Gaussian mixture models and the EM algorithm

Ramesh Sridharan*

These notes give a short introduction to Gaussian mixture models (GMMs) and the Expectation-Maximization (EM) algorithm, first for the specific case of GMMs, and then more generally. These notes assume you're familiar with basic probability and basic calculus. If you're interested in the full derivation (Section 3), some familiarity with entropy and KL divergence is useful but not strictly required.

The notation here is borrowed from *Introduction to Probability* by Bertsekas & Tsitsiklis: random variables are represented with capital letters, values they take are represented with lowercase letters, p_X represents a probability distribution for random variable X, and $p_X(x)$ represents the probability of value x (according to p_X). We'll also use the shorthand notation X_1^n to represent the sequence $X_1, X_2, \ldots, X_n$, and similarly x_1^n to represent $x_1, x_2, \ldots, x_n$.

These notes follow a development somewhat similar to the one in *Pattern Recognition and Machine Learning* by Bishop.

1 Review: the Gaussian distribution

If random variable X is Gaussian, it has the following PDF:

$$p_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

The two parameters are μ, the mean, and σ^2, the variance (σ is called the standard deviation).

We'll use the terms "Gaussian" and "normal" interchangeably to refer to this distribution. To save us some writing, we'll write $p_X(x) = \mathcal{N}(x; \mu, \sigma^2)$ to mean the same thing (where the $\mathcal{N}$ stands for normal).

1.1 Parameter estimation for Gaussians: μ

Suppose we have i.i.d observations X_1^n from a Gaussian distribution with unknown mean μ and known variance σ^2. If we want to find the maximum likelihood estimate for the

*Contact: rameshvs@csail.mit.edu

parameter μ, we'll find the log-likelihood, differentiate, and set it to 0.

$$p_{X_1^n}(x_1^n) = \prod_{i=1}^{n} \mathcal{N}(x_i; \mu, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-(x_1-\mu)^2/2\sigma^2}$$

$$\ln p_{X_1^n}(x_1^n) = \sum_{i=1}^{n} \ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) - \frac{1}{2\sigma^2}(x_i - \mu)^2$$

$$\frac{d}{d\mu} \ln p_{X_1^n}(x_1^n) = \sum_{i=1}^{n} \frac{1}{\sigma^2}(x_i - \mu)$$

Setting this equal to 0, we see that the maximum likelihood estimate is $\widehat{\mu} = \frac{1}{N}\sum_i x_i$: it's the average of our observed samples. Notice that this estimate doesn't depend on the variance σ^2! Even though we started off by saying it was known, its value didn't matter.

2 Gaussian Mixture Models

A Gaussian mixture model (GMM) is useful for modeling data that comes from one of several groups: the groups might be different from each other, but data points within the same group can be well-modeled by a Gaussian distribution.

2.1 Examples

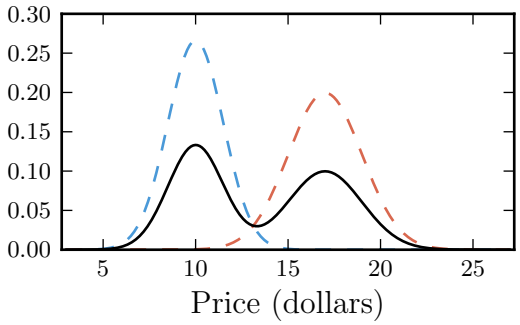
For example, suppose the price of a randomly chosen paperback book is normally distributed with mean \$10.00 and standard deviation \$1.00. Similarly, the price of a randomly chosen hardback is normally distributed with mean \$17 and variance \$1.50. Is the price of a randomly chosen book normally distributed?

The answer is no. Intuitively, we can see this by looking at the fundamental property of the normal distribution: it's highest near the center, and quickly drops off as you get farther away. But, the distribution of a randomly chosen book is bimodal: the center of the distribution is near \$13, but the probability of finding a book near that price is lower than the probability of finding a book for a few dollars more or a few dollars less. This is illustrated in Figure 1a.

