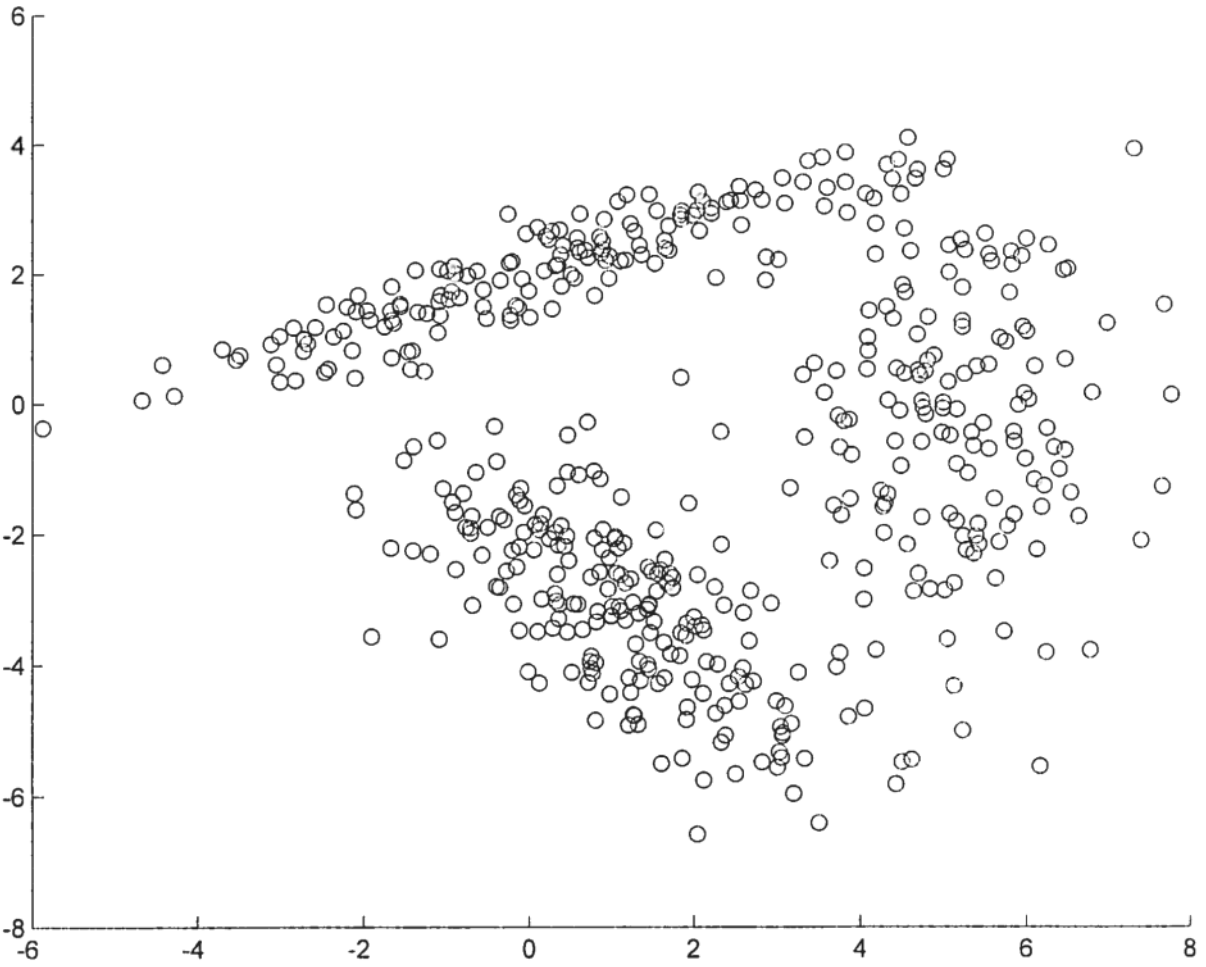
GAUSSIAN MIXTURE MODELS AND THE EM ALGORITHM

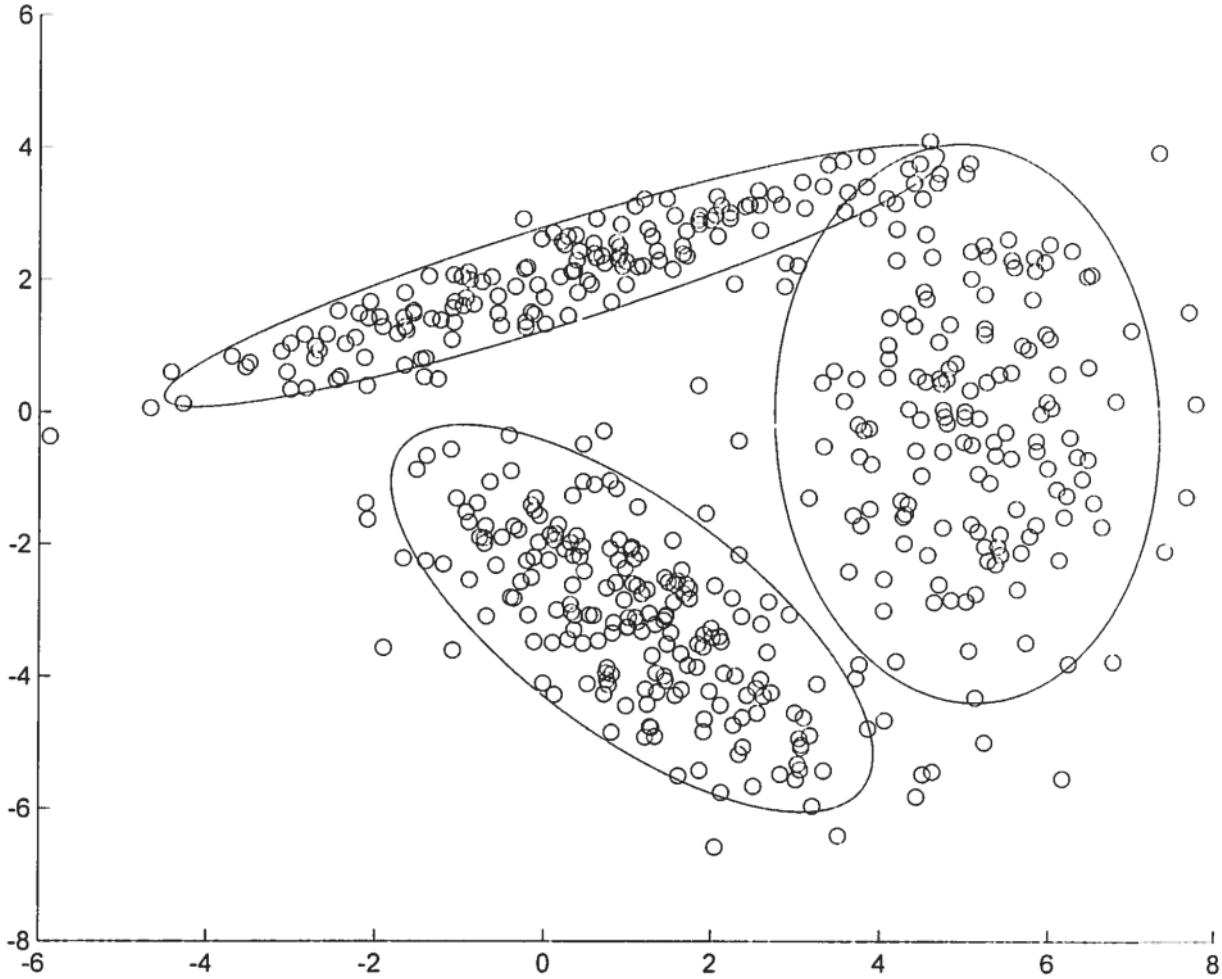
Clustering with Gaussian Mixture Models

Suppose we wish to cluster the following data set:



The data cluster naturally into 3 groups. Each cluster

is naturally described by a bivariate Gaussian density. In the figure below, the ellipses represent 90% contours (contours that contain 90% of the probability mass) of the components of a Gaussian mixture model (GMM) learned by maximum likelihood estimation.



In these notes, we'll learn about GMMs and the EM algorithm, an iterative algorithm for maximum likelihood estimation.

