Gaussian Mixture Models and Expectation Maximization

Duke Course Notes
Cynthia Rudin

Gaussian Mixture Models is a "soft" clustering algorithm, where each point probabilistically "belongs" to all clusters. This is different than k-means where each point belongs to one cluster ("hard" cluster assignments).

We remind the reader of the following fact: $\log \sum$ is not fun. In other words, when we have log of a sum, there is no way to reduce it.

This problem occurs within the log likelihood for GMM, so it is difficult to maximize the likelihood. The Expectation-Maximization (EM) procedure is a way to handle $\log \sum$. It uses Jensen's inequality to create a lower bound (called an auxiliary function) for the likelihood that uses $\sum \log$ instead. We can maximize the auxiliary function, which leads to an increase in the likelihood. We repeat this process at each iteration (constructing the auxiliary function and maximizing it), leading to a local maximum of the log likelihood for GMM. Let us walk through this process, deriving the EM algorithm along the way.

Here is GMM's generative model:

- First, generate which cluster i is going to be generated from:

  Here w is same as π from book

  $$z_i|\mathbf{w} \quad \sim \quad \text{Categorical}(\mathbf{w})$$

  which means that w_k is the probability that i's cluster is k. That is,

  $$P(z_i = k|\mathbf{w}) = w_k.$$

  Here, w_k are called the mixture weights, and they are a discrete probability distribution: $\sum_k w_k = 1$, $0 \leq w_k \leq 1$.

- Then, generate $\mathbf{x}_i$ from the cluster's distribution:

  $$\mathbf{x}_i|z_i = k \quad \sim \quad N\left(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)$$

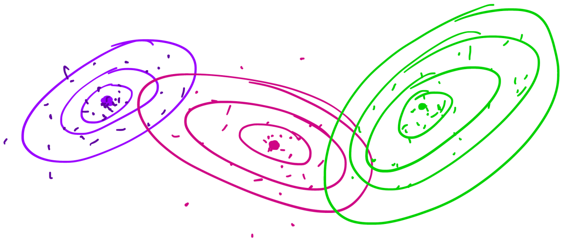
Just to recap the notation:

$$\begin{aligned}
\mathbf{x}_i &\rightarrow \text{data} \\
z_i &\rightarrow \text{cluster assignment for } i \\
\boldsymbol{\mu} &\rightarrow \text{center of cluster } k \\
\boldsymbol{\Sigma}_k &\rightarrow \text{spread of cluster } k \\
w_k &\rightarrow \text{proportion of data in cluster } k \text{ (mixture weights)}
\end{aligned}$$

As a reminder, here is the formula for the normal distribution:

$$p(\mathbf{X} = \mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{p/2}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{X} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{X} - \boldsymbol{\mu})\right).$$

Here is a picture of the generative process, where first I generated the cluster centers and covariances, and then generated points for each cluster, where the number of points I generated is proportional to the mixture weights.



Likelihood for GMM

$$\text{likelihood} \;=\; P\left(\{\mathbf{X}_1, ..., \mathbf{X}_n\} = \{\mathbf{x}_1, ..., \mathbf{x}_n\}|\mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\right),$$

where $\mathbf{w} = [w_1, ..., w_k]$, $\boldsymbol{\mu} = [\mu_1, .., \mu_k]$, $\boldsymbol{\Sigma} = [\boldsymbol{\Sigma}_1, ..., \boldsymbol{\Sigma}_k]$. I will denote the collection of these variables as θ. Assuming independence of data points,

$$\begin{aligned}
\text{likelihood}(\theta) &= \prod_i P(\mathbf{X}_i = \mathbf{x}_i|\theta), \\
&= \prod_i \sum_{k=1}^{K} \overset{N(x_i|\mu_K,\Sigma_K)}{P(\mathbf{X}_i = \mathbf{x}_i|z_i = k, \theta)}\overset{w_K}{P(z_i = k|\theta)} \quad \text{(law of total probability)} \\
&= \prod_i \sum_{k=1}^{K} N(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) w_k.
\end{aligned}$$

On the second and third lines above, the sum is over possible cluster assignments k for point i. Taking the log,

$$\begin{aligned}\log \text{likelihood}(\theta) &= \log \prod_i \sum_k P(\mathbf{X}_i = \mathbf{x}_i | z_i = k, \theta) P(z_i = k|\theta) \\ &= \sum_i \log \sum_k P(\mathbf{X}_i = \mathbf{x}_i | z_i = k, \theta) P(z_i = k|\theta).\end{aligned}$$

As we know, we cannot pass the log through the sum.