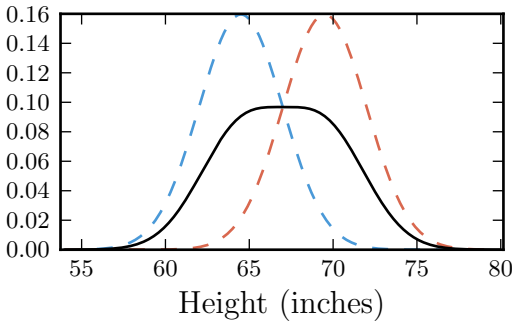
Another example: the height of a randomly chosen man is normally distributed with a mean around 5′9.5" and standard deviation around 2.5". Similarly, the height of a randomly chosen woman is normally distributed with a mean around 5′4.5" and standard deviation around 2.5" [1] Is the height of a randomly chosen person normally distributed?

The answer is again no. This one is a little more deceptive: because there's so much overlap between the height distributions for men and for women, the overall distribution is in fact highest at the center. But it's still not normally distributed: it's too wide and flat in the center (we'll formalize this idea in just a moment). This is illustrated in Figure 1b. These are both examples of *mixtures of Gaussians*: distributions where we have several groups and

[1] In the metric system, the means are about 177 cm and 164 cm, and the standard deviations are about 6 cm.



(a) Probability density for paperback books (red), hardback books (blue), and all books (black, solid)



(b) Probability density for heights of women (red), heights of men (blue), and all heights (black, solid)

Figure 1: Two Gaussian mixture models: the component densities (which are Gaussian) are shown in dotted red and blue lines, while the overall density (which is not) is shown as a solid black line.

the data within each group is normally distributed. Let's look at this a little more formally with heights.

2.2 The model

Formally, suppose we have people numbered $i = 1, \ldots, n$. We observe random variable $Y_i \in \mathbb{R}$ for each person's height, and assume there's an unobserved label $C_i \in \{M, F\}$ for each person representing that person's gender [2]. Here, the letter c stands for "class". In general, we can have any number of possible labels or classes, but we'll limit ourselves to two for this example. We'll also assume that the two groups have the same known variance σ^2, but different unknown means μ_M and μ_F. The distribution for the class labels is Bernoulli:

$$p_{C_i}(c_i) = q^{\mathbb{1}(c_i=M)}(1-q)^{\mathbb{1}(c_i=F)}$$

We'll also assume q is known. To simplify notation later, we'll let $\pi_M = q$ and $\pi_F = 1 - q$, so we can write

$$p_{C_i}(c_i) = \prod_{c \in \{M,F\}} \pi_c^{\mathbb{1}(c_i=c)} \tag{1}$$

The conditional distributions within each class are Gaussian:

$$p_{Y_i|C_i}(y_i|c_i) = \prod_c \mathcal{N}(y_i; \mu_c, \sigma^2)^{\mathbb{1}(c_i=c)} \tag{2}$$

[2] Naive Bayes model, this is somewhat similar. However, here our features are always Gaussian, and in the general case of more than 1 dimension, we won't assume independence of the features.

2.3 Parameter estimation: a first attempt

Suppose we observe i.i.d. heights $Y_1 = y_1, \ldots, Y_n = y_n$, and we want to find maximum likelihood estimates for the parameters μ_M and μ_F. This is an *unsupervised learning* problem: we don't get to observe the male/female labels for our data, but we want to learn parameters based on those labels.[3]

Exercise: Given the model setup in (1) and (2), compute the joint density of all the data points $p_{Y_1,\ldots,Y_N}(y_1, \ldots, y_n)$ in terms of μ_M, μ_F, σ, and q. Take the log to find the log-likelihood, and then differentiate with respect to μ_M. Why is this hard to optimize?