Gaussian Mixture Models

Gaussian Mixture Models

Recall the multivariate Gaussian density

$$\phi(x; \mu, \Sigma) = (2\pi)^{-\frac{d}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right\}$$

where

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

$$\mu \in \mathbb{R}^d$$

$$\Sigma \in \mathbb{R}^{d \times d}, \quad \Sigma > 0.$$

A random variable X follows a Gaussian mixture model if its probability density function f has the form

$$f(x) = \sum_{k=1}^{K} w_k \, \phi(x; \mu_k, \Sigma_k)$$

where

$$w_k \geq 0, \quad \sum_k w_k = 1$$

$$\mu_k \in \mathbb{R}^d$$

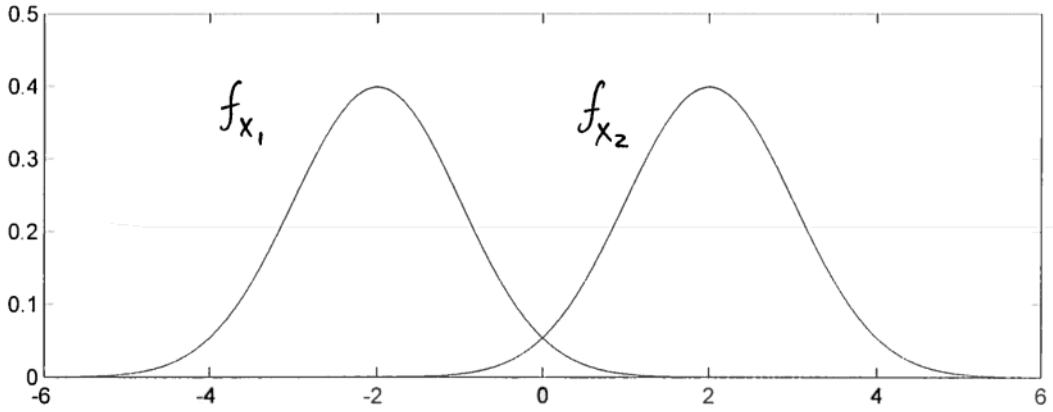
$$\Sigma_k \in \mathbb{R}^{d \times d}, \quad \Sigma_k > 0.$$

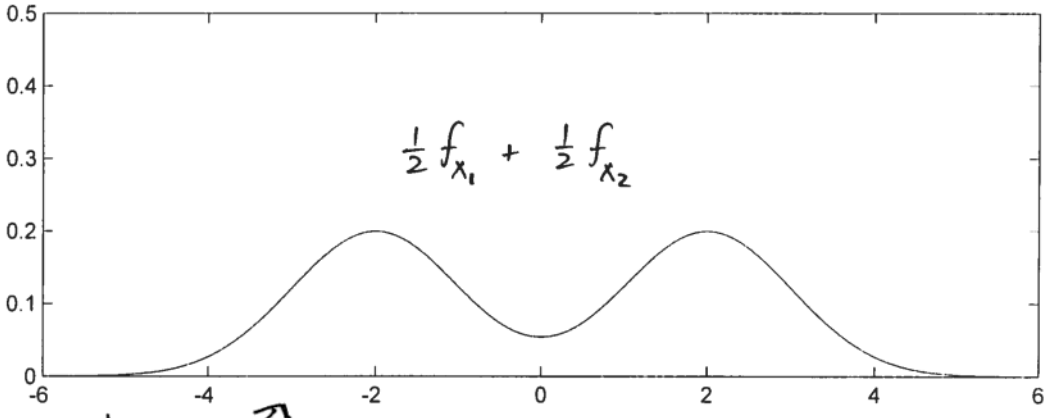
When first learning about GMMs, it is common to confuse

a mixture of Gaussians with a sum of Gaussians.

Just remember, a sum of Gaussian RVs is another Gaussian, and therefore unimodal. On the other hand, a mixture of Gaussians can be multimodal and is therefore non Gaussian.