You might think this problem is specific just to the one we're working on (Gaussian mixture models) but the problem is *much* more general! In fact, every time you have a latent variable like $\mathbf{z}$, the same problem happens. Latent variables occur in lots of problems, not just clustering. For clustering, they happen almost all the time since you do not know which cluster a point may really belong to, so they cluster assignment is latent (hidden). Here is where we need a tool. That tool is Expectation-Maximization (EM). We will get back to Gaussian Mixture models after introducing EM.

Expectation Maximization

EM creates an iterative procedure where we update the z_i's and then update $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and $\mathbf{w}$. It is an alternating minimization scheme similar to k-means.

- E-step: compute cluster assignments (which are probabilistic)
- M-step: update θ (which are the clusters' properties)

Incidentally, if we looked instead at the "complete" log likelihood $p(\mathbf{x}, |\mathbf{z}, \theta)$ (meaning that you *know* the z_i's), there is no sum and the issue with the sum and the log goes away! This is because you no longer need to sum over k, you already know which cluster k unit i is in.

Let's start over from scratch. We are now in a very general setting. The data are still drawn independently, and each data has a hidden variable associated with it. Notation for data and hidden variables is:

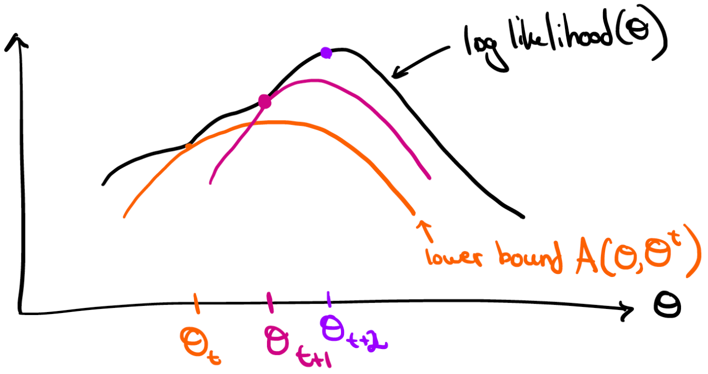
$x_1, ..., x_n$	data
$z_1, ..., z_n$	hidden variables, taking values $k = 1...K$
θ	parameters

Then,

$$\begin{aligned}
\log \text{likelihood}(\theta) &= \log P(X_1, ..., X_n = x_1, ..., x_n|\theta) \\
&= \sum_i \log P(X_i = x_i|\theta) \;\; \text{(by independence)} \\
&= \sum_i \log \sum_k P(X_i = x_i, Z_i = k|\theta) \;\; \text{(hidden variables)} \\
&= \sum_i \log \sum_k P(Z_i = k|\theta) P(X_i = x_i|Z_i = k, \theta).
\end{aligned}$$

The idea of Expectation Maximization (EM) is to find a lower bound on likelihood(θ) that involves $P(\mathbf{x}, \mathbf{z}|\theta)$. Maximizing the lower bound always leads to higher values of likelihood(θ).

The figure below illustrates a few iterations of EM. Starting at θ_t with iteration t in orange, we construct the surrogate lower bound $A(\theta, \theta_t)$. When we maximize it, our likelihood increases. The maximum of $A(\theta, \theta_t)$ occurs at θ_{t+1} that we will use at the next iteration. We evaluate the log likelihood of θ_{t+1}, again construct a surrogate lower bound $A(\theta, \theta_{t+1})$, and maximize it to get to the next iteration, which occurs at point θ_{t+2}, etc. At each iteration, the likelihood increases.