Solution: We'll start with the density for a single data point $Y_i = y_i$:

$$\begin{aligned} p_{Y_i}(y_i) &= \sum_{c_i} p_{C_i}(c_i) p_{Y_i|C_i}(y_i|c_i) \\ &= \sum_{c_i} \left(\pi_c \mathcal{N}(y_i; \mu_C, \sigma^2)\right)^{\mathbb{1}(c_i=c)} \\ &= q\mathcal{N}(y_i; \mu_M, \sigma^2) + (1-q)\mathcal{N}(y_i; \mu_F, \sigma^2) \end{aligned}$$

Now, the joint density of all the observations is:

$$p_{Y_1^n}(y_1^n) = \prod_{i=1}^{n} \left(q\mathcal{N}(y_i; \mu_M, \sigma^2) + (1-q)\mathcal{N}(y_i; \mu_F, \sigma^2)\right),$$

and the log-likelihood of the parameters is then

$$\ln p_{Y_1^n}(y_1^n) = \sum_{i=1}^{n} \ln \left(\pi_M \mathcal{N}(y_i; \mu_M, \sigma^2) + \pi_F \mathcal{N}(y_i; \mu_F, \sigma^2)\right), \tag{3}$$

We've already run into a small snag: the sum prevents us from applying the log to the normal densities inside. So, we should already be a little worried that our optimization won't go as smoothly as it did for the simple mean estimation we did back in Section 1.1. By symmetry, we only need to look at one of the means; the other will follow almost the same process. Before we dive into differentiating, we note that

$$\begin{aligned} \frac{d}{d\mu}\mathcal{N}(x; \mu, \sigma^2) &= \frac{d}{d\mu}\left[\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right] \\ &= \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \cdot \frac{2(x-\mu)}{2\sigma^2} \\ &= \mathcal{N}(x; \mu, \sigma^2) \cdot \frac{(x-\mu)}{\sigma^2} \end{aligned}$$

Differentiating (3) with respect to μ_M, we obtain

$$\sum_{i=1}^{n} \frac{1}{\pi_M \mathcal{N}(y_i; \mu_M, \sigma^2) + \pi_F \mathcal{N}(y_i; \mu_F, \sigma^2)} \pi_M \mathcal{N}(y_i; \mu_M, \sigma^2) \frac{y_i - \mu_M}{\sigma^2} = 0 \tag{4}$$

At this point, we're stuck. We have a mix of ratios of exponentials and linear terms, and there's no way we can solve this in closed form to get a clean maximum likelihood expression!

[3] Note that in a truly unsupervised setting, we wouldn't be able to tell which one of the two was male and which was female: we'd find two distinct clusters and have to label them based on their values after the fact.

2.4 Using hidden variables and the EM Algorithm

Taking a step back, what would make this computation easier? If we knew the hidden labels C_i exactly, then it would be easy to do ML estimates for the parameters: we'd take all the points for which $C_i = M$ and use those to estimate μ_M like we did in Section 1.1, and then repeat for the points where $C_i = F$ to estimate μ_F. Motivated by this, let's try to compute the distribution for C_i given the observations. We'll start with Bayes' rule:

$$\begin{aligned} p_{C_i|Y_i}(c_i|y_i) &= \frac{p_{Y_i|C_i}(y_i|c_i)p_{C_i}(c_i)}{p_{Y_i}(y_i)} \\ &= \frac{\prod_{c\in\{M,F\}} \left(\pi_c \mathcal{N}(y_i;\mu_c,\sigma^2)\right)^{\mathbb{1}(c=c_i)}}{\pi_M \mathcal{N}(y_i;\mu_M,\sigma^2) + \pi_F \mathcal{N}(y_i;\mu_F,\sigma^2)} = q_{C_i}(c_i) \end{aligned} \tag{5}$$

