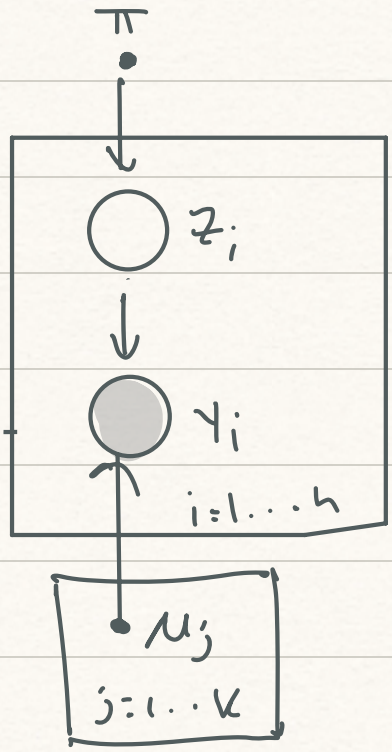
The expectation-maximization (EM) algorithm

Gaussian mixture model:



$$z_i \sim \text{Cat}(\pi)$$

$$y_i \mid z_i = j \sim \mathcal{N}(\mu_j, 1)$$

Model parameters

$$\theta \equiv \{\mu_{1:K}, \pi\}$$

"Complete data likelihood" or "joint probability"

$$p_\theta(y_{1:n}, z_{1:n}) = \prod_{i=1}^{n} \prod_{j=1}^{K} \left[\mathcal{N}(y_i; \mu_j, 1)\, \pi_j\right]^{\delta_{ij}}$$

where $\delta_{ij} \equiv \mathbb{1}(z_i = j)$

"Likelihood" or "marginal likelihood / evidence"

$$p_\theta(y_{1:n}) = \sum_{z_1=1}^{K} \cdots \sum_{z_n=1}^{K} p_\theta(y_{1:n}, z_{1:n})$$

↑ intractable

"Maximum likelihood estimation" or "type-II MLE"

$$\hat{\theta}_{MLE} \leftarrow \operatorname*{argmax}_\theta p_\theta(y_{1:n})$$

Note: There are two parallel interpretations. In some models, we will interpret Z as "missing data", in other models we will think of Z as "latent variables". When Z are "missing data", then P(Z, Y) is the "complete data likelihood" and P(Y) is just the "likelihood". When Z are "latent variables" then P(Y, Z) is the "joint", P(Y | Z) is the "likelihood", and P(Y) is the "marginal likelihood", and maximizing P(Y) is doing "type II MLE" (i.e., maximum marginal likelihood estimation). But this is all just semantics / interpretation; the math is what's important. We will call P(Y) the marginal likelihood or evidence, but some literature calls it the likelihood.

Evidence lower bound (ELBO)

$$\log P_\theta(Y) = \log \sum_Z P_\theta(Y, Z), \qquad Y \equiv Y_{1:n}, \quad Z \equiv Z_{1:n}$$

For any distribution q(Z) that is **absolutely continuous** wrt P(Y, Z):

$$= \log \mathop{\mathbb{E}}_{Z \sim q(Z)} \left[\frac{P_\theta(Y, Z)}{q(Z)} \right]$$

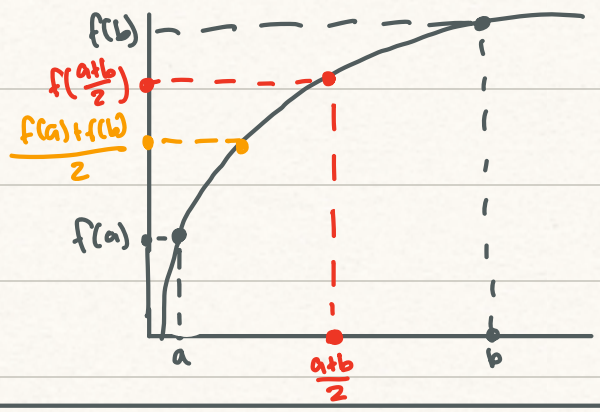
If p(Z=z, Y) = 0 then q(Z=z) = 0, for all z.