Note that this procedure leads to local maxima, not necessarily global maxima.

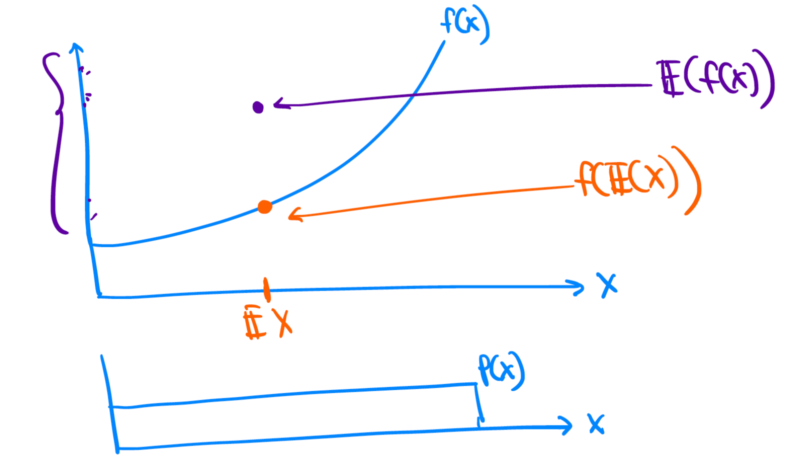
Let us write out the procedure for constructing A, starting with the log likelihood.

$$\begin{aligned}
\log \text{likelihood}(\theta) &= \sum_i \log \sum_k P(X_i = x_i, Z_i = k|\theta) \;\; \text{(from above)} \\
&= \sum_i \log \sum_k P(Z_i = k|x_i, \theta_t) \frac{P(X_i = x_i, Z_i = k|\theta)}{P(Z_i = k|x_i, \theta_t)}
\end{aligned}$$

where we have multiplied by 1 in disguise, namely $P(Z_i = k|x_i, \theta_t)$ in both the numerator and denominator. (This turns out to be the best possible choice for this 1 in disguise.) The weighted average $\sum_k P(Z_i = k|x_i, \theta_t)\langle\text{stuff}\rangle$ can be viewed as an expectation because it's a sum of elements weighted by probabilities that add up to 1. We will call it $\mathbb{E}_z$.

$$\text{log likelihood}(\theta) \;=\; \sum_i \log \mathbb{E}_z \frac{P(X_i = x_i, Z_i = k|\theta)}{P(Z_i = k|x_i, \theta_t)}.$$

We will now use Jensen's inequality for convex functions, which allows us to switch a log and an expectation. However, it is easy to forget which way Jensen's inequality goes. I have a picture that helps me remember. The distribution on the x-axis is uniform. We first find $\mathbb{E}(X)$, then $f(\mathbb{E}(X))$, which is fairly small. Afterwards we note that $\mathbb{E}(f(X))$ is larger, because it averages over $f(x)$, which has large values in it because f is convex.



At this point, it is clear which way Jensen's inequality goes.

Lemma (Jensen's Inequality). If f is convex, then $f(\mathbb{E}X) \leq \mathbb{E}(f(X))$.

If f is convex, $-f$ is concave, thus $-f(\mathbb{E}X) \geq -\mathbb{E}(f(X)) = \mathbb{E}(-f(X))$. Here, $-f(x) = \log(x)$ which is concave, thus, $\log(\mathbb{E}X) \geq \mathbb{E}\log X$.