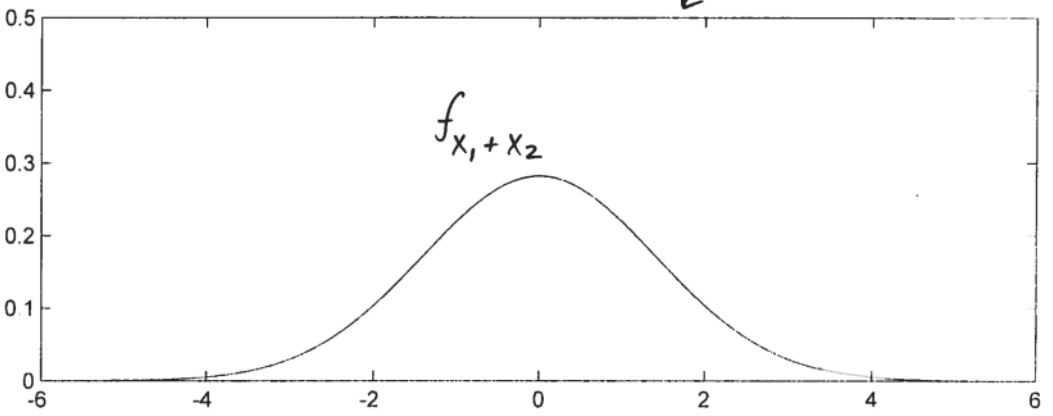
$X_1 \sim N(-2,1)$ $X_2 \sim N(2,1)$





mixture

not a mixture



$X_1 + X_2 \sim N(0,2)$

if X_1, X_2 are independent

Suppose

$$\theta = (w_1, \ldots, w_K, \mu_1, \ldots, \mu_K, \Sigma_1, \ldots, \Sigma_K)$$

is known. How can we simulate a realization of the GMM?

The idea:

- First, select a "component" k at random, weighted according to w_k
- Then draw a realization $X \sim N(\mu_k, \Sigma_k)$.

Why does this work? Let $S \in \{1, \ldots, K\}$ be a discrete RV such that

$$\Pr\{S = k\} = w_k.$$

The pdf $f(x)$ of X is such that for any event A,

$$\Pr(X \in A) = \int_A f(x)\, dx$$

But by the law of total expectation

$$\begin{aligned}
\Pr(X \in A) &= \sum_{k=1}^{K} \Pr(X \in A \mid S = k) \cdot \Pr(S = k) \\
&= \sum_{k=1}^{K} \left(\int_A \phi(x; \mu_k, \Sigma_k)\, dx \right) w_k \\
&= \int_A \left(\sum_{k=1}^{K} w_k \phi(x; \mu_k, \Sigma_k) \right) dx,
\end{aligned}$$

hence $f(x) = \sum_{k=1}^{K} w_k \phi(x; \mu_k, \Sigma_k)$.

The variable S is an example of a state variable, and is said to be hidden or latent, because it is usually unobserved. We will imagine that every realization of a GMM is associated with a hidden state variable.

Maximum Likelihood Estimation

To do clustering, we want to infer the parameters $\theta = (w_1, \ldots, w_K, \mu_1, \ldots, \mu_K, \Sigma_1, \ldots, \Sigma_K)$ from observations $x_1, \ldots, x_n \in \mathbb{R}^d$. To do this we will use maximum likelihood estimation, viewing K as fixed.

When $K = 1$, the MLE has a closed form solution:

$$\hat{\mu}_1 = \frac{1}{n} \sum_{i=1}^{n} x_i$$

$$\hat{\Sigma}_1 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{\mu}_1)(x_i - \hat{\mu}_1)^T$$

When $K > 1$, however, there is no closed form solution.

Denote $\underline{x} = (x_1, \ldots, x_n)$ for brevity. The likelihood is

$$L(\theta; \underline{x}) := \prod_{i=1}^{n} f(x_i; \theta)$$

$$= \prod_{i=1}^{n} \left(\sum_{k} w_k \phi(x_i ; \mu_k, \Sigma_k) \right)$$

and the log-likelihood is

$$\ell(\theta ; \underline{x}) := \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} w_k \phi(x_i ; \mu_k, \Sigma_k) \right)$$

Since maximization wrt θ is intractable, we will pursue an iterative strategy. This strategy hinges critically on the state variables

$$\underline{s} = (s_1, \ldots, s_n)$$

associated with the observations. A natural idea is an alternating algorithm like in k-means:

- Given θ, update the estimate of $\underline{s}$
- Given $\underline{s}$, update the estimate of θ

Each step can be performed efficiently.

The EM algorithm can be thought of as a variant where the cluster assignments are soft.

The Expectation Maximization Algorithm

The variable

$$\underline{z} = (\underline{x}, \underline{s})$$

is called the complete data. Define the indicator variable

$$\Delta_{i,k} = \begin{cases} 1 & \text{if } s_i = k \\ 0 & \text{if } s_i \neq k \end{cases}.$$

