Approximate Inference for Neural Networks

Mauricio A. Álvarez PhD,
H.F. Garcia C. Guarnizo (TA)

Universidad Tecnológica de Pereira, Pereira, Colombia

Outline

1. EM algorithm
2. Variational EM
3. Approximate inference for Neural Networks

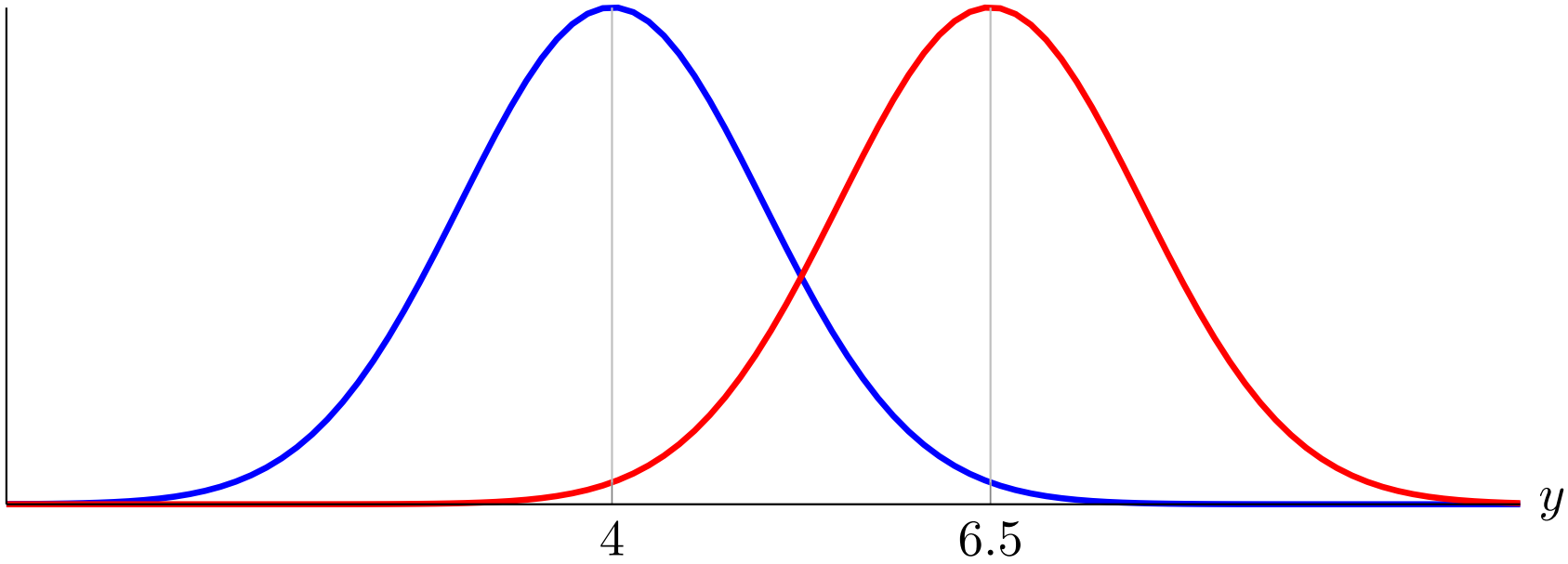
Outline

1 **EM algorithm**

2 Variational EM

3 Approximate inference for Neural Networks

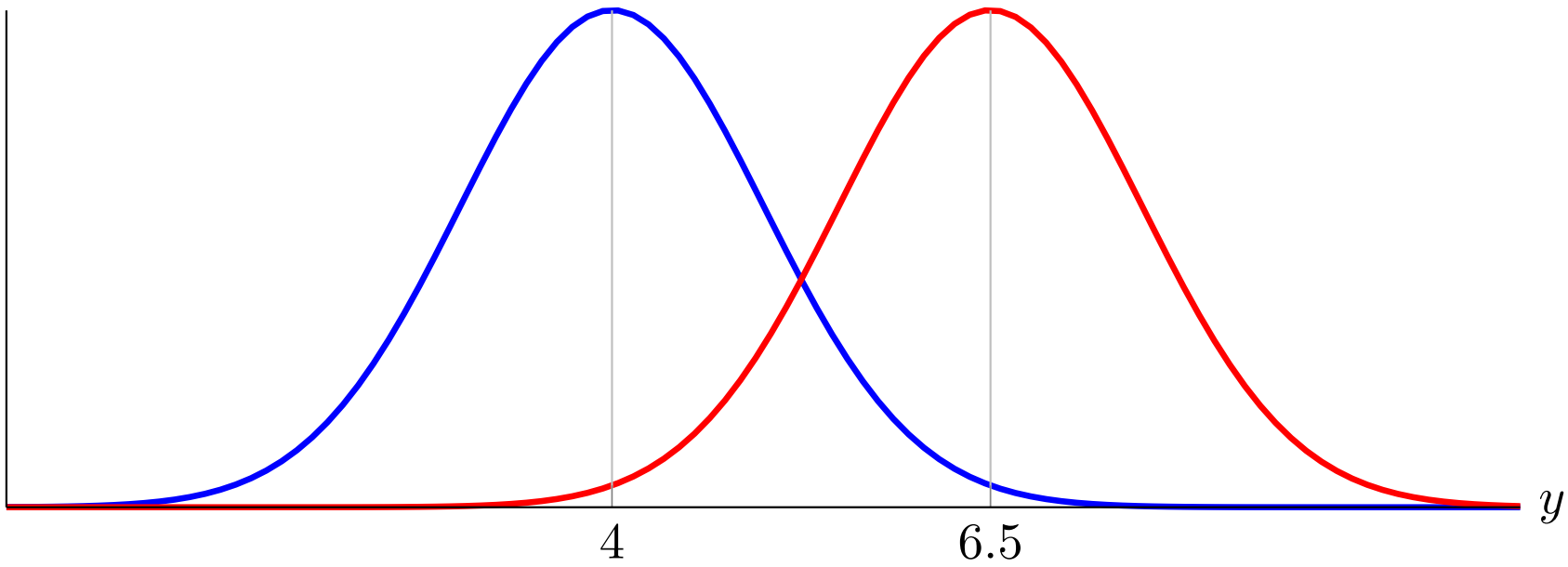
Problem definition - Example



z_n

y_n

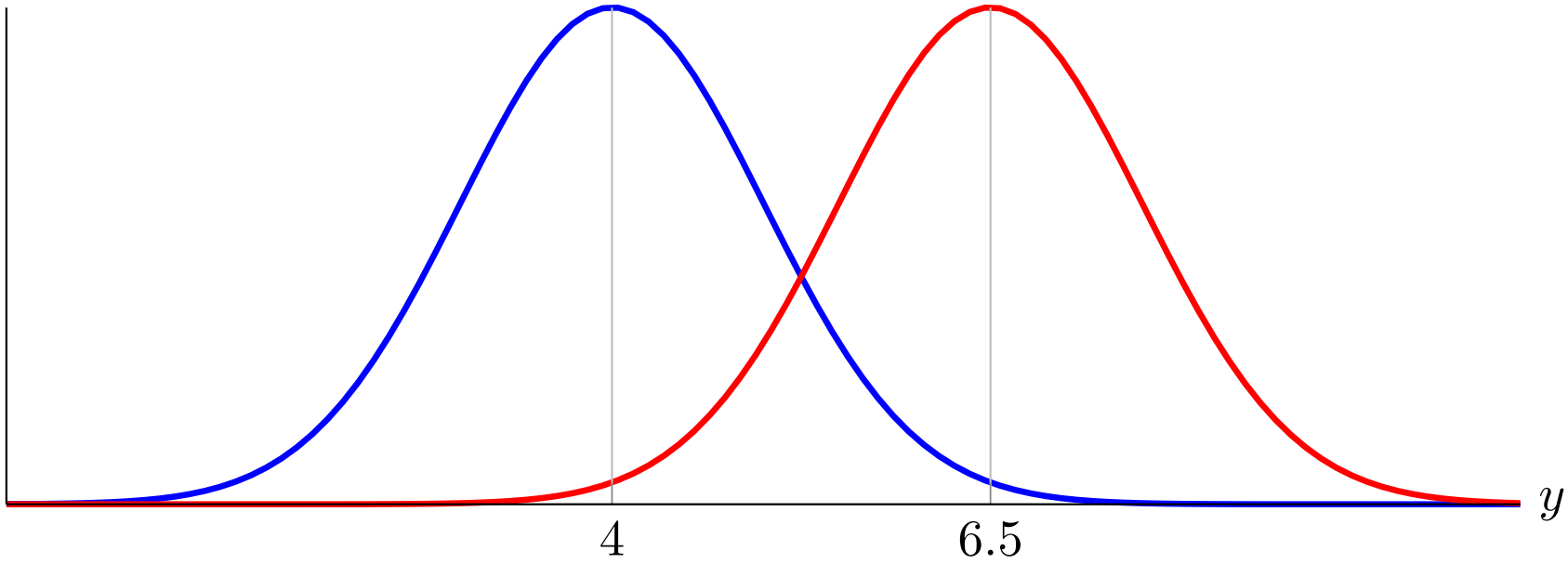