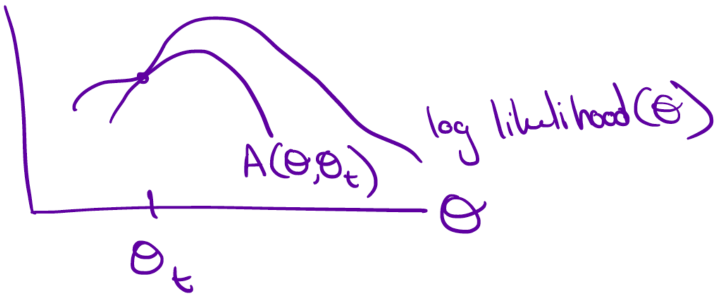
Back to where we were:

$$\begin{aligned}
\log \text{likelihood}(\theta) &= \sum_i \log \mathbb{E}_z \frac{P(X_i = x_i, Z_i = k|\theta)}{P(Z_i = k|x_i, \theta_t)} \\
&\geq \sum_i \mathbb{E}_z \log \frac{P(X_i = x_i, Z_i = k|\theta)}{P(Z_i = k|x_i, \theta_t)} \quad \text{(Jensen's inequality)} \\
&= \sum_i \sum_k P(Z_i = k|x_i, \theta_t) \log \frac{P(X_i = x_i, Z_i = k|\theta)}{P(Z_i = k|x_i, \theta_t)} =: A(\theta, \theta_t).
\end{aligned}$$

$A(\cdot, \theta_t)$ is called the auxiliary function.

Sanity check

Let's make sure that $A(\theta_t, \theta_t)$ is log likelihood(θ_t).



$$A(\theta_t, \theta_t) = \sum_i \sum_k P(Z_i = k|x_i, \theta_t) \log \frac{P(X_i = x_i, Z_i = k|\theta_t)}{P(Z_i = k|x_i, \theta_t)}$$

From the definition of conditional probability,
$P(X_i = x_i, Z_i = k|\theta_t) = P(Z_i = k|x_i, \theta_t)P(X_i = x_i|\theta_t)$. Plugging this in,

$$A(\theta_t, \theta_t) = \sum_i \sum_k P(Z_i = k|x_i, \theta_t) \log P(X_i = x_i|\theta_t).$$

Note that $\sum_k P(Z_i = k|x_i, \theta_t) = 1$ because this is a sum over a whole probability distribution, and the other term doesn't depend on k. So,

$$A(\theta_t, \theta_t) = \sum_i \log P(X_i = x_i|\theta_t) = \log \prod_i P(X_i = x_i|\theta_t) = \log \text{likelihood}(\theta_t).$$

Sanity check complete.

Back to EM

Recall our auxiliary function, which is a function of θ.

$$A(\theta, \theta_t) := \sum_i \sum_k P(Z_i = k | x_i, \theta_t) \log \frac{P(X_i = x_i, Z_i = k | \theta)}{P(Z_i = k | x_i, \theta_t)},$$

where I have highlighted two terms that are the same.

- E-step: compute $P(Z_i = k | x_i, \theta_t) =: \gamma_{ik}$ for each i, k. → Responsibility

- M-step:
$$\max_\theta A(\theta, \theta_t) = \sum_i \sum_j \gamma_{ik} \log \frac{P(X_i = x_i, Z_i = k | \theta)}{\gamma_{ik}}.$$
The term in the denominator doesn't depend on θ so it is not involved in the maximization. Thus it becomes:
$$\max_\theta \sum_i \sum_j \gamma_{ik} \log P(X_i = x_i, Z_i = k | \theta).$$
To maximize, we take the derivative and set it to 0, as usual.

Why is the "E" step called "Expectation" rather than "Probability"? Let us define indicator $\xi_{ik} = 1$ if $Z_i = k$ and 0 otherwise. (Remember, I showed you in a past lecture that expectations of indicator variables are probabilities.) Then,

$$P(Z_i = k | x_i, \theta_t) = 1 \cdot P(\xi_{ik} = 1 | x_i, \theta_t) + 0 \cdot P(\xi_{ik} = 0 | x_i, \theta_t) = \mathbb{E}_{\xi_{ik}} \xi_{ik}.$$

This might not be very satisfying, but it's too late to rename it I suppose.