Switching the expectation and log, we then have by **Jensen's inequality**:

$$\geq \mathop{\mathbb{E}}_{q(Z)} \left[\log \frac{P_\theta(Y, Z)}{q(Z)} \right]$$

Jensen's inequality: For any concave function f(x), the expectation of f(x) lower bounds f(expectation of x).

$$f(\mathop{\mathbb{E}}_{P(x)}[x]) \geq \mathop{\mathbb{E}}_{P(x)}[f(x)]$$



This is a function of q and the parameters, which we will call the **ELBO**:

$$\equiv B(q, \theta)$$

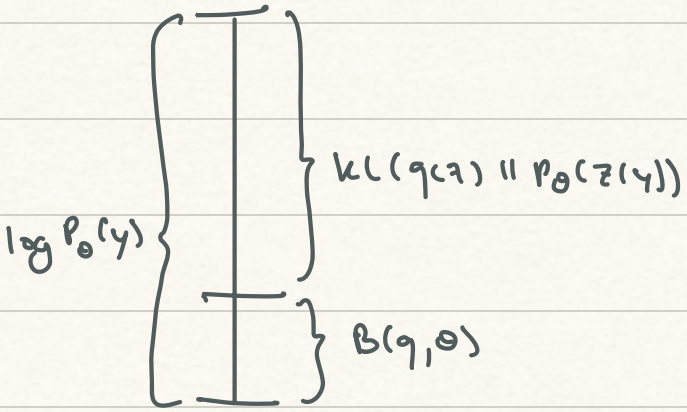
Let's do some ELBO surgery:

$$B(q, \theta) = \underbrace{\mathbb{E}_q[\log P_\theta(Y)]}_{= P_\theta(Y)} + \underbrace{\mathbb{E}_q\left[\log \frac{P_\theta(Z \mid Y)}{q(Z)}\right]}_{= -KL(q(Z) \,\|\, p_\theta(Z \mid Y))}$$

p(y) does not depend on z

this is a (negative) KL-divergence

Since the KL divergence must be non-negative, the ELBO must lower bound the evidence.



The Kullback-Leibler divergence gives a notion of similarity between two distributions q and p over the same support. It is not a metric (not symmetric) and is 0 only if p=q.

- $KL(q \| p) = \int q(z) \log \frac{q(z)}{p(z)} dz$
- $KL(q \| p) \geq 0$
- $KL(q \| p) = 0$ IFF $q = p$
- $KL(q \| p) \neq KL(p \| q)$ "not necessarily"

The EM algorithm

Goal: $\hat{\theta} \leftarrow \underset{\theta}{\operatorname{argmax}}\ p_\theta(y)$

initialize ϑ_0

for $m = 1, 2, \ldots$ until convergence in ϑ_m:

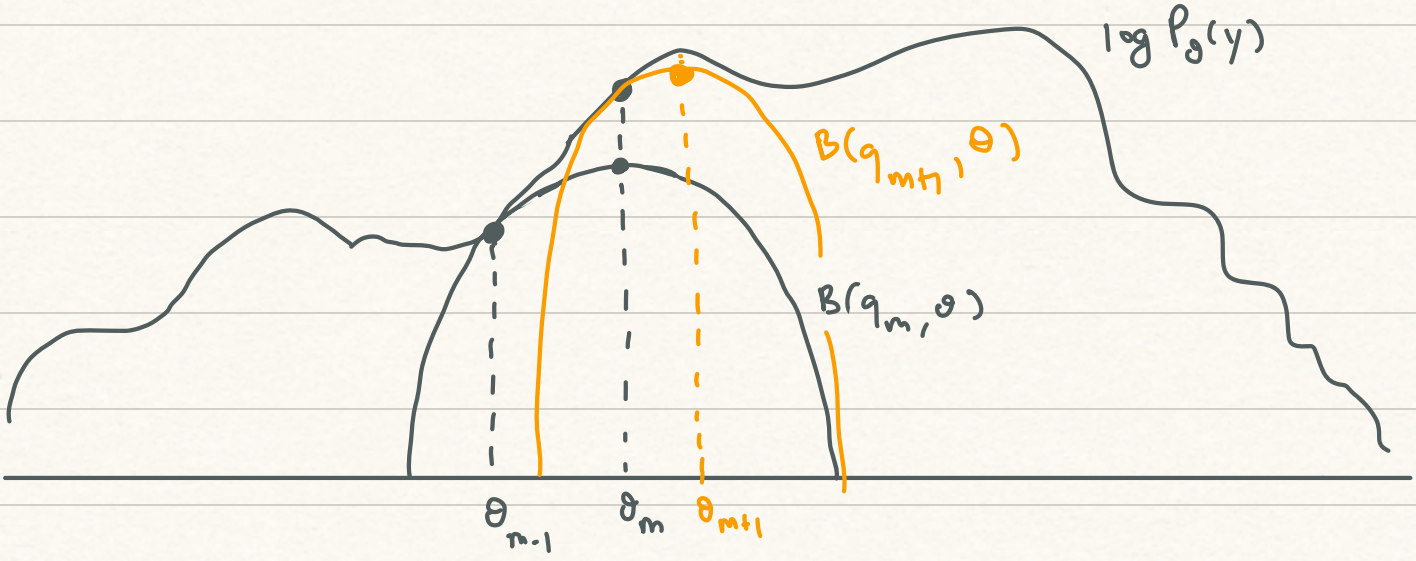
(E-step) $q_m = \underset{q}{\operatorname{argmax}}\ B(q, \vartheta_{m-1})$