Problem definition - Example



z_n 0

y_n

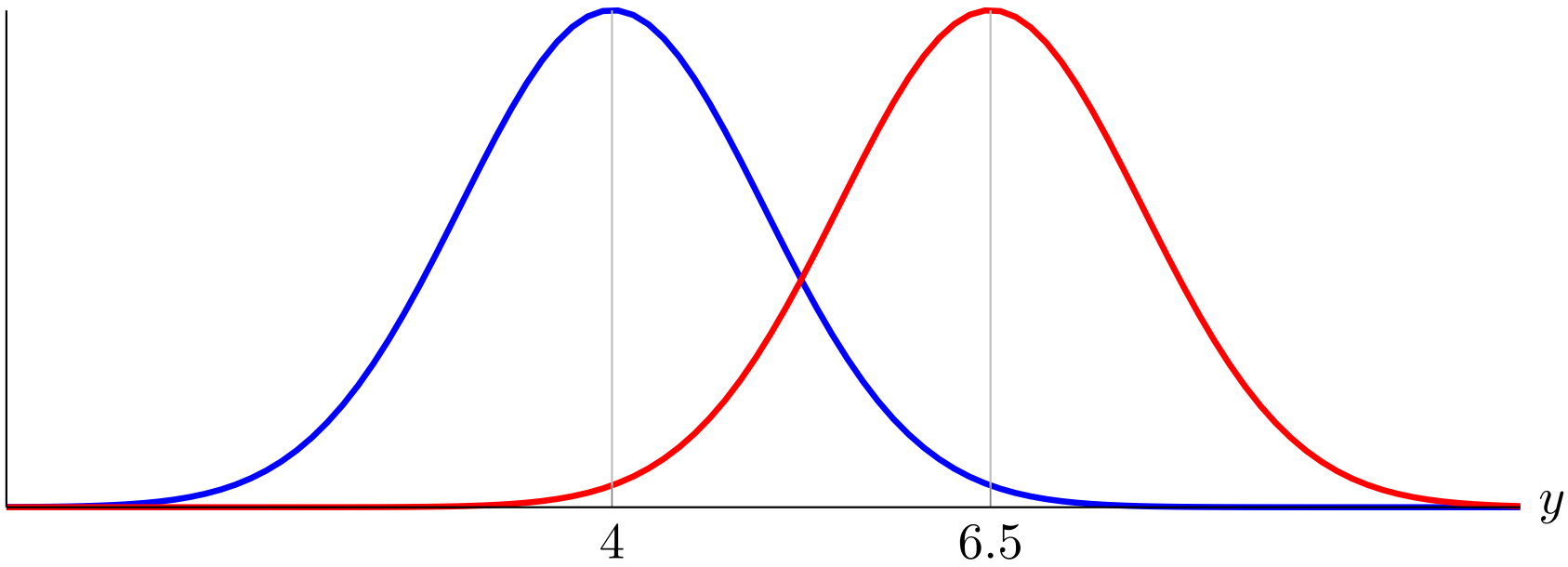
Problem definition - Example



$z_n \quad 0$

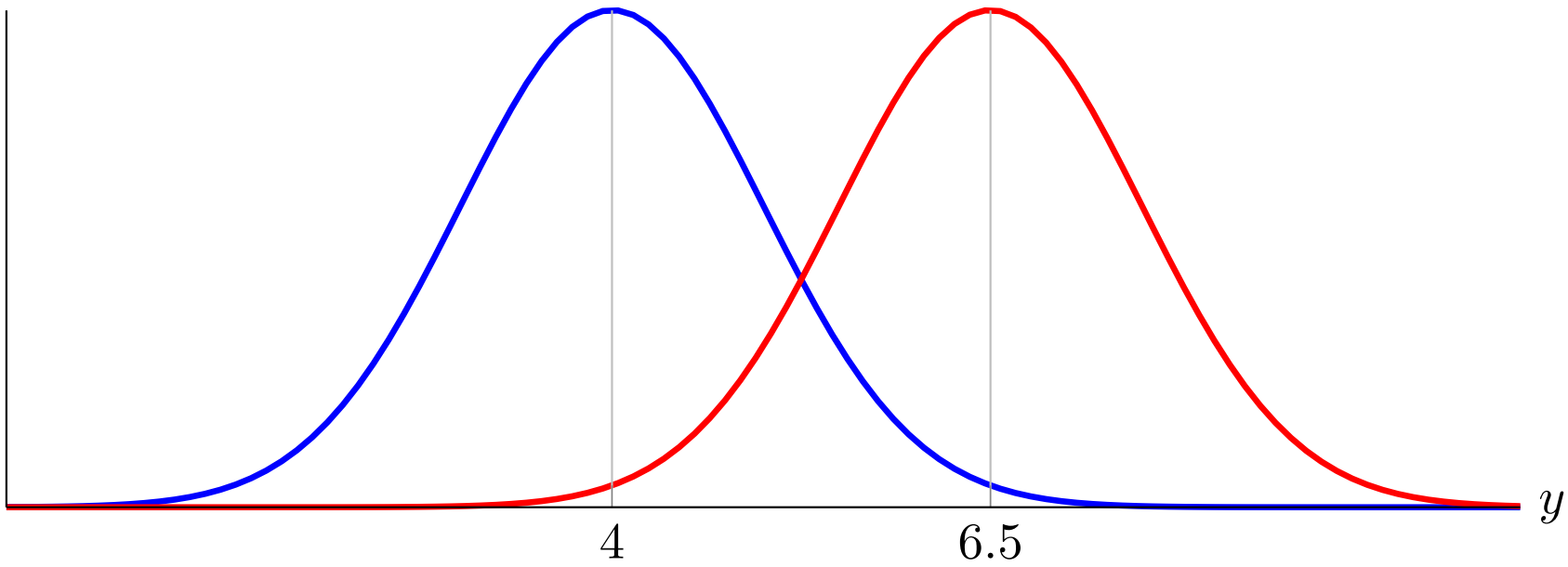
$y_n \quad 4.3$

Problem definition - Example



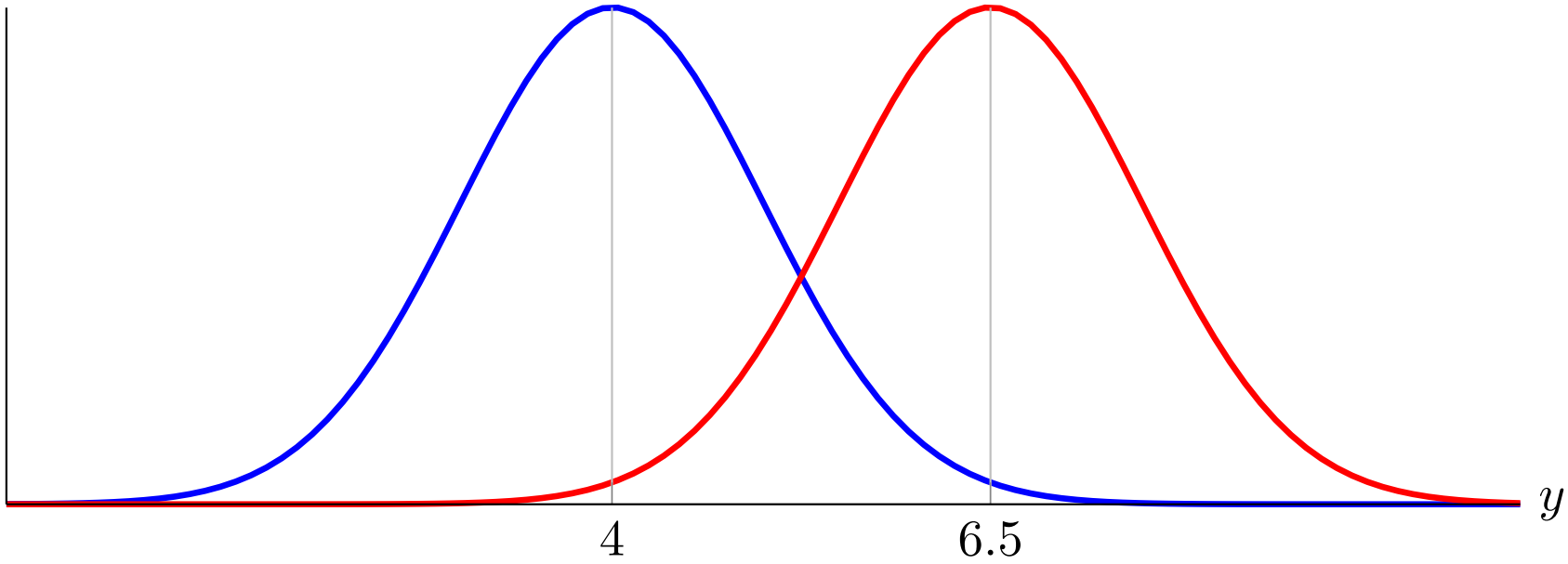
z_n	0	1
y_n	4.3	

Problem definition - Example



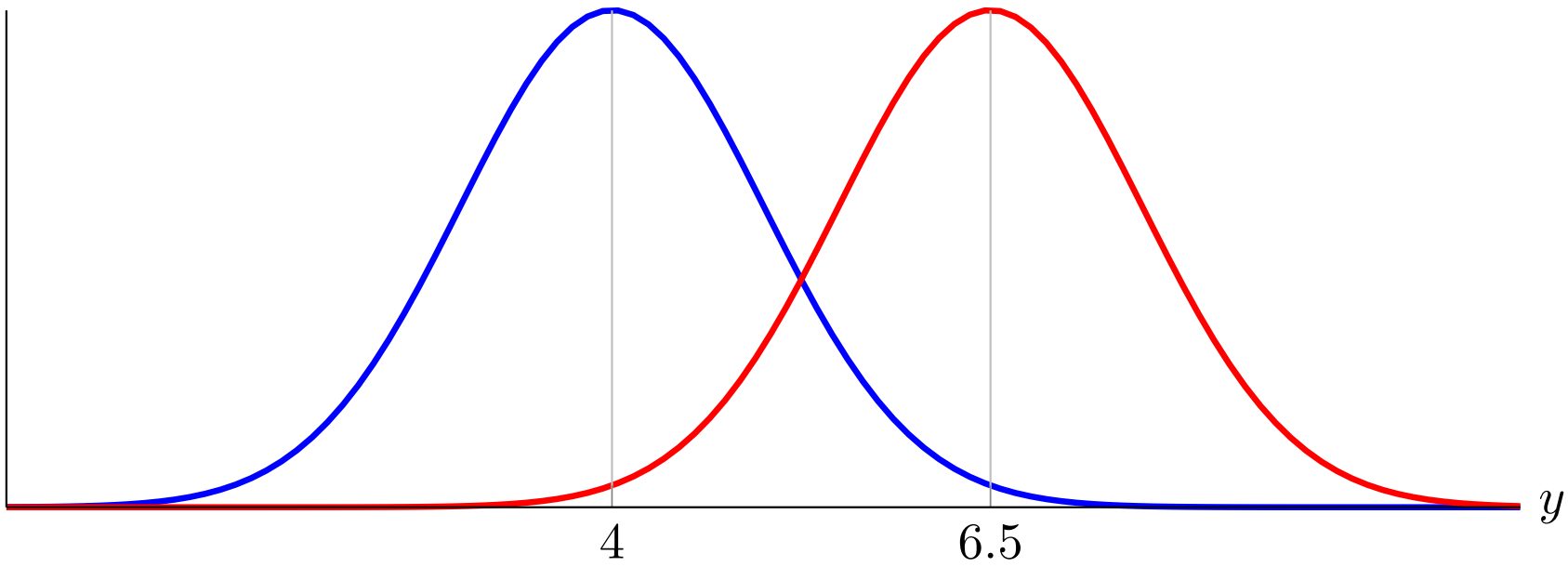
z_n	0	1
y_n	4.3	6.0

Problem definition - Example



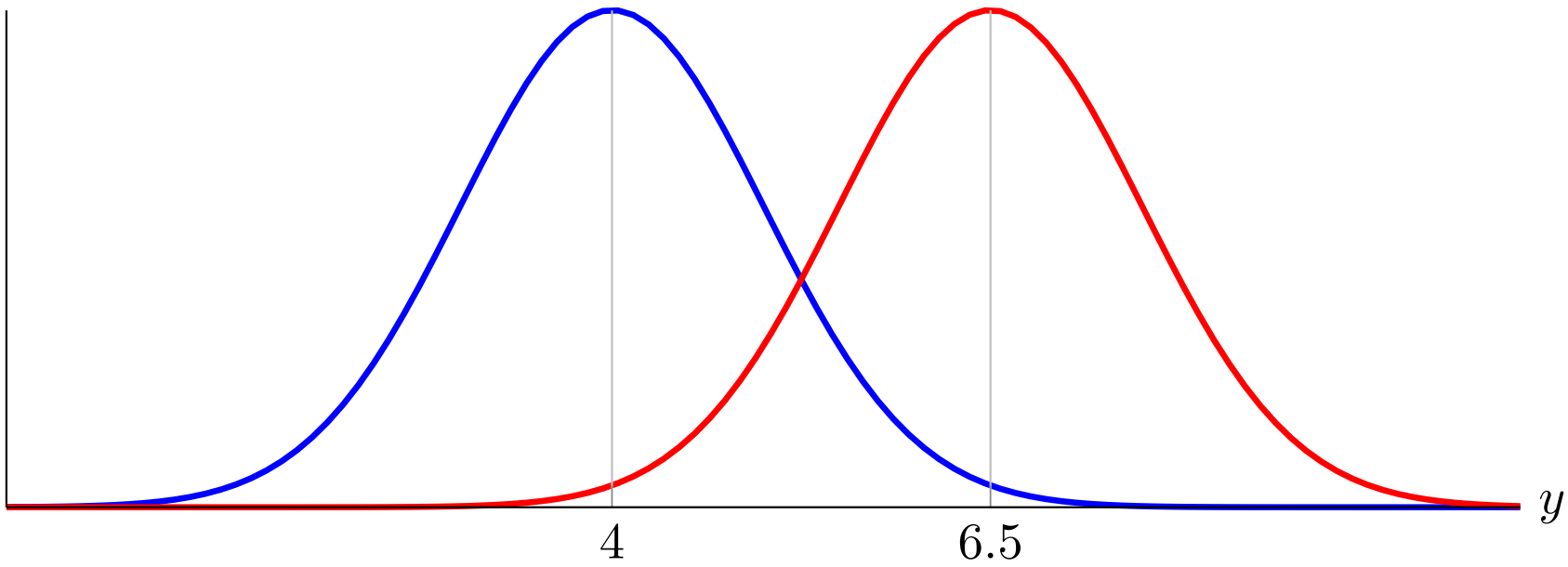
z_n	0	1	1
y_n	4.3	6.0	