Back to GMM

Let us now apply EM to GMM. Here is a reminder of the notation:

$$\begin{aligned} w_{kt} &= \text{probability to belong to cluster } k \text{ at iteration } t \\ \boldsymbol{\mu}_{kt} &= \text{mean of cluster } k \text{ at iteration } t \\ \boldsymbol{\Sigma}_{kt} &= \text{covariance of } k \text{ at iteration } t \end{aligned}$$

and θ_t is the collection of $(w_{kt}, \boldsymbol{\mu}_{kt}, \boldsymbol{\Sigma}_{kt})$'s at iteration t.

- E-step: Using Bayes Rule

  $$P(Z_i = k|\mathbf{x}_i, \theta_t) = \frac{P(\mathbf{X}_i = \mathbf{x}_i|z_i = k, \theta_t)P(Z_i = k|\theta_t)}{P(\mathbf{X}_i = \mathbf{x}_i|\theta_t)}.$$

  The denominator equals a sum over k of terms like those in the numerator, by the law of total probability. We can calculate all of the terms thanks to our assumptions for GMM.

  $$P(Z_i = k|\mathbf{x}_i, \theta_t) = \frac{N(\mathbf{x}_i\ ;\ \boldsymbol{\mu}_{kt}, \boldsymbol{\Sigma}_{kt})w_{kt}}{\sum_{k'} N(\mathbf{x}_i\ ;\ \boldsymbol{\mu}_{k't}, \boldsymbol{\Sigma}_{k't})w_{k't}} =: \gamma_{ik}.$$

  This is similar to k-means where we assign each point to a cluster at iteration t. Here, though the cluster assignments are probabilistic. (I could have indexed γ_{ik} also by t since it changes at each t, but instead I will just replace its value at each iteration.)

- M-step: Here is the auxiliary function we will maximize:

  $$\max_{\theta} A(\theta, \theta_t) = \sum_i \sum_j \gamma_{ik} \log P(X_i = x_i, Z_i = k|\theta).$$

  Update θ, which is the collection $\mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$, by setting derivatives of A to 0, with one constraint: $\sum_k w_k = 1$, so that the categorical distribution is well-defined. After a small amount of calculation (skipping steps here, setting the derivatives to zero and solving), the result for the cluster means is:

  $$\boldsymbol{\mu}_{k,t+1} = \frac{\sum_i \mathbf{x}_i \gamma_{ik}}{\sum_i \gamma_{ik}}.$$

  which is the mean of the $\mathbf{x}_i$'s, weighted by the probability of being in cluster k. (It's hard to imagine this calculation could turn out any other way.) Again skipping steps, setting the derivatives of the auxiliary function to 0 to get $\boldsymbol{\Sigma}_{k,t+1}$:

  $$\boldsymbol{\Sigma}_{k,t+1} = \frac{\sum_i \gamma_{ik}(\mathbf{x}_i - \boldsymbol{\mu}_{k,t+1})(\mathbf{x}_i - \boldsymbol{\mu}_{k,t+1})^T}{\sum_i \gamma_{ik}}.$$

  The update for $\mathbf{w}$ is tricker because of the constraint. We need to do constrained optimization. The Lagrangian is:

  $$L(\theta, \theta_t) = A(\theta, \theta_t) + \lambda \left(1 - \sum_k w_k\right)$$

where λ is the Lagrange multiplier. Remember that w_k is part of θ. Taking the derivative, and using index k' so as not to be confused with the sum over k:

$$\begin{aligned}\frac{\partial L(\theta, \theta_t)}{\partial w_{k'}} &= \frac{\partial A(\theta, \theta_t)}{\partial w_{k'}} - \lambda \\ &= \frac{\partial}{\partial w_{k'}} \left(\sum_i \sum_k \gamma_{ik} \log P(\mathbf{X}_i = \mathbf{x}_i, Z_i = k | \theta) \right) - \lambda. \qquad (1)\end{aligned}$$