The complete-data log likelihood is

$$\begin{aligned}
&\log l(\theta; \underline{x}, \underline{s}) \\
&= \log \left(\prod_{i=1}^{n} \Pr\{S_i = s_i ; \theta\} \, f(x_i \mid s_i ; \theta) \right) \\
&= \log \left(\prod_{i} w_{s_i} \cdot \phi(x_i ; \mu_{s_i}, \Sigma_{s_i}) \right) \\
&= \sum_{i=1}^{n} \log \left(w_{s_i} \, \phi(x_i ; \mu_{s_i}, \Sigma_{s_i}) \right) \\
&= \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \Delta_{i,k} \cdot w_k \cdot \phi(x_i ; \mu_k, \Sigma_k) \right) \\
&= \sum_{i=1}^{n} \sum_{k=1}^{K} \Delta_{i,k} \left[\log w_k + \log \phi(x_i ; \mu_k, \Sigma_k) \right]
\end{aligned}$$

$\Delta_{i,k}$: only term depending on $\underline{S} = (S_1, \ldots, S_n)$

Ideally, if we knew S_i / $\Delta_{i,k}$, we could maximize the complete data log-likelihood, which is tractable.

Since we don't have the state variables, we can instead replace $\Delta_{i,k}$ with its expected value, and then maximize with respect to θ.

Of course, to compute the expected value of $\Delta_{i,k}$ requires knowledge of θ, so we have a "chicken and egg" problem. This suggest an iterative approach.

The EM algorithm is an iterative algorithm that produces a sequence of estimates $\theta^{(1)}, \theta^{(2)}, \ldots$ by alternating between the following steps:

E-Step

Calculate the expected complete-data log-likelihood:

$$Q(\theta, \theta^{(j)}) = \mathbb{E}_{\underline{S}|\underline{x}}\left[l(\theta; \underline{x}, \underline{S}) \,\middle|\, \underline{x}; \theta^{(j)} \right]$$

↑ Conditional expectation w.r.t. $\underline{S}|\underline{x}$. Here $\underline{S} = (S_1, \ldots, S_n)$ is capitalized since it is viewed as random.

↑ the pmf of $\underline{S}|\underline{x}$ depends on the GMM parameters;

use current estimate

This works out to

$$Q(\theta, \theta^{(j)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma_{i,k}^{(j)} \left[\log w_k + \log \phi(x_i; \mu_k, \Sigma_k) \right]$$

where

$$\gamma_{i,k}^{(j)} = \mathbb{E}\left[\Delta_{i,k} \mid \underline{x} ; \theta^{(j)} \right]$$

$$= \Pr\left\{ \Delta_{i,k} = 1 \mid \underline{x} ; \theta^{(j)} \right\}$$

$$= \Pr\left\{ S_i = k \mid \underline{x} ; \theta^{(j)} \right\}$$

$$= \frac{\Pr\{ S_i = k ; \theta^{(j)} \} \cdot f(x_i \mid S_i = k ; \theta^{(j)})}{f(x_i ; \theta^{(j)})}$$

Bayes rule

$$= \frac{w_k^{(j)} \cdot \phi(x_i ; \mu_k^{(j)}, \Sigma_k^{(j)})}{\sum_{\ell=1}^{K} w_\ell^{(j)} \phi(x_i ; \mu_\ell^{(j)}, \Sigma_\ell^{(j)})}.$$

This is the fraction of the density value at x_i explained by the k^{th} component. It is

sometimes called the responsibility of cluster k for x_i, and is a soft measure of cluster membership.

M-Step Compute

$$\theta^{(j+1)} = \arg\max_{\theta} Q(\theta, \theta^{(j)})$$

where

$$Q(\theta, \theta^{(j)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma_{i,k}^{(j)} \left[\log w_k - \frac{d}{2} \log 2\pi - \frac{1}{2} \log |\Sigma_k| - \frac{1}{2} (x_i - \mu_k)^T \Sigma^{-1} (x_i - \mu_k) \right].$$

We won't derive the solution (although it's not beyond our scope), which is

$$\mu_k^{(j+1)} = \frac{\sum_i \gamma_{i,k}^{(j)} x_i}{\sum_i \gamma_{i,k}^{(j)}}$$ ← weighted sample mean and covariance

$$\Sigma_k^{(j+1)} = \frac{\sum_i \gamma_{i,k}^{(j)} (x_i - \mu_k^{(j+1)})(x_i - \mu_k^{(j+1)})^T}{\sum_i \gamma_{i,k}^{(j)}}$$

$$w_1^{(j+1)} = \frac{1}{n} \sum^{n} \gamma_{i,k}^{(j)}$$ fraction of all data explained

$$w_k^{(j+1)} = \frac{1}{n} \sum_{i=1}^{n} \gamma_{i,k}^{(j)}$$

fraction of ... data explained by k^{th} component

The algorithm may be terminated when

$$\ell(\theta^{(j+1)}; \underline{x}) - \ell(\theta^{(j)}; \underline{x}) \leq \varepsilon$$

where ε is small. We will see later that the likelihood is non-decreasing.