Problem definition - Example



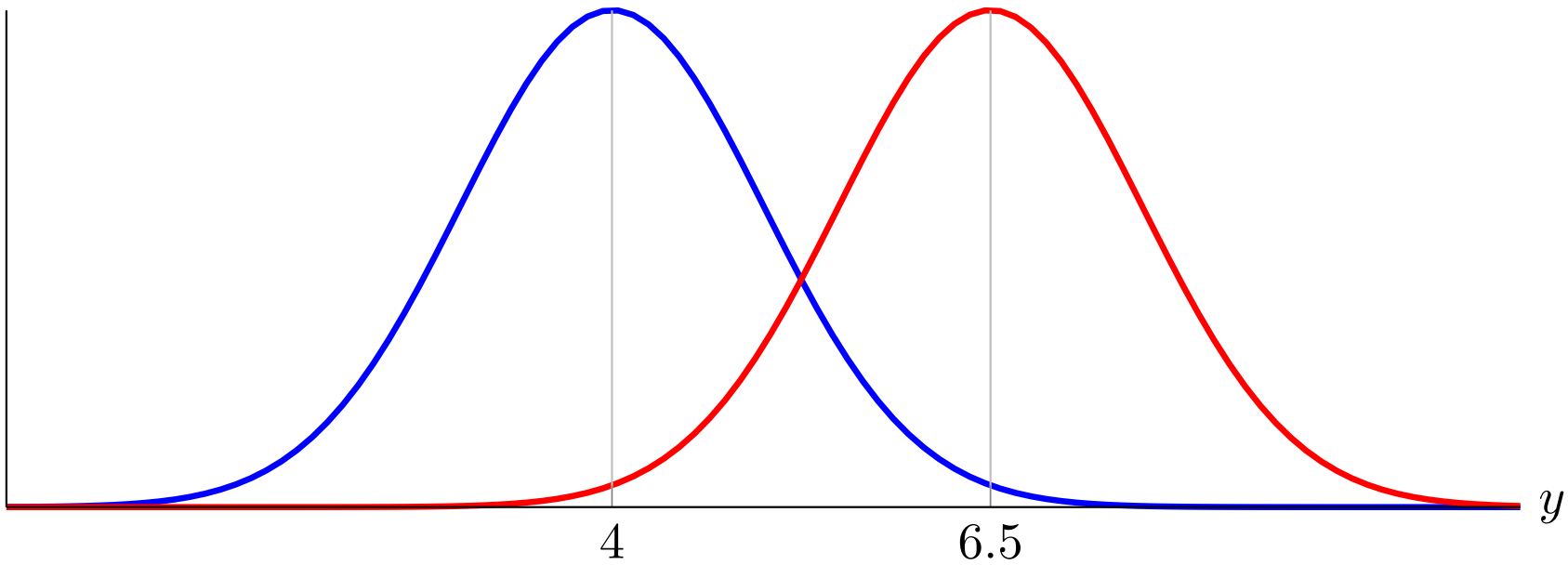
z_n	0	1	1
y_n	4.3	6.0	7.2

Problem definition - Example



z_n	0	1	1	0
y_n	4.3	6.0	7.2	

Problem definition - Example



z_n	0	1	1	0
y_n	4.3	6.0	7.2	3.8

Problem definition - Example

For z:

$$p(z) = \pi^z (1-\pi)^{1-z}, \quad p(z=1) = \pi, \quad p(z=0) = 1-\pi.$$

For y:

$$p(y|z) = \left(\mathcal{N}\left(y|\mu_1, \sigma_1^2\right)\right)^z \left(\mathcal{N}\left(y|\mu_2, \sigma_2^2\right)\right)^{1-z}.$$