Aside, we know, by the probabilistic model for generating data according to GMM (here we're using the fact that we solved for some of θ already for iteration $t+1$, so I'll refrain from coloring them),

$$\begin{aligned}P(\mathbf{X}_i = \mathbf{x}_i, Z_i = k | \theta) &= P(Z_i = k | \mathbf{w}) \cdot P(\mathbf{X}_i = \mathbf{x} | Z_i = k, \boldsymbol{\mu}_{k,t+1}, \boldsymbol{\Sigma}_{k,t+1}) - \lambda \\ &= w_k \cdot N(\mathbf{x}\ ;\ \boldsymbol{\mu}_{k,t+1}, \boldsymbol{\Sigma}_{k,t+1}) - \lambda.\end{aligned}$$

Plugging back into (1)

$$\begin{aligned}\frac{\partial L(\theta, \theta_t)}{\partial w_{k'}} &= \sum_i \frac{\partial}{\partial w_{k'}} \left[\gamma_{ik'} \log [w_{k'}\ N(\mathbf{x}; \boldsymbol{\mu}_{k',t+1}, \boldsymbol{\Sigma}_{k',t+1})] \right] - \lambda \\ &= \sum_i \frac{\partial}{\partial w_{k'}} \left[\gamma_{ik'} \log (w_{k',t+1}) \right] + \frac{\partial}{\partial w_{k'}} \left[N(\mathbf{x}; \boldsymbol{\mu}_{k',t+1}, \boldsymbol{\Sigma}_{k',t+1}) \right] - \lambda\end{aligned}$$

Here, $N(\mathbf{x}; \boldsymbol{\mu}_{k',t+1}, \boldsymbol{\Sigma}_{k',t+1})$ does not depend on $w_{k'}$ so we can remove that term.

$$\begin{aligned}\frac{\partial L(\theta, \theta_t)}{\partial w_{k'}} &= \sum_i \frac{\partial}{\partial w_{k'}} \left[\gamma_{ik'} \log (w_{k'}) \right] - \lambda \\ &= \sum_i \gamma_{ik'} \frac{1}{w_{k'}} - \lambda = \frac{1}{w_{k'}} \sum_i \gamma_{ik'} - \lambda\end{aligned}$$

Setting the derivative to 0, we can now solve for $w_{k',t+1}$:

$$w_{k',t+1} = \frac{\sum_i \gamma_{ik'}}{\lambda}.$$

We know that $\sum_{k'} w_{k',t+1}$=1, so λ is the normalization factor:

$$\lambda = \sum_k \sum_i \gamma_{ik} = \sum_i \left(\sum_k P(Z_i = k | \mathbf{x}_i, \theta) \right) = \sum_i 1 = n$$

where $\sum_k P(Z_i = k|\mathbf{x}_i, \theta) = 1$ because it is the sum over the whole probability distribution. Thus, we finally have our last update for the iterative procedure to optimize the parameters of GMM.

$$w_{k',t+1} = \frac{\sum_i \gamma_{ik'}}{n}.$$

We are now done with Gaussian mixture models. I'll leave you with a final big-picture summary of the update procedure, which looks quite similar to k-means:

E: What is the current estimate of the probability that $\mathbf{x}_i$ comes from cluster k? It is γ_{ik}.

M: Update parameters $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$ and $\mathbf{w}$.

In k means,

E → what is the cluster that x_i comes from?

M → update the mean for each cluster (μ)

d - dimensions

c - clusters

c mixture weights

for each cluster:

d means & d^2 variance terms

$$\gamma_{ij} = \frac{\pi_K \cdot (2\pi)^{-d/2} \cdot |\Sigma_K|^{-1/2} \cdot \exp((x-\mu)^T \Sigma_j^{-1} (x-\mu))}{\pi_1 \cdot (2\pi)^{-d/2} |\Sigma_1|}$$

$c + c \times (d + d^2)$

$c = 3$ and $d = 8$

$3 + 3 \times (8 + 64)$

$3 + 3 \times 72 = 219$ parameters

$c(1 + d + d^2)$

$6 \times (1 + d + d^2) = \boxed{418 \text{ parameters}}$