(M-step) $\vartheta_m = \underset{\theta}{\operatorname{argmax}}\ B(q_m, \theta)$

return ϑ_m

- This will converge to a **local mode** of p(y).
- Random restarts are a good idea (to help find better local modes).
- The ELBO is **tight after the E-step** and **always increases.**

EM as a minorize-maximize algorithm



"**minorize**": to construct a surrogate function that touches the function-to-maximize at the current point, and lies below it everywhere else.

"**maximize**": to find a new point that maximizes the surrogate function.

E-step

$$q_m = \underset{q}{\operatorname{argmax}}\ B(q, \theta_{m-1})$$

$$= \underset{q}{\operatorname{argmax}}\ -KL(q(z) \,\|\, p_{\theta_{m-1}}(z|y)) + p_{\theta_{m-1}}(y)$$

$$= p_{\theta_{m-1}}(z|y)$$

M-step

$$\theta_m = \underset{\theta}{\operatorname{argmax}}\ B(q_m, \theta)$$

$$= \underset{\theta}{\operatorname{argmax}}\ \mathbb{E}_{q_m(z)}[\log p_\theta(y,z)] - \mathbb{E}_{q_m(z)}[q_m(z)]$$

$$= \underset{\theta}{\operatorname{argmax}}\ \mathbb{E}_{p_{\theta_{m-1}}(z|y)}[\log p_\theta(y,z)]$$

Plugging in the E-step, a more compact way to write the algorithm is as follows:

until convergence:

$$\theta_{new} = \underset{\theta}{\operatorname{argmax}}\ T(\theta; \theta^{old}) \equiv \underset{\theta}{\operatorname{argmax}}\ \mathbb{E}_{p_{\theta_{old}}(z|y)}[\log p_\theta(y,z)]$$

Where now we have something resembling the transition operator in MCMC (i.e., an objective function that depends on the current iterate.

EM for the Gaussian mixture model

$$P_\theta(y, z) = \prod_{i=1}^{n} \prod_{j=1}^{k} \left[\mathcal{N}(y_i; \mu_j, I\sigma_0^2)\, \pi_j \right]^{\mathbb{1}(z_i = j)}, \quad \theta = \{\mu_{1:k}, \pi\}$$

E-step: $q_m(z) = p_{\theta_{m-1}}(z \mid y) = \prod_{i=1}^{n} p_{\theta_{m-1}}(z_i \mid y)$

$$p_\theta(z_i = j \mid y) \equiv r_{ij} = \frac{\mathcal{N}(y_i; \mu_j, 1)\pi_j}{\sum_{j'} \mathcal{N}(y_i; \mu_{j'}, 1)\pi_{j'}}$$