The the joint probability:

$$p(y,z) = \left(\pi\mathcal{N}\left(y|\mu_1, \sigma_1^2\right)\right)^z \left((1-\pi)\mathcal{N}\left(y|\mu_2, \sigma_2^2\right)\right)^{1-z}.$$

Additionally:

$$\begin{aligned} p(z=1|y) &= \frac{p(y|z=1)p(z=1)}{p(y)} \\ &= \frac{\mathcal{N}(y|\mu_1, \sigma_1^2)\pi}{\pi\mathcal{N}\left(y|\mu_1, \sigma_1^2\right) + (1-\pi)\mathcal{N}\left(y|\mu_2, \sigma_2^2\right)} \end{aligned}$$

Problem definition - Example

Let's assume that we have $(\mathbf{y}, \mathbf{z}) = \{y_n, z_n\}_{n=1}^{N}$.
The parameters $\theta = \{\pi, \mu_1, \mu_2, \sigma_1^2, \sigma_2^2\}$, can be estimated as

$$\begin{aligned}
\theta_{\text{ML}} &= \arg\max_{\theta} \ \ln\left(p(\mathbf{y}, \mathbf{z}|\theta)\right), \\
&= \arg\max_{\theta} \ \ln\left(\prod_{n=1}^{N} p(y_n, z_n|\theta)\right), \\
&= \arg\max_{\theta} \sum_{n=1}^{N} \ln\left(p(y_n, z_n|\theta)\right).
\end{aligned}$$

Where:

$$\begin{aligned}
\ln p(y_n, z_n|\theta) =& z_n \left(\ln(\pi) + \ln \mathcal{N}\left(y_n|\mu_1, \sigma_1^2\right)\right) \\
&+ (1 - z_n)\left(\ln(1-\pi) + \ln \mathcal{N}\left(y_n|\mu_2, \sigma_2^2\right)\right).
\end{aligned}$$

Problem definition - Example

Let's assume that $\mathbf{y} = \{y_n\}_{n=1}^N$, y $\mathbf{z}$ is unknown (latent).
The parameters $\theta = \{\pi, \mu_1, \mu_2, \sigma_1^2, \sigma_2^2\}$, can be estimated as

$$\begin{aligned}\theta_{\text{ML}} &= \arg\max_{\theta} \sum_{n=1}^{N} \ln\left(p(y_n|\theta)\right), \\ &= \arg\max_{\theta} \sum_{n=1}^{N} \ln\left(\sum_{z} p(y_n, z_n|\theta)\right).\end{aligned}$$

Where:

$$\ln\left(\sum_{z} p(y_n, z_n|\theta)\right) = \ln\left(\pi\mathcal{N}\left(y_n|\mu_1, \sigma_1^2\right) + (1-\pi)\mathcal{N}\left(y_n|\mu_2, \sigma_2^2\right)\right).$$

Problem definition - General

The model consists of observations $\mathbf{y}$ and a latent random variable $\mathbf{z}$. Then

$$p(\mathbf{y}|\theta) = \prod_{n=1}^{N} p(y_n|\theta)$$
$$= \prod_{n=1}^{N} \sum_{\mathbf{z}} p(y_n, \mathbf{z}|\theta) = \prod_{n=1}^{N} \sum_{\mathbf{z}} p(y_n|\mathbf{z}, \theta) p(\mathbf{z}|\theta)$$

The estimation of θ_{ML} is given by

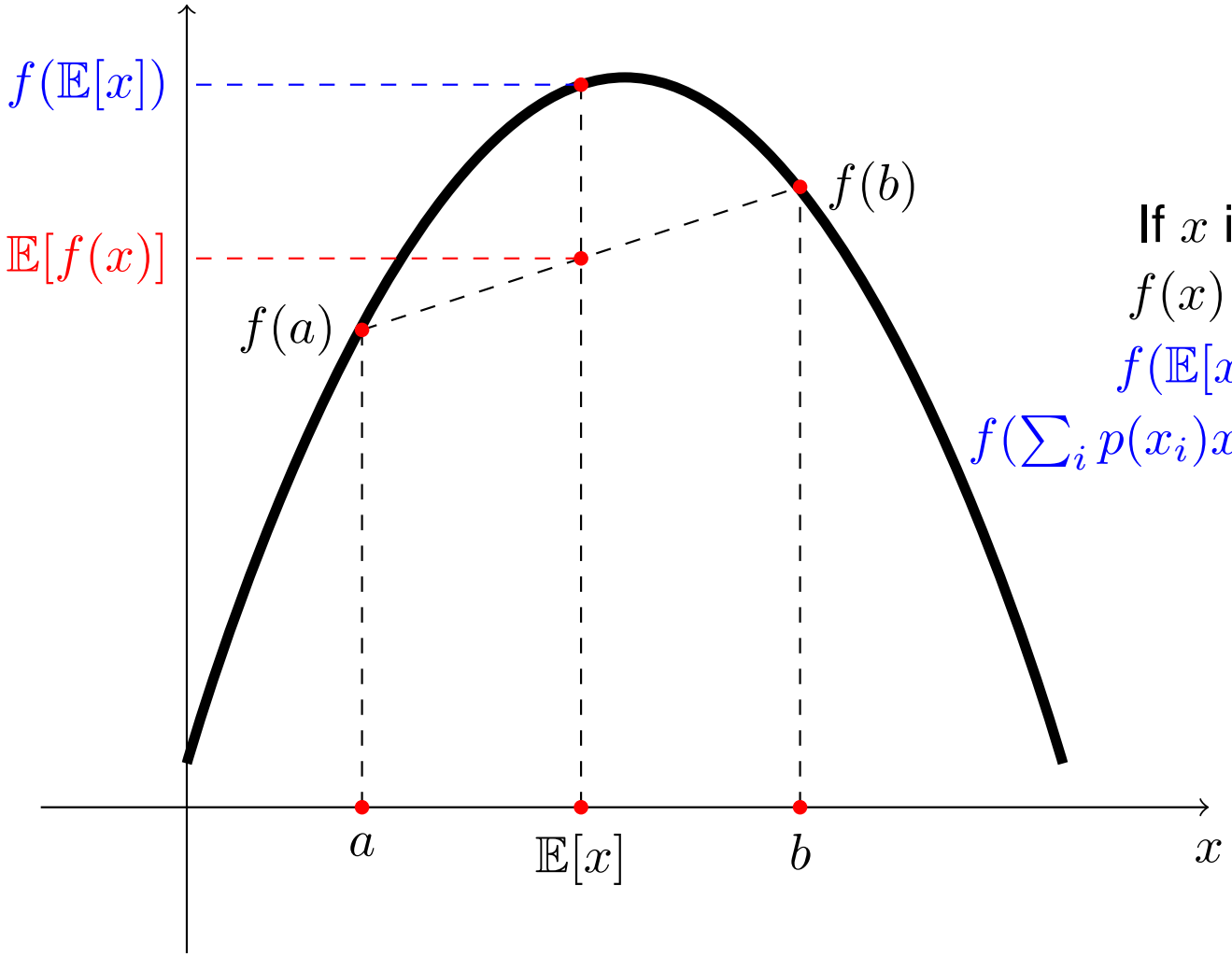
$$\theta_{\text{ML}} = \arg\max_{\theta} \ln\left(p(\mathbf{y}|\theta)\right)$$
$$= \arg\max_{\theta} \sum_{n=1}^{N} \ln\left(\sum_{\mathbf{z}} p(y_n|\mathbf{z}, \theta) p(\mathbf{z}|\theta)\right)$$

Problem definition

$$\theta_{\text{ML}} = \arg\max_{\theta} \sum_{n=1}^{N} \ln \left(\sum_{\mathbf{z}} p(y_n, |\mathbf{z}, \theta) p(\mathbf{z}|\theta) \right)$$

- The sum over $\mathbf{z}$ couples the parameters θ.
- Gradients have no closed form.
- $\mathbf{z}$ and θ are also coupled.

