a \( 1 \) of \( K \) representation)

This lets us recover the underlying generating process decomposition:

And we can easily estimate each Gaussian, along with the mixture weights!
EM Algorithm

Remember that this was a problem:

$$\ln p(x | \mu, \Sigma) = \sum_{n=1}^{N} \ln p(x_n | \mu, \Sigma)$$

Again, if an oracle gave us the values of the latent variables (component that generated each point) we could work with the complete log likelihood

$$p(X, Z | \mu, \Sigma) = \prod_{n=1}^{N} \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)$$

$$\ln p(X, Z | \mu, \Sigma) = \sum_{n=1}^{N} \sum_{k=1}^{K} \ln \pi_k + \ln \mathcal{N}(x_n | \mu_k, \Sigma_k)$$

This is thus equivalent to

$$\sum_{n \text{ for which } z_n = k} \ln \pi_k + \ln \mathcal{N}(x_n | \mu_k, \Sigma_k)$$

$$+ \sum_{n \text{ for which } z_n = 1} \ln \pi_1 + \ln \mathcal{N}(x_n | \mu_1, \Sigma_1)$$

$$+ \cdots + \sum_{n \text{ for which } z_n = K} \ln \pi_K + \ln \mathcal{N}(x_n | \mu_K, \Sigma_K)$$

Latent Variable View

$$\ln p(x | \mu, \Sigma) = \sum_{n=1}^{N} \ln \pi_n + \ln \mathcal{N}(x_n | \mu_n, \Sigma_n)$$

note: for a given n, there are k of these latent variables, and only ONE of them is 1 (all the rest are 0)

These are coupled because the mixing weights all sum to 1, but it is no big deal to solve
EM Algorithm

Unfortunately, oracle’s don’t exist (or if they do, they don’t want to talk to us)
So we don’t know values of the the \( z_{nk} \) variables
What EM proposes to do:
1) compute \( p(Z|X,\theta) \), the posterior distribution over \( z_{nk} \)
given our current best guess at the values of \( \theta \)
2) compute the expected value of the log likelihood \( \ln(p(X,Z|\theta)) \)
with respect to the distribution \( p(Z|X,\theta) \)
3) find \( \theta_{new} \) that maximizes that function.
   This is our new best guess at the values of \( \theta \).
4) iterate...

Insight

So now, after replacing the binary latent variables with their continuous
expected values:
all points contribute to the estimation of all components
each point has unit mass to contribute, but splits it across the \( K \) components
the amount of weight a point contributes to a component is proportional to
the relative likelihood that the point was generated by that component

Latent Variable View (with an oracle)

these are coupled because the mixing weights
all sum to 1, but it is no big deal to solve

Latent Variable View (with EM, \( \gamma^i_{nk} \) a constant \( \gamma^i_{nk} \) at iteration \( i \) )

these are coupled because the mixing weights
all sum to 1, but it is no big deal to solve

EM Algorithm for GMM

**E**

\[
\gamma_j(x_n) = \frac{\pi_j \mathcal{N}(x_n; \mu_j, \Sigma_j)}{\sum_k \pi_k \mathcal{N}(x_n; \mu_k, \Sigma_k)} \quad \text{ownership weights}
\]

**M**

\[
\mu_j = \frac{\sum_{n=1}^{N} \gamma_j(x_n) x_n}{\sum_{n=1}^{N} \gamma_j(x_n)} \quad \Sigma_j = \frac{\sum_{n=1}^{N} \gamma_j(x_n) (x_n - \mu_j)(x_n - \mu_j)^T}{\sum_{n=1}^{N} \gamma_j(x_n)} \quad \text{means}
\]

\[
\pi_j = \frac{1}{N} \sum_{n=1}^{N} \gamma_j(x_n) \quad \text{mixing probabilities}
\]
Gaussian Mixture Example:
Start

After first iteration

After 2nd iteration

After 3rd iteration

After 4th iteration

After 5th iteration
After 6th iteration

Recall: Labeled vs Unlabeled Data

- Labeled data: Easy to estimate params (do each color separately)
- Unlabeled data: Hard to estimate params (we need to assign colors)

After 20th iteration

EM produces a “Soft” labeling

- Each point makes a weighted contribution to the estimation of ALL components

General EM

The General EM Algorithm

Given a joint distribution $p(X, Z|\theta)$ over observed variables $X$ and latent variables $Z$, governed by parameters $\theta$, the goal is to maximize the likelihood function $p(X|\theta)$ with respect to $\theta$.