M-step:

$$\operatorname*{argmax}_\theta \; \mathbb{E}_{q_m}\left[\log p_\theta(y, z)\right]$$

$$= \operatorname*{argmax}_{\mu_{1:k}, \pi} \; \mathbb{E}_{q_m}\left[\sum_{i=1}^{n}\sum_{j=1}^{k} z_{ij} \log\left(\mathcal{N}(y_i; \mu_j, 1)\pi_j\right)\right]$$

$$= \operatorname*{argmax}_{\mu_{1:k}, \pi} \; \sum_i \sum_j \mathbb{E}_{q_m}[z_{ij}] \log \mathcal{N}(y_i; \mu_j, 1) + \sum_i \sum_j \mathbb{E}_{q_m}[z_{ij}] \log \pi_j$$

This breaks into independent optimization problems:

$$= \operatorname*{argmax}_{\mu_{1:k}} -\sum_i \sum_j r_{ij}^m (y_i - \mu_j)^2, \quad \operatorname*{argmax}_{\pi} \sum_i \sum_j r_{ij}^m \log \pi_j$$

The optimization problem for the means further breaks into K independent ones:

$$= \operatorname*{argmax}_{\mu_j} -\sum_i r_{ij}^m (y_i - \mu_j)^2 \quad \forall j$$

This is just a weighted least-squares problem:

$$\mu_j^m = \frac{\sum_i r_{ij}^m y_i}{\sum_i r_{ij}^m} \quad \forall j$$

Meanwhile, this is just a cross-entropy loss problem:

$$\pi_j^m = \frac{1}{n}\sum_{i=1}^{n} r_{ij}^m$$

EM for the expfam mixture model

$$z_i \sim \text{Cat}(\pi)$$

$$y_i | z_i = j \sim g(\eta_j)$$

Now say the class-conditional likelihood is any exponential family...

$$g(y_i | \eta_j) = h(y_i) \exp(\eta_j^T t(y_i) - a(\eta_j))$$

M-step:

$$\underset{\eta, \pi}{\text{argmax}} \left[\mathbb{E}_{q(z)} [\log p(y | z)] + \mathbb{E}_{q(z)} [p(z)] \right]$$

Again, the M-step breaks into separate optimization problems, and the problem for pi is the same as before, so we only focus on estimating the natural parameters of the likelihood:

$$\underset{\eta}{\text{argmax}} \; \mathbb{E}_{q(z)} \left[\sum_i \sum_j \gamma_{ij} (\eta_j^T t(y_i) - a(\eta_j)) \right]$$

These again separate into independent optimization. Each one looks like this:

$$\underset{\eta_j}{\text{argmax}} \left[\eta_j^T \left(\sum_i r_{ij} \, t(y_i) \right) - \left(\sum_i r_{ij} \right) a(\eta_j) \right]$$

Taking the gradient of the objective and setting it to zero...

$$\nabla_{\eta_j} \; " \quad " = 0$$

$$\hookrightarrow \frac{\sum_i r_{ij} t(y_i)}{\sum_i r_{ij}} = \nabla_{\eta_j} a(\eta_j)$$

Note that for any (minimal) exponential family, we have the fact that the mapping from the natural parameter to the mean parameter is exactly the gradient of the log normalizer:

$$\text{For } g(y | \eta): \quad \mu(\eta) = \mathbb{E}[t(y)] = \nabla_{\eta_j} a(\eta_j)$$

Therefore the maximizer is the weighted average of the sufficient statistics passed through the inverse of the mean map:

$$\hookrightarrow \eta_j = \mu^{-1} \left(\frac{\sum_i r_{ij} t(y_i)}{\sum_i r_{ij}} \right)$$

EM for MAP estimation

Let's now put priors on the parameters and do maximum-a-posteriori (MAP) estimation:

$$\vartheta \leftarrow \operatorname{argmax} P(y, \theta)$$

The ELBO now becomes:

$$B(q, \theta) = \mathbb{E}_q \left[\log \frac{P(y, z, \theta)}{q(z)} \right]$$

Note that the q distribution is still only over the latent variables Z:

$$= \mathbb{E}_q \left[\log \frac{P(y, z \mid \theta)}{q(z)} \right] + \log P(\theta)$$

The M-step becomes:

$$\vartheta \leftarrow \operatorname{argmax} \mathbb{E}_q \left[\log P(y, z \mid \theta) \right] + \log P(\theta)$$

For example, let's add the following priors to the Gaussian mixture model:

$$\text{e.g.,} \quad \mu_j \sim \mathcal{N}(m_0, \lambda_0^{-1}) \qquad \pi \sim \mathrm{Dir}(\alpha)$$