Jensen's Inequality



If x is a r.v. and $f(x)$ is concave:

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

$$f\left(\sum_i p(x_i)x_i\right) \geq \sum_i p(x_i)f(x_i)$$

EM algorithm - Jensen's Inequality

$$\log p(\mathbf{y}|\theta) = \sum_{n=1}^{N} \ln \left(\sum_{\mathbf{z}} p(y_n, \mathbf{z}|\theta) \right)$$

EM algorithm - Jensen's Inequality

$$\log p(\mathbf{y}|\theta) = \sum_{n=1}^{N} \ln \left(\sum_{\mathbf{z}} p(y_n, \mathbf{z}|\theta) \right)$$
$$= \sum_{n=1}^{N} \ln \left(\sum_{\mathbf{z}} q(\mathbf{z}) \frac{p(y_n, \mathbf{z}|\theta)}{q(\mathbf{z})} \right)$$

EM algorithm - Jensen's Inequality

$$
\begin{aligned}
\log p(\mathbf{y}|\theta) &= \sum_{n=1}^{N} \ln \left(\sum_{\mathbf{z}} p(y_n, \mathbf{z}|\theta) \right) \\
&= \sum_{n=1}^{N} \ln \left(\sum_{\mathbf{z}} q(\mathbf{z}) \frac{p(y_n, \mathbf{z}|\theta)}{q(\mathbf{z})} \right) \\
&\geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(y_n, \mathbf{z}|\theta)}{q(\mathbf{z})} \right)
\end{aligned}
$$

Distribution $q(\mathbf{z})$

Assuming a value of $\theta^{(t)}$, what form $q(\mathbf{z})$ should have to maximize

$$\ln p(\mathbf{y}|\theta^{(t)}) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)}) p(y_n|\theta^{(t)})}{q(\mathbf{z})} \right)$$

Distribution $q(\mathbf{z})$

Assuming a value of $\theta^{(t)}$, what form $q(\mathbf{z})$ should have to maximize

$$\ln p(\mathbf{y}|\theta^{(t)}) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)}) p(y_n|\theta^{(t)})}{q(\mathbf{z})} \right)$$

$$\geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \left[\ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)})}{q(\mathbf{z})} \right) + \ln p(y_n|\theta^{(t)}) \right]$$

Distribution $q(\mathbf{z})$

Assuming a value of $\theta^{(t)}$, what form $q(\mathbf{z})$ should have to maximize

$$\ln p(\mathbf{y}|\theta^{(t)}) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)}) p(y_n|\theta^{(t)})}{q(\mathbf{z})} \right)$$

$$\geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \left[\ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)})}{q(\mathbf{z})} \right) + \ln p(y_n|\theta^{(t)}) \right]$$

organizing,

$$\sum_{n=1}^{N} \ln p(y_n|\theta^{(t)}) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)})}{q(\mathbf{z})} \right) + \sum_{n=1}^{N} \ln p(y_n|\theta^{(t)}),$$

Distribution $q(\mathbf{z})$

Assuming a value of $\theta^{(t)}$, what form $q(\mathbf{z})$ should have to maximize

$$\ln p(\mathbf{y}|\theta^{(t)}) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)}) p(y_n|\theta^{(t)})}{q(\mathbf{z})} \right)$$

$$\geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \left[\ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)})}{q(\mathbf{z})} \right) + \ln p(y_n|\theta^{(t)}) \right]$$

organizing,

$$\sum_{n=1}^{N} \ln p(y_n|\theta^{(t)}) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|y_n, \theta^{(t)})}{q(\mathbf{z})} \right) + \sum_{n=1}^{N} \ln p(y_n|\theta^{(t)}),$$

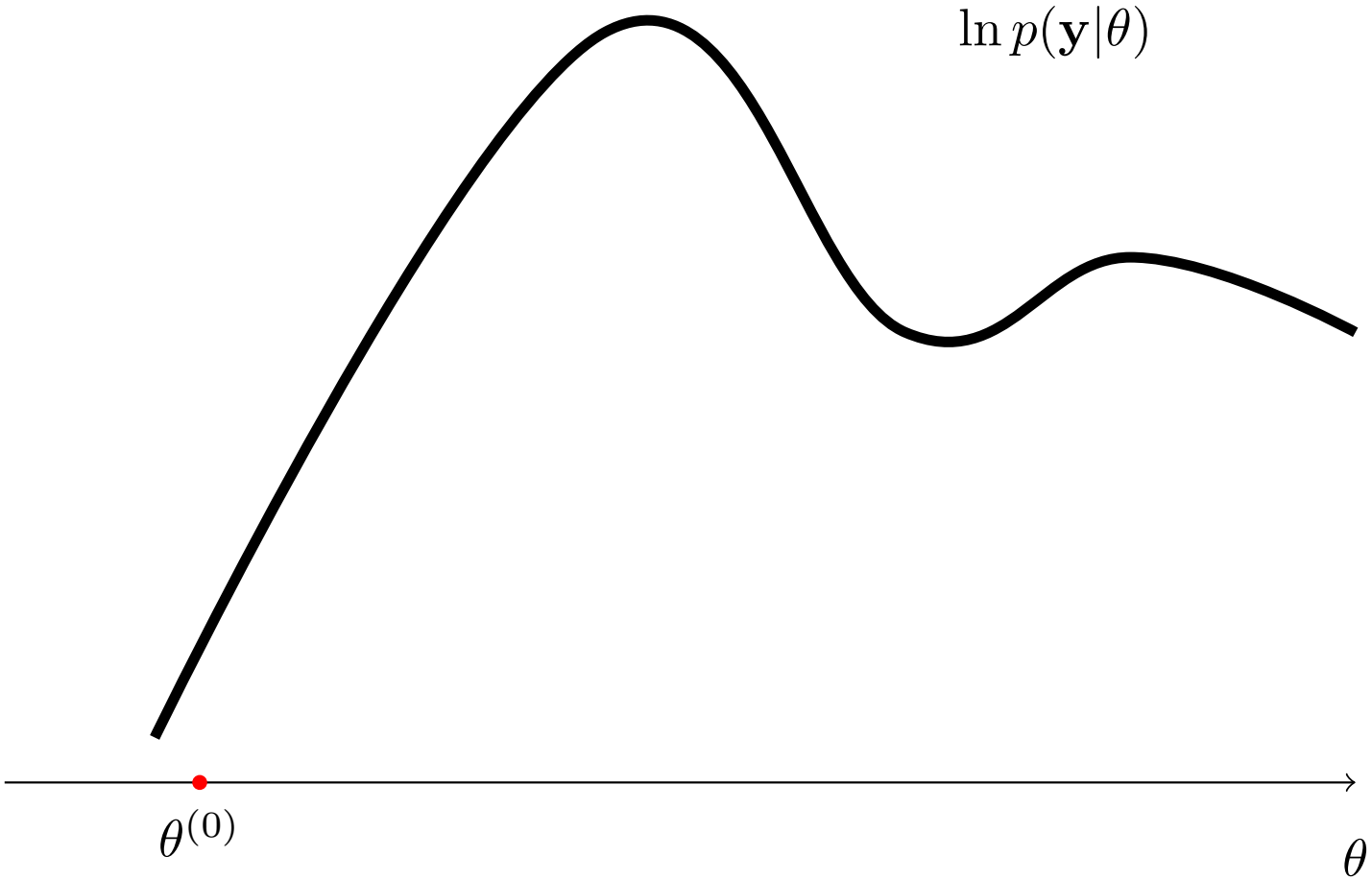
then, $q(\mathbf{z}) = p(\mathbf{z}|y_n, \theta^{(t)})$.