Initialization

Like k-means, the EM alg. for GMMs is sensitive to initialization. One possibility is

$\mu_k^{(0)}$ = random x_i (distinct)

$\Sigma_k^{(0)}$ = sample covariance of all data (same for all k)

$w_k^{(0)} = \frac{1}{K}$

As with k-means, it may be beneficial to run the

algorithm many times and take the one with largest likelihood.

Another idea is to initialize EM by first running the k-means algorithm and basing $\theta^{(0)}$ on that cluster map.

Connection to k-means

The k-means algorithm can be viewed as a special case of the EM algorithm. Consider the GMM

$$f(x) = \sum_{i=1}^{n} w_k \phi(x_i ; \mu_k, \sigma^2 I)$$

common, isotropic covariance

where $\sigma^2 > 0$ is fixed. The EM algorithm is now to iterate

- $$\mu_k = \frac{\sum \gamma_{i,k} x_i}{\sum \gamma_{i,k}}$$

$$w_k = \frac{1}{n} \sum_{i=1}^{n} \gamma_{i,k}$$

- $$\gamma_{i,k} = \frac{w_k \phi(x_i, \mu_k, \sigma^2 I)}{\quad}$$

$$\sum_{\ell=1} w_\ell \, \phi(x_i ; \mu_\ell, \sigma^2 I)$$

Now, as $\sigma^2 \to 0$, we have

$$\gamma_{i,j} \longrightarrow \begin{cases} 1 & \text{if } k = \arg\min_\ell \|x_i - \mu_\ell\| \\ 0 & \text{otherwise,} \end{cases}$$

which gives the k-means algorithm.

The EM Algorithm in General

The EM algorithm is not specific to GMMs, but applies to many other maximum likelihood estimation problems where unobserved variables would make computations easier. As before, denote

$$\underline{x} = (x_1, \ldots, x_n)$$

$$\underline{s} = (s_1, \ldots, s_n)$$

where s_i is some variable that explains how x_i was generated. Also, let $\ell(\theta ; \underline{x})$ denote the log-likelihood and $\ell(\theta ; \underline{x}, \underline{s})$ the

complete-data log-likelihood. The general EM algorithm is as follows:

Initialize $\theta^{(0)}$

Repeat

E-Step: Form

$$Q(\theta, \theta^{(j)}) := \mathbb{E}\left[\ell(\theta; \underline{x}, \underline{S}) \mid \underline{x}; \theta^{(j)}\right]$$