Let's look at the posterior probability that $C_i = M$:

$$p_{C_i|Y_i}(M|y_i) = \frac{\pi_M \mathcal{N}(y_i;\mu_M,\sigma^2)}{\pi_M \mathcal{N}(y_i;\mu_M,\sigma^2) + \pi_F \mathcal{N}(y_i;\mu_F,\sigma^2)} = q_{C_i}(M) \tag{6}$$

This should look very familiar: it's one of the terms in (4)! And just like in that equation, we have to know all the parameters in order to compute this too. We can rewrite (4) in terms of q_{C_i}, and cheat a little by pretending it doesn't depend on μ_M:

$$\sum_{i=1}^{n} q_{C_i}(M)\frac{y_i - \mu_M}{\sigma^2} = 0 \tag{7}$$

$$\mu_M = \frac{\sum_{i=1}^{n} q_{C_i}(M) y_i}{\sum_{i=1}^{n} q_{C_i}(M)} \tag{8}$$

This looks much better: μ_M is a weighted average of the heights, where each height is weighted by how likely that person is to be male. By symmetry, for μ_F, we'd compute the weighted average with weights $q_{C_i}(F)$.

So now we have a circular setup: we could easily compute the posteriors over C_1^n if we knew the parameters, and we could easily estimate the parameters if we knew the posterior over C_1^n. This naturally suggests the following strategy: we'll fix one and solve for the other. This approach is generally known as the *EM algorithm*. Informally, here's how it works:

- First, we fix the parameters (in this case, the means μ_M and μ_F of the Gaussians) and solve for the posterior distribution for the hidden variables (in this case, q_{C_i}, the class labels). This is done using (6).
- Then, we fix the posterior distribution for the hidden variables (again, that's q_{C_i}, the class labels), and optimize the parameters (the means μ_M and μ_F) using the expected values of the hidden variables (in this case, the probabilities from q_{C_i}). This is done using (4).

- Repeat the two steps above until the values aren't changing much (i.e., until convergence).

Note that in order to get the process started, we have to initialize the parameters somehow. In this setting, the initialization matters a lot! For example, suppose we set $\mu_M = 3'$ and $\mu_F = 5'$. Then the computed posteriors q_{C_i} would all favor F over M (since most people are closer to $5'$ than $3'$), and we would end up computing μ_F as roughly the average of all our heights, and μ_M as the average of a few short people.

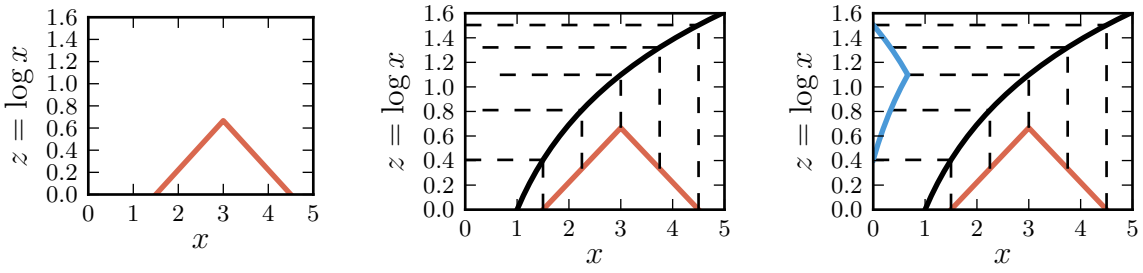


Figure 2: An illustration of a special case of Jensen's inequality: for any random variable X, $\mathbb{E}[\log X] \geq \log \mathbb{E}[X]$. Let X be a random variable with PDF as shown in red. Let $Z = \log X$. The center and right figures show how to construct the PDF for Z (shown in blue): because of the log, it's skewed towards smaller values compared to the PDF for X. $\log \mathbb{E}[X]$ is the point given by the center dotted black line, or $\mathbb{E}[X]$. But $\mathbb{E}[\log X]$, or $\mathbb{E}[Z]$, will always be smaller (or at least will never be larger) because the log "squashes" the bigger end of the distribution (where Z is larger) and "stretches" the smaller end (where Z is smaller).