1. Choose an initial setting for the parameters $\theta^{(1)}$.
2. E step: Evaluate $p(Z|X, \theta^{(k)})$.
   
   Evaluate $q(\theta|\theta^{(k)}) = \sum_{\theta} q(\theta|X, Z, \theta^{(k)}) p(X, Z|\theta^{(k)})$.

3. M step: Estimate $\theta^{(k+1)}$ given by
   
   $\theta^{(k+1)} = \arg\max_{\theta} \mathcal{Q}(\theta|\theta^{(k)})$.

4. Check for convergence of either the log likelihood or the parameter values. If the convergence criterion is not satisfied, then let $\theta^{(k)} = \theta^{(k+1)}$.

Intuitive Explanation

(in terms of function maximization and lower bounds)
Intuitive Explanation

Why does this work?
By construction, \( b_1(x_1) = f(x_1) \)
\( b_1(x_2) \geq b_1(x_1) \) [it is a maximum]
\( f(x_2) \geq b_1(x_2) \) [\( b_1 \) is a lower bound]
so, it is guaranteed that
\( f(x_2) \geq f(x_1) \)
and in general, at each iteration
\( f(x_{\text{new}}) \geq f(x_{\text{old}}) \)
If \( f(x) \) is bounded above, then process should converge to a (local) maximum

More Rigorous Proof

We will use Jensen’s inequality for convex functions (see, for example, Bishop, PRML, p 96)

\[
\ln \sum_{k} \alpha_k = \ln \sum_{k} \frac{\alpha_k}{\lambda_k} \geq \sum_{k} \lambda_k \ln \left( \frac{\alpha_k}{\lambda_k} \right)
\]

where \( \lambda_k > 0 \) and \( \sum \lambda_k = 1 \), for any set of points \( \{x_i\} \).

With some manipulation, and reversing the inequality because \( \log \) is a concave rather than convex function...

Proof that EM works

\[
\ln p(X|\theta) = \ln \sum_{Z} P(X,Z|\theta) \text{ definition of probability. Now use Jensen’s inequality...}
\geq \sum_{Z} p(Z|X,\theta^{\text{old}}) \ln \left( \frac{p(X,Z|\theta)}{p(Z|X,\theta^{\text{old}})} \right)
\]

lower bound \( b(x) \) in our earlier picture.
Bishop calls this L(q,\theta)
Proof that EM works

Note, that when $\theta = \theta_{old}$

$$
\sum_Z p(Z|X, \theta_{old}) \ln \frac{p(X, Z|\theta_{old})}{p(Z|X, \theta_{old})}
$$

If we expand

$p(X, Z|\theta_{old}) = p(Z|X, \theta_{old}) p(X|\theta_{old})$

this equation $L(q; \theta_{old})$ becomes just

$\ln p(X|\theta_{old})$

Showing that our lower bound "touched" the function $\ln p(\theta)$ at the current estimate $\theta_{old}$ (as promised by our earlier picture!)

Proof that EM works

So...

and we have increased our log likelihood.

Therefore, finding $\text{argmax}_\theta L(\theta, \theta_{old})$ is a good thing to do.

But wait, there's more...

Proof that EM works

$$
L(\theta, \theta_{old}) = \sum_Z p(Z|X, \theta_{old}) \ln \frac{p(X, Z|\theta_{old})}{p(Z|X, \theta_{old})}
$$

This is the expected value $Q(\theta; \theta_{old})$ computed in the E-step of EM!!!

$$
= \sum_Z p(Z|X, \theta_{old}) \ln p(X|\theta_{old}) - \sum_Z p(Z|X, \theta_{old}) \ln p(Z|X, \theta_{old})
$$

this doesn't depend on $\theta$; ignore it.

$\arg\max_\theta L(\theta, \theta_{old}) = \arg\max_\theta Q(\theta, \theta_{old})$

therefore, EM is optimizing the right thing!