M-Step: Compute

$$\theta^{(j+1)} = \arg\max_{\theta} Q(\theta, \theta^{(j)})$$

Until termination criterion satisfied

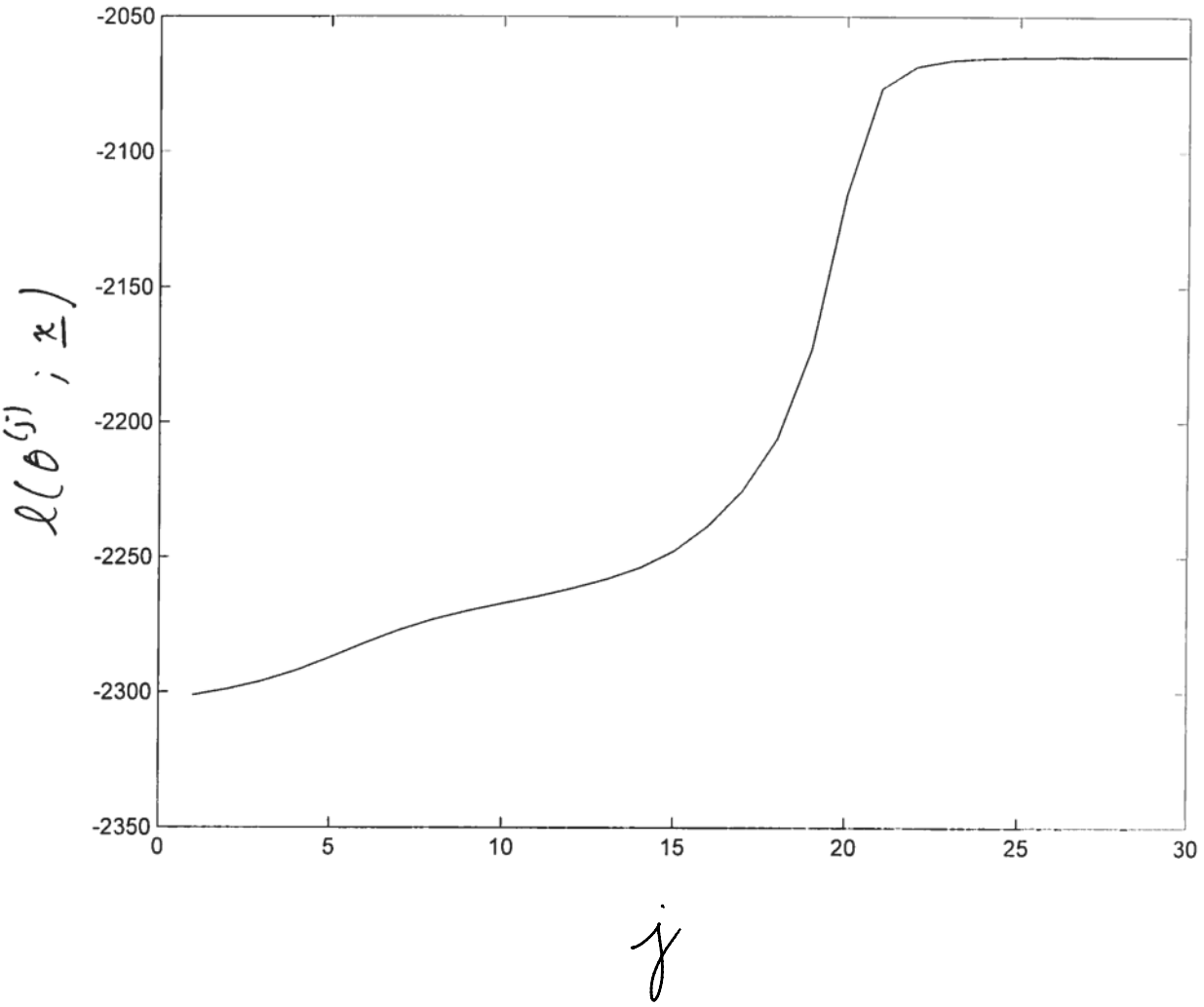
An important property of the EM algorithm is the following ascent property:

Theorem For each $j = 0, 1, 2, \ldots$

$$\ell(\theta^{(j+1)}; \underline{x}) \geq \ell(\theta^{(j)}; \underline{x})$$

Here's an example for the three component GMM

shown at the beginning of the notes:



To prove the monotonicity property, we need the following lemma:

Lemma Let Y be a random variable with density $q(y)$, and let $p(y)$ be another pdf. Then

$$\mathbb{E}_{Y\sim q}[\log p(Y)] \leq \mathbb{E}_{Y\sim q}[\log q(Y)],$$

and equality is attained iff $p(y) = q(y)$ almost everywhere.

Proof Jensen's inequality states that for any RV X and convex function ϕ, $\mathbb{E}[\phi(X)] \geq \phi(\mathbb{E}[X])$, and if ϕ is strictly convex, equality holds iff $X \equiv \mathbb{E}[X]$.

Therefore,

$$\mathbb{E}_{y\sim q}\left[\log\left(\frac{p(y)}{q(y)}\right)\right] \leq \log\left(\mathbb{E}_{y\sim q}\left[\frac{p(y)}{q(y)}\right]\right)$$

$$= \log\left(\int \frac{p(y)}{q(y)} \cdot q(y)\, dy\right)$$

$$= \log\left(\int p(y)\, dy\right)$$

$$= \log(1)$$

$$= 0.$$

Since log is strictly concave, equality holds iff $p(y) = q(y)$ almost everywhere.

Remark A similar result holds for discrete RVs.

Now, denote

$f(\underline{x}, \underline{s}; \theta)$ = complete data likelihood

$f(\underline{x}; \theta)$ = observed data likelihood

$$f(\underline{s} | \underline{x}; \theta) = \frac{f(\underline{x}, \underline{s}; \theta)}{f(\underline{x}; \theta)}$$

Then

$$Q(\theta, \theta^{(j)}) - l(\theta; \underline{x})$$

$$= \mathbb{E}\left[\log\left(\frac{f(\underline{x}, \underline{s}; \theta)}{f(\underline{x}; \theta)}\right) \middle| \underline{x}, \theta^{(j)}\right]$$

$$= \mathbb{E}\left[\log f(\underline{s} | \underline{x}; \theta) \middle| \underline{x}, \theta^{(j)}\right]$$

$$\leq \mathbb{E}\left[\log f(\underline{s} | \underline{x}; \theta^{(j)}) \middle| \underline{x}; \theta^{(j)}\right]$$

↖ by the lemma

$$= \mathbb{E}\left[\log\left(\frac{f(\underline{x}, s; \theta^{(j)})}{f(\underline{x}; \theta^{(j)})}\right) \Big| \underline{x}; \theta^{(j)}\right]$$

$$= Q(\theta^{(j)}, \theta^{(j)}) - \ell(\theta^{(j)}; \underline{x})$$

The ascent property follows from

$$\ell(\theta^{(j+1)}; \underline{x}) = \underbrace{Q(\theta^{(j+1)}, \theta^{(j)})}_{\text{def of } \theta^{(j+1)}} + \underbrace{\ell(\theta^{(j+1)}; \underline{x}) - Q(\theta^{(j+1)}, \theta^{(j)})}_{\text{above inequality}}$$

$$\geq Q(\theta^{(j)}, \theta^{(j)}) + \ell(\theta^{(j)}; \underline{x}) - Q(\theta^{(j)}, \theta^{(j)})$$

$$= \ell(\theta^{(j)}; \underline{x}).$$

The above inequality also establishes that

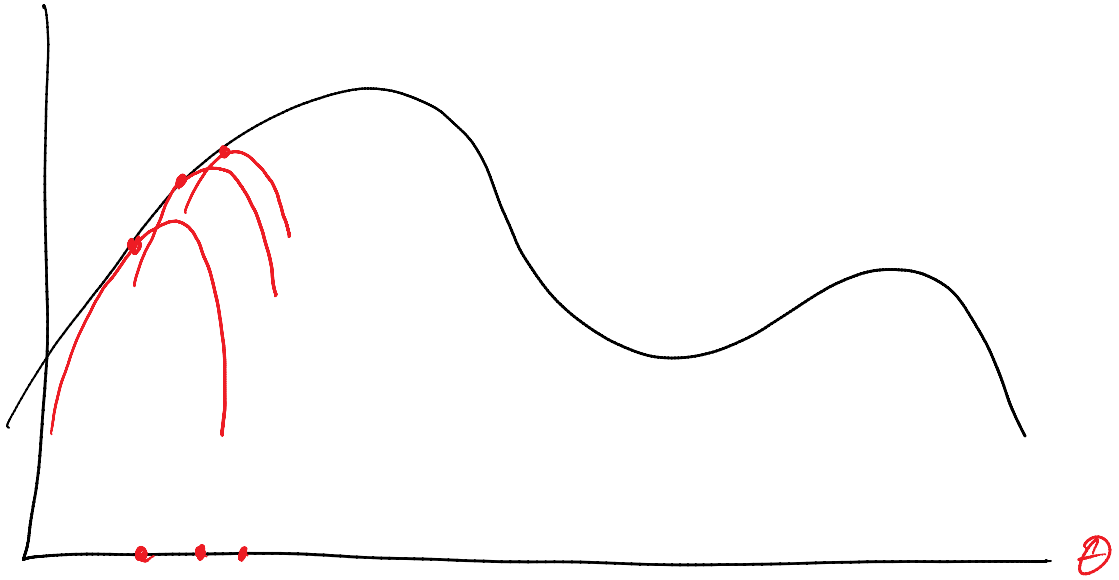
$$\ell(\theta; \underline{x}) \geq Q(\theta, \theta^{(j)}) + \ell(\theta^{(j)}; x) - Q(\theta^{(j)}, \theta^{(j)})$$

with equality when $\theta = \theta^{(j)}$. In other words

$$Q(\theta, \theta^{(j)}) + l(\theta^{(j)}; \underline{x}) - Q(\theta^{(j)}, \theta^{(j)})$$

minorizes $l(\theta; \underline{x})$. Thus, the EM algorithm can be viewed as a minorize-maximize (MM) algorithm that alternates between:

- form a minorizing function (E-Step)
- maximize the minorizing function (M-Step)



Final Thoughts

For an alternate derivation of EM see Andrew Ng's lecture notes.