The optimization problem for one of the means is just the same as before, but now adding the log prior to the objective function:

$$\mu_j \leftarrow \operatorname*{argmax}_{\mu_j} - \sum_i r_{ij} (\mu_j - y_i)^2 + \log \mathcal{N}(\mu_j; m_0, \lambda_0^{-1})$$

$$\mu_j \leftarrow \operatorname*{argmax}_{\mu_j} - \sum_i r_{ij} (\mu_j - y_i)^2 - \lambda_0 (\mu_j - m_0)^2$$

This is still just a weighted least squares problem:

$$\mu_j \leftarrow \frac{m_0 + \sum_i r_{ij} y_i}{\lambda_0 + \sum_i r_{ij}}$$

Now consider the optimization problem for Pi; the objective function is the same as the one before but now adding the log Dirichlet prior:

$$\pi \leftarrow \underset{\pi}{\operatorname{argmax}} \sum_i \sum_j r_{ij} \log \pi_j + \log \mathrm{Dir}(\pi, \alpha)$$

$$\pi \leftarrow \underset{\pi}{\operatorname{argmax}} \sum_i \sum_j r_{ij} \log \pi_j + \sum_j (\alpha_j - 1) \log \pi_j$$

You can solve this using Lagrange multipliers (a good exercise). The solution is:

$$\pi_j \leftarrow \frac{\alpha_j + \sum_i r_{ij} - 1}{\sum_{j'} (\alpha_{j'} + \sum_i r_{ij} - 1)}$$

Last time, we derived the complete conditionals for both of these parameters with the same priors. The complete conditionals were:

$$P(\mu_j \mid -) = \mathcal{N}(\mu_j; m_j, \lambda_j^{-1})$$

$$m_j = \lambda_j^{-1}(m_0 + \sum_i f_{ij} y_i)$$

$$\lambda_j = \lambda_0 + \sum_i f_{ij}$$

$$P(\pi \mid -) = \mathrm{Dir}(\pi; \vec{\alpha}_n)$$

$$\alpha_{n,j} = \alpha_j + \sum_i f_{ij}$$

The M-step update for the Gaussian looks exactly like the mean of its complete conditional, except that the indicators have been replaced with expectations of the indicators (under q(Z)).

Similarly, the M-step update for Pi is exactly the mode of the Dirichlet complete conditional, except again the indicators have been replaced with expectations of indicators.

This is not a coincidence!

EM and exponential families: a match made in heaven

An alternative way to go about deriving the M-step update is as follows. First note that:

$$E_q[\log P(Y, Z, \theta)] \propto_\theta E_q[\log P(\theta \mid Y, Z)]$$

And further note that:

$$\operatorname*{argmax}_\theta E_q[\log \dots] = \operatorname*{argmax}_\theta \exp\left(E_q[\log \dots]\right)$$

Therefore:

$$\operatorname*{argmax}_\theta E_q[\log P(Y, Z, \theta)] = \operatorname*{argmax}_\theta \exp\left(E_q[\log P(\theta \mid -)]\right)$$

In our model (and in many) the parameters are conditionally independent in their posterior, so this breaks apart into separate optimization problems (as before). If we have all conditionally conjugate priors, then the complete conditional for every parameter can be written as:

$$P(\eta \mid -) = f(\eta \mid \lambda_n) \propto h(\eta) \exp(\lambda_n^\top t(\eta))$$

where f(...) is an exponential family, and lambda_n is its posterior natural parameter, which is a function of Z and Y.

$$\operatorname*{argmax}_\eta \exp\left(E_q[\log h(\eta) \exp(\lambda_n^\top t(\eta))]\right)$$

$$= \operatorname*{argmax}_\eta \exp\left(\log h(\eta) + E_q[\lambda_n]^\top t(\eta)\right)$$

$$= \operatorname*{argmax}_\eta h(\eta) \exp\left(E_q[\lambda_n]^\top t(\eta)\right)$$

$$= \operatorname*{argmax}_\eta f(\eta \mid E_q[\lambda_n])$$

And so the optimal update to eta is the mode of an exponential family f(...), which is the same as the complete conditional except replacing the natural parameter with its expectation under q(Z)!