3 The EM Algorithm: a more formal look

Note: This section assumes you have a basic familiarity with measures like entropy and KL divergence, and how they relate to expectations of random variables. You can still understand the algorithm itself without knowing these concepts, but the derivations depend on understanding them.

By this point you might be wondering what the big deal is: the algorithm described above may sound like a hack where we just arbitrarily fix some stuff and then compute other stuff. But, as we'll show in a few short steps, the EM algorithm is actually maximizing a lower bound on the log likelihood (in other words, each step is guaranteed to improve our answer until convergence). A bit more on that later, but for now let's look at how we can derive the algorithm a little more formally.

Suppose we have observed a random variable Y. Now suppose we also have some hidden variable C that Y depends on. Let's say that the distributions of C and Y have some parameters θ that we don't know, but are interested in finding.

In our last example, we observed heights $Y = \{Y_1, \ldots, Y_n\}$ with hidden variables (gender labels) $C = \{C_1, \ldots, C_n\}$ (with i.i.d. structure over Y and C), and our parameters θ were μ_M and μ_F, the mean heights for each group.

Before we can actually derive the algorithm, we'll need a key fact: Jensen's inequality. The specific case of Jensen's inequality that we need says that:

$$\log(\mathbb{E}[X]) \geq \mathbb{E}[\log(X)] \tag{9}$$

For a geometric intuition of why this is true, see Figure 2. For a proof and more detail, see Wikipedia [4] [5].

[4] `http://en.wikipedia.org/wiki/Jensen_inequality`

[5] This figure is based on the one from the Wikipedia article, but for a concave function instead of a convex one.

Now we're ready to begin: Section 3.1 goes through the derivation quickly, and Section 3.2 goes into more detail about each step.

3.1 The short verion

We want to maximize the log-likelihood:

$$\begin{aligned}
&\log p_Y(y;\theta) \\
\text{(Marginalizing over } C \text{ and introducing } q_C(c)/q_C(c)) \quad &= \log\left(\sum_c q_C(c)\frac{p_{Y,C}(y,c;\theta)}{q_C(c)}\right) \\
\text{(Rewriting as an expectation)} \quad &= \log\left(\mathbb{E}_{q_C}\left[\frac{p_{Y,C}(y,C;\theta)}{q_C(C)}\right]\right) \\
\text{(Using Jensen's inequality)} \quad &\geq \mathbb{E}_{q_C}\left[\log\frac{p_{Y,C}(y,C;\theta)}{q_C(C)}\right]
\end{aligned}$$

Let's rearrange the last version:

$$\mathbb{E}_{q_C}\left[\log\frac{p_{Y,C}(y,C;\theta)}{q_C(C)}\right] = \mathbb{E}_{q_C}\left[\log p_{Y,C}(y,C;\theta)\right] - \mathbb{E}_{q_C}[\log q_C(C)]$$

Maximizing with respect to θ will give us:

$$\boxed{\widehat{\theta} \leftarrow \operatorname*{argmax}_\theta \mathbb{E}_{q_C}\left[\log p_{Y,C}(y,C;\theta)\right]}$$

That's the M-step. Now we'll rearrange a different way:

$$\begin{aligned}
\mathbb{E}_{q_C}\left[\log\frac{p_{Y,C}(y,C;\theta)}{q_C(C)}\right] &= \mathbb{E}_{q_C}\left[\log\frac{p_Y(y;\theta)p_{C|Y}(C|y;\theta)}{q_C(C)}\right] \\
&= \log p_Y(y;\theta) - \mathbb{E}_{q_C}\left[\log\frac{q_C(C)}{p_{C|Y}(C|y;\theta)}\right] \\
&= \log p_Y(y;\theta) - D(q_C(\cdot)||p_{C|Y}(\cdot|y;\theta))
\end{aligned}$$

Maximizing with respect to q_C will give us:

$$\boxed{\widehat{q_C}(\cdot) \leftarrow p_{C|Y}(\cdot|y;\theta)}$$

That's the E-step.