Parameter estimation

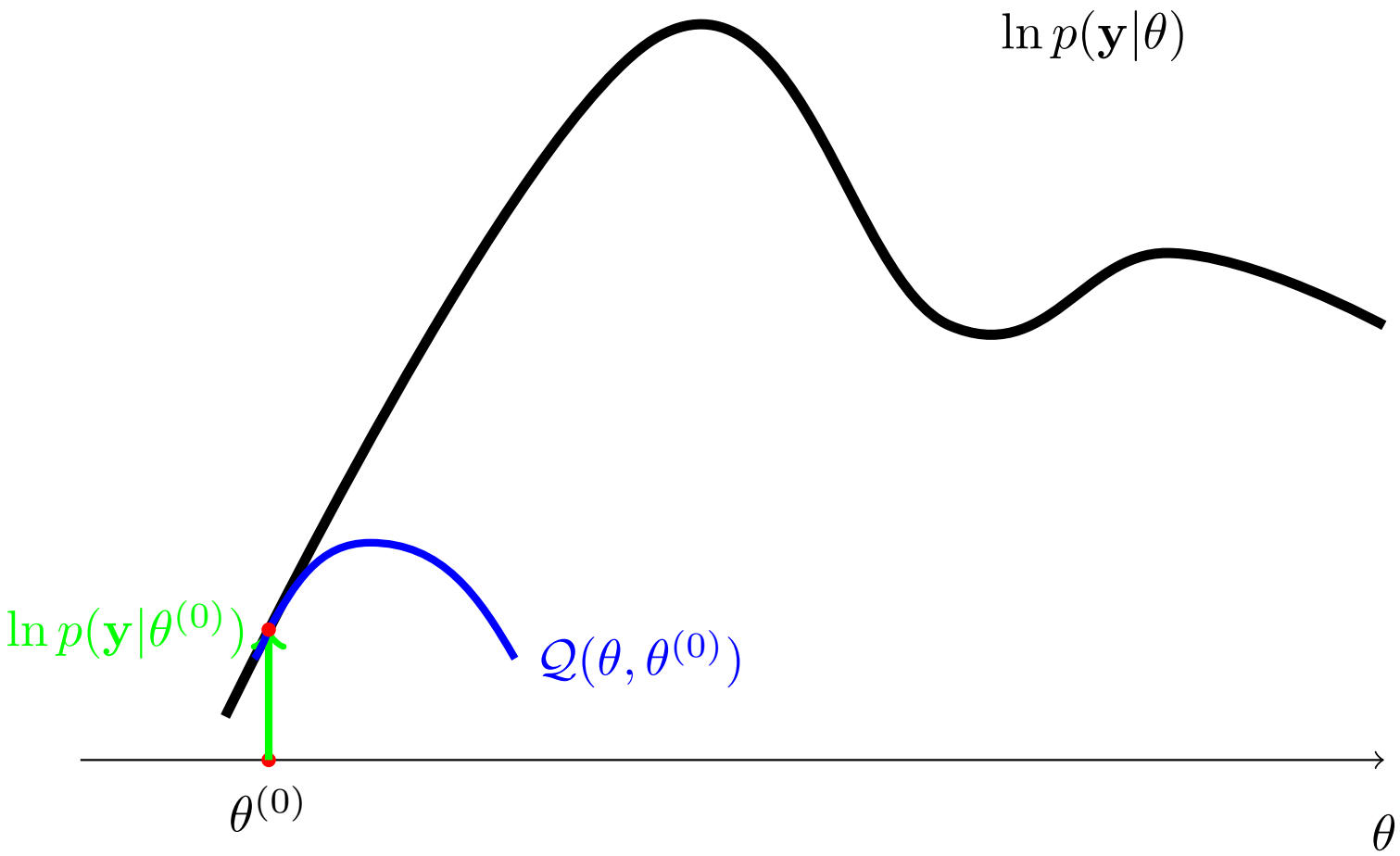
If $q(\mathbf{z}) = p(\mathbf{z}|y_n, \theta^{(t)})$,

$$\begin{aligned}
\theta^{(t+1)} &= \arg\max_{\theta} \sum_{n=1}^{N} \sum_{\mathbf{z}} p(\mathbf{z}|y_n, \theta^{(t)}) \ln \left(\frac{p(y_n, \mathbf{z}|\theta)}{p(\mathbf{z}|y_n, \theta^{(t)})} \right) \\
&= \arg\max_{\theta} \sum_{n=1}^{N} \mathbb{E}_{q(\mathbf{z})} \left[\ln \left(p(y_n, \mathbf{z}|\theta) \right) \right] \\
&= \arg\max_{\theta} \mathcal{Q}\left(\theta, \theta^{(t)} \right)
\end{aligned}$$

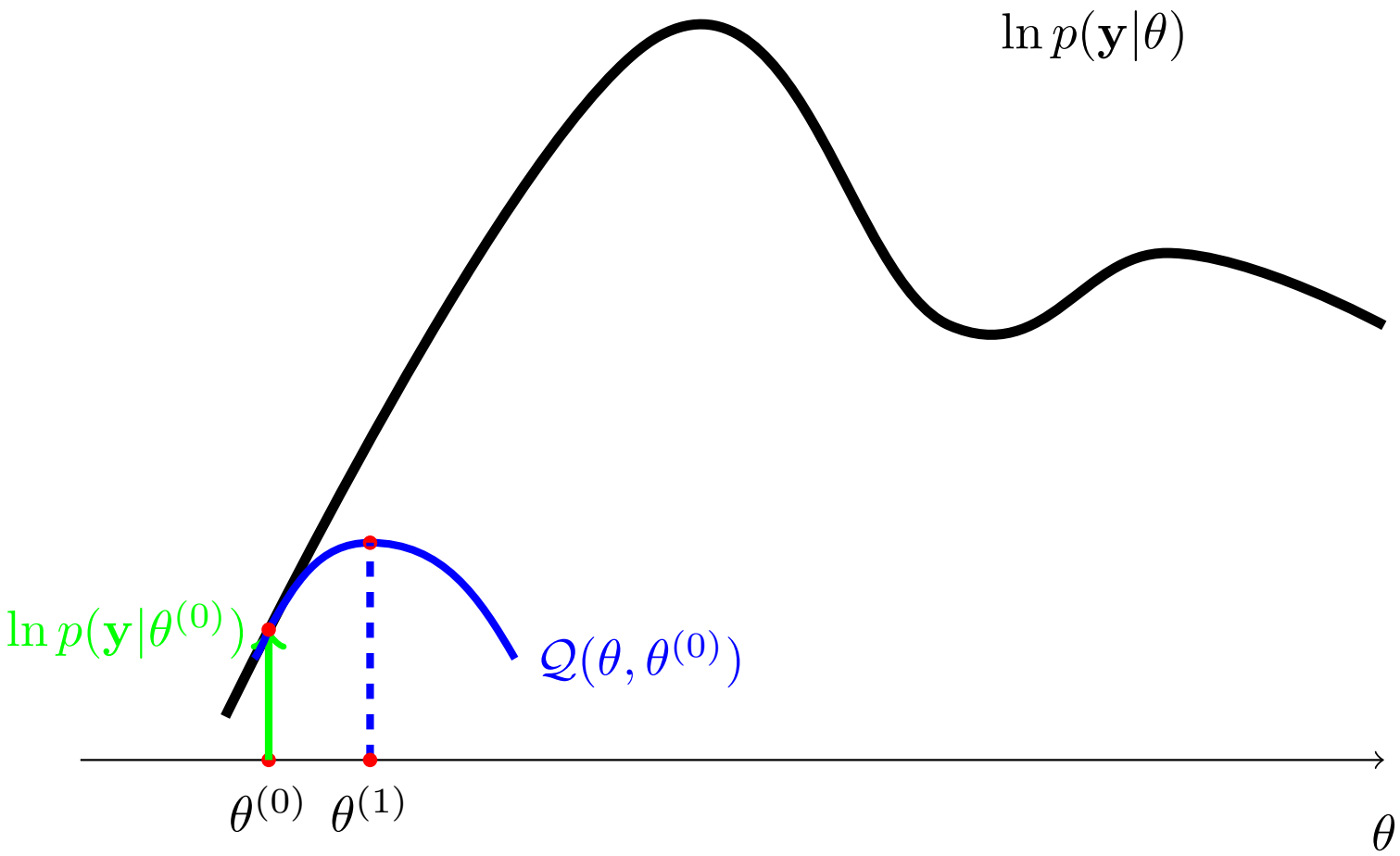
EM algorithm - Visual explanation



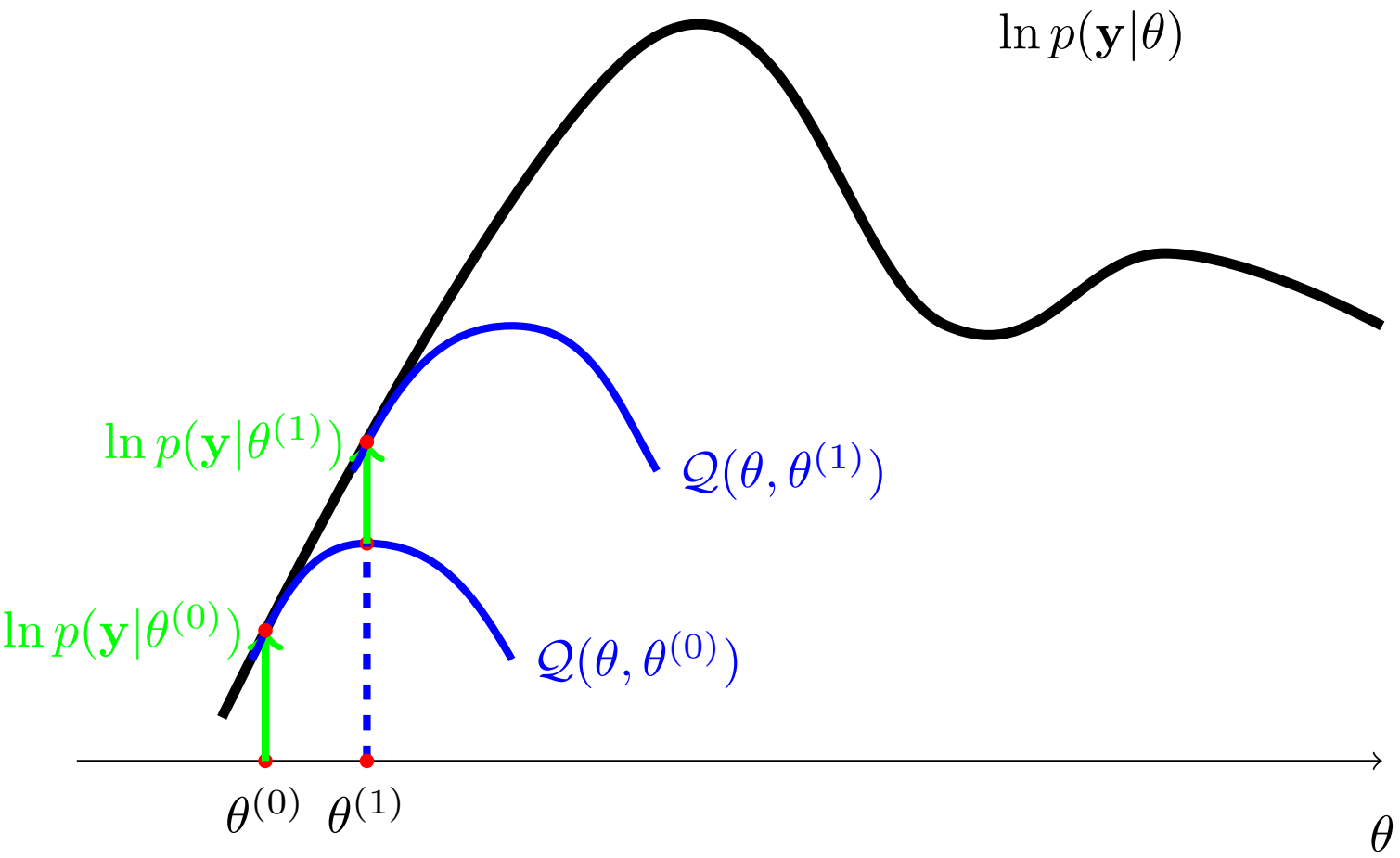
EM algorithm - Visual explanation



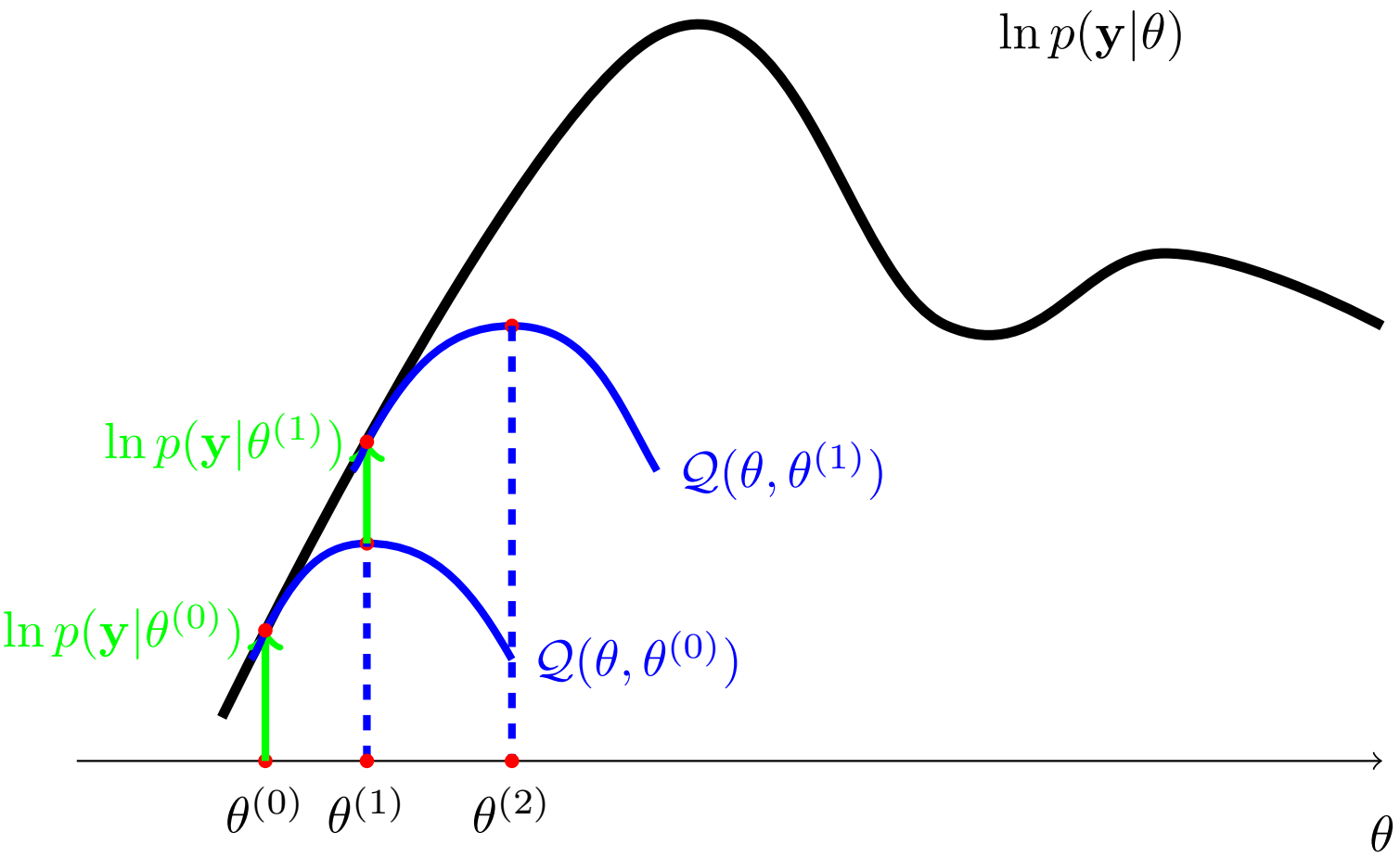
EM algorithm - Visual explanation



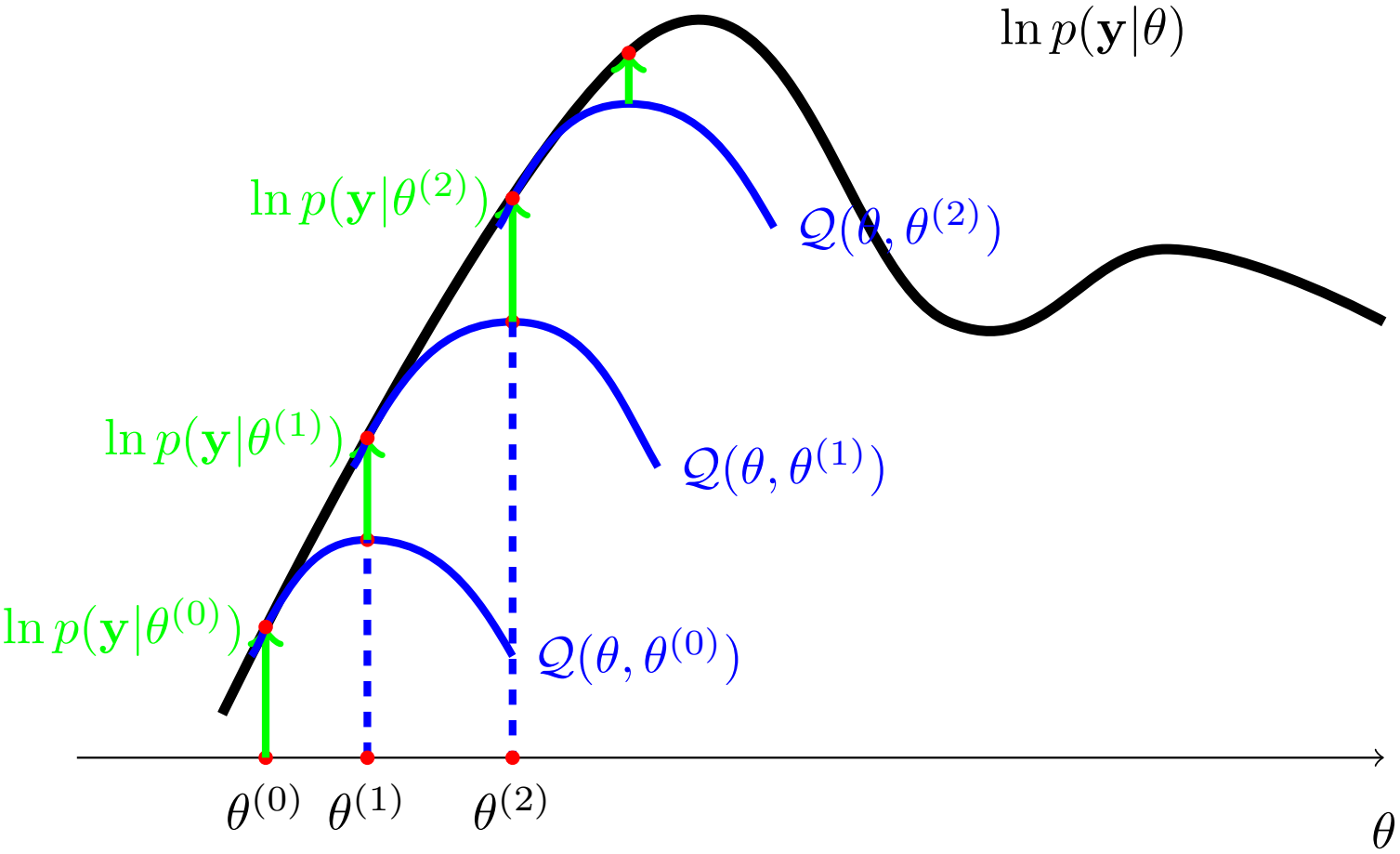
EM algorithm - Visual explanation



EM algorithm - Visual explanation



EM algorithm - Visual explanation



EM Algorithm - Pseudo-code

Given the joint distribution $p(\mathbf{y}, \mathbf{z}|\theta)$, the objective is to maximize $p(\mathbf{y}|\theta)$ w.r.t. θ.

1. Select the initial parameters $\theta^{(t)} \leftarrow \theta^{(0)}$.
2. Evaluate E-step, $q(\mathbf{z}) = p(\mathbf{z}|\mathbf{y}, \theta^{(t)})$.
3. Evaluate M-step,
$$\theta^{(\text{t+1})} = \arg\max_{\theta} \mathcal{Q}\left(\theta, \theta^{(t)}\right)$$
4. Verify convergence. If it doesn't satisfy, then
$$\theta^{(t)} \leftarrow \theta^{(t+1)},$$
return to step 2.

EM algorithm - Remark

$$\log p(\mathbf{y}|\theta) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(y_n, \mathbf{z}|\theta)}{q(\mathbf{z})} \right)$$

$$\sum_{n=1}^{N} \ln p(y_n|\theta) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|y_n, \theta)}{q(\mathbf{z})} \right) + \sum_{n=1}^{N} \ln p(y_n|\theta),$$

$$\sum_{n=1}^{N} \ln p(y_n|\theta) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(p(y_n|\mathbf{z}, \theta) \right) + \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(\frac{p(\mathbf{z}|\theta)}{q(\mathbf{z})} \right).$$

$$\sum_{n=1}^{N} \ln p(y_n|\theta) \geq \sum_{n=1}^{N} \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(p(y_n, \mathbf{z}|\theta) \right) - \sum_{\mathbf{z}} q(\mathbf{z}) \ln \left(q(\mathbf{z}) \right).$$

EM algorithm - Summary

1. We just need to calculate the expected value of the joint distribution w.r.t. the posterior of the latent variables.
2. We are able to estimate the parameters θ by maximizing an easier objective function.

Outline

1 EM algorithm

2 **Variational EM**

3 Approximate inference for Neural Networks

Problem formulation

In this case,

$$p(\mathbf{z}|\mathbf{y}) = \frac{p(\mathbf{y}, \mathbf{z})}{p(\mathbf{y})}$$

is intractable. Most of the time because

$$\begin{aligned} p(\mathbf{y}) &= \int_{\mathbf{z}} p(\mathbf{y}, \mathbf{z}) \, \mathrm{d}\, \mathbf{z} \\ &= \int_{\mathbf{z}} p(\mathbf{y}|\mathbf{z}) p(\mathbf{z}) \, \mathrm{d}\, \mathbf{z} \\ &= \mathbb{E}_{p(\mathbf{z})} \left[p(\mathbf{y}|\mathbf{z}) \right] \end{aligned}$$

is difficult to calculate.

Variational EM - Objective

We are interested in estimating $p(\mathbf{z}|\mathbf{y})$. This can be achieved by solving the following optimization problem,

$$q^*(\mathbf{z}) = \text{arg min } \mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y}))$$

where

$$\mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) = \int q(\mathbf{z}) \ln \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{y})} \mathrm{d}\mathbf{z}$$

Variational EM - Objective

Analysing the KL divergence between $q(\mathbf{z})$ and $p(\mathbf{z}|\mathbf{y})$,

$$\begin{aligned}
\mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) &= \int q(\mathbf{z}) \ln \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{y})} \mathrm{d}\mathbf{z} \\
&= \int q(\mathbf{z}) \ln q(\mathbf{z}) \mathrm{d}\mathbf{z} - \int q(\mathbf{z}) \ln p(\mathbf{z}|\mathbf{y}) \mathrm{d}\mathbf{z} \\
&= \int q(\mathbf{z}) \ln q(\mathbf{z}) \mathrm{d}\mathbf{z} - \int q(\mathbf{z}) \ln \frac{p(\mathbf{z}, \mathbf{y})}{p(\mathbf{y})} \mathrm{d}\mathbf{z} \\
&= \int q(\mathbf{z}) \ln q(\mathbf{z}) \mathrm{d}\mathbf{z} - \int q(\mathbf{z}) \ln p(\mathbf{z}, \mathbf{y}) \mathrm{d}\mathbf{z} + \ln p(\mathbf{y})
\end{aligned}$$

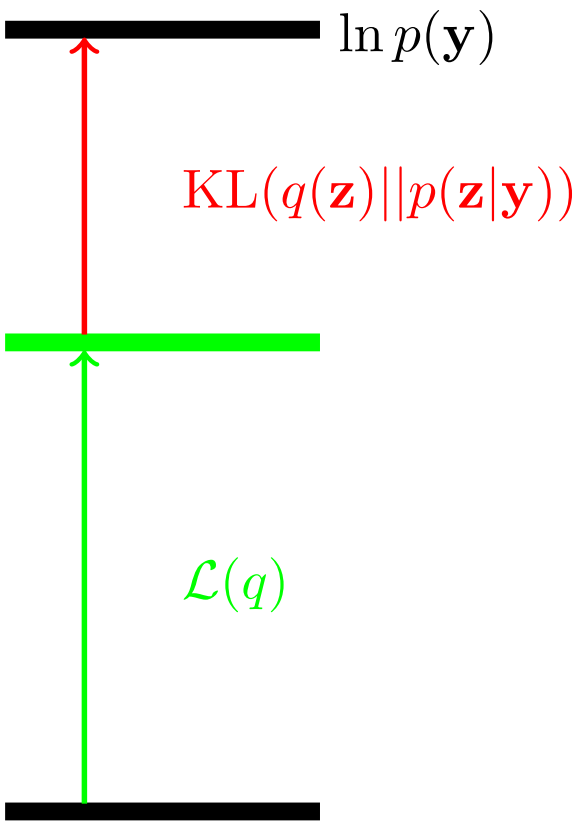
Variational EM - The evidence lower bound

From previous equation we have,

$$\begin{aligned}\ln p(\mathbf{y}) &= \mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) + \int q(\mathbf{z}) \ln p(\mathbf{z}, \mathbf{y}) \mathrm{d}\mathbf{z} - \int q(\mathbf{z}) \ln q(\mathbf{z}) \mathrm{d}\mathbf{z} \\ &= \mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) + \mathcal{L}(q).\end{aligned}$$

Given that $\mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) \geq 0$, then the lower bound of $\ln p(\mathbf{y})$ is $\mathcal{L}(q)$.

Variational EM - The evidence lower bound



Variational EM - Summary

1. If the posterior $p(\boldsymbol{Z}|\boldsymbol{y})$ is intractable or complex (main issue is to calculate $p(\boldsymbol{y})$).
2. We can approximate the posterior by minimizing $\text{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z}|\boldsymbol{y}))$, which results in maximizing $\mathcal{L}(q)$.
3. The E-step is an iterative process.

Variational EM - Remark

$$\mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) = \int q(\mathbf{z}) \ln \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{y})} \mathbf{dz}$$

$$\mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) = \int q(\mathbf{z}) \ln q(\mathbf{z}) \mathbf{dz} - \int q(\mathbf{z}) \ln p(\mathbf{z}, \mathbf{y}) \mathbf{dz} + \ln p(\mathbf{y})$$

$$\mathrm{KL}(q(\mathbf{z})||p(\mathbf{z}|\mathbf{y})) = -\int q(\mathbf{z}) \ln p(\mathbf{y}|\mathbf{z}) \mathbf{dz} + \int q(\mathbf{z}) \ln \frac{q(\mathbf{z})}{p(\mathbf{z})} \mathbf{dz