3.2 The long version

We'll try to do maximum likelihood. Just like we did earlier, we'll try to compute the log-likelihood by marginalizing over C:

$$\log p_Y(y;\theta) = \log\left(\sum_c p_{Y,C}(y,c)\right)$$

Just like in Section 2.3, we're stuck here: we can't do much with a log of a sum. Wouldn't it be nice if we could swap the order of them? Well, an expectation is a special kind of sum, and Jensen's inequality lets us swap them if we have an expectation. So, we'll introduce a new distribution q_C for the hidden variable C:

$$\log p_Y(y;\theta) \tag{10}$$

(Marginalizing over C and introducing $q_C(c)/q_C(c)$) $$= \log\left(\sum_c q_C(c)\frac{p_{Y,C}(y,c;\theta)}{q_C(c)}\right)$$

(Rewriting as an expectation) $$= \log\left(\mathbb{E}_{q_C}\left[\frac{p_{Y,C}(y,C;\theta)}{q_C(C)}\right]\right)$$

(Using Jensen's inequality) $$\geq \mathbb{E}_{q_C}\left[\log\frac{p_{Y,C}(y,C;\theta)}{q_C(C)}\right] \tag{11}$$

Using definition of conditional probability $$= \mathbb{E}_{q_C}\left[\log\frac{p_Y(y;\theta)p_{C|Y}(C|y;\theta)}{q_C(C)}\right] \tag{12}$$

Now we have a lower bound on $\log p_Y(y;\theta)$ that we can optimize pretty easily. Since we've introduced q_C, we now want to maximize this quantity with respect to both θ and q_C.

We'll use (11) and (12), respectively, to do the optimizations separately. First, using (11) to find the best parameters:

$$\mathbb{E}_{q_C}\left[\log\frac{p_{Y,C}(y,C;\theta)}{q_C(C)}\right] = \mathbb{E}_{q_C}\left[\log p_{Y,C}(y,C;\theta)\right] - \mathbb{E}_{q_C}[\log q_C(C)]$$

In general, q_C doesn't depend on θ, so we'll only care about the first term:

$$\boxed{\widehat{\theta} \leftarrow \operatorname*{argmax}_\theta \mathbb{E}_{q_C}\left[\log p_{Y,C}(y,C;\theta)\right]} \tag{13}$$

This is called the *M-step*: the M stands for maximization, since we're maximizing with respect to the parameters. Now, let's find the best q_C using (12).

$$\mathbb{E}_{q_C}\left[\log\frac{p_Y(y;\theta)p_{C|Y}(C|y;\theta)}{q_C(C)}\right] = \mathbb{E}_{q_C}[\log p_Y(y;\theta)] + \mathbb{E}_{q_C}\left[\log\frac{p_{C|Y}(C|y;\theta)}{q_C(C)}\right]$$

The first term doesn't depend on c, and the second term almost looks like a KL divergence:

$$= \log p_Y(y;\theta) - \mathbb{E}_{q_C}\left[\log\frac{q_C(C)}{p_{C|Y}(C|y;\theta)}\right]$$
$$= \log p_Y(y;\theta) - D(q_C(\cdot)||p_{C|Y}(\cdot|y;\theta))\ ^6 \tag{14}$$

So, when maximizing this quantity, we want to make the KL divergence as small as possible. KL divergences are always greater than or equal to 0, and they're exactly 0 when the two distributions are equal. So, the optimal q_C is $p_{C|Y}(c|y;\theta)$:

$$\boxed{\widehat{q_C}(c) \leftarrow p_{C|Y}(c|y;\theta)} \tag{15}$$

[6] Remember that this is a lower bound on $\log p_Y(y;\theta)$: that is, $\log p_Y(y;\theta) \geq \log p_Y(y;\theta) - D(q_C(\cdot)||p_{C|Y}(\cdot|y;\theta))$. From this, we can see that the "gap" in the lower bound comes entirely from the KL divergence term.

This is called the *E-step*: the E stands for expectation, since we're computing q_C so that we can use it for expectations. So, by alternating between (13) and (15), we can maximize a lower bound on the log-likelihood. We've also seen from (15) that the lower bound is tight (that is, it's equal to the log-likelihood) when we use (15).

3.3 The algorithm

Inputs: Observation y, joint distribution $p_{Y,C}(y, c; \theta)$, conditional distribution $p_{C|Y}(c|y; \theta)$, initial values $\theta^{(0)}$

function EM($p_{Y,C}(y, c; \theta), p_{C|Y}(c|y; \theta), \theta^{(0)}$)
  for iteration $t \in 1, 2, \ldots$ **do**
    $q_C^{(t)} \leftarrow p_{C|Y}(c|y; \theta^{(t-1)})$ (**E-step**)
    $\theta^{(t)} \leftarrow \text{argmax}_\theta \, \mathbb{E}_{q_C^{(t)}} [p_{Y,C}(y, C; \theta)]$ (**M-step**)
    if $\theta^{(t)} \approx \theta^{(t-1)}$ **then**
      return $\theta^{(t)}$

3.4 Example: Applying the general algorithm to GMMs

Now, let's revisit our GMM for heights and see how we can apply the two steps. We have the observed variable $Y = \{Y_1, \ldots, Y_n\}$, and the hidden variable $C = \{C_1, \ldots, C_n\}$. For the E-step, we have to compute the posterior distribution $p_{C|Y}(c|y)$, which we already did in (5) and (6). For the M-step, we have to compute the expected joint probability.

$$\begin{aligned}
\mathbb{E}_{q_C}[\ln p_{Y,C}(y, C)] &= \mathbb{E}_{q_C}[\ln p_{Y|C}(y|C) p_C(C)] \\
&= \mathbb{E}_{q_C}\left[\ln \prod_{i=1}^{n} \prod_{c \in \{M,F\}} \left(\pi_c \mathcal{N}(y_i; \mu_c, \sigma^2)\right)^{\mathbb{1}(C_i = c)}\right] \\
&= \mathbb{E}_{q_C}\left[\sum_{i=1}^{n} \sum_{c \in \{M,F\}} \mathbb{1}(C_i = c) \left(\ln \pi_c + \ln \mathcal{N}(y_i; \mu_c, \sigma^2)\right)\right] \\
&= \sum_{i=1}^{n} \sum_{c \in \{M,F\}} \mathbb{E}_{q_C}\left[\mathbb{1}(C_i = c)\right] \left(\ln \pi_c + \ln \frac{1}{\sigma\sqrt{2\pi}} - \frac{(y_i - \mu_c)^2}{2\sigma^2}\right)
\end{aligned}$$

$\mathbb{E}_{q_C}[\mathbb{1}(C_i = c)]$ is the probability that C_i is c, according to q. Now, we can differentiate with respect to μ_M:

$$\frac{d}{d\mu_M} \mathbb{E}_{q_C}[\ln p_{Y|C}(y|C) p_C(C)] = \sum_{i=1}^{n} q_{C_i}(M) \left(\frac{y_i - \mu_M}{\sigma^2}\right) = 0$$

This is exactly the same as what we found earlier in (7), so we know the solution will again be the weighted average from (8):

$$\mu_M = \frac{\sum_{i=1}^{n} q_{C_i}(M) y_i}{\sum_{i=1}^{n} q_{C_i}(M)}$$

Repeating the process for μ_F, we get

$$\mu_F = \frac{\sum_{i=1}^{n} q_{C_i}(F) y_i}{\sum_{i=1}^{n} q_{C_i}(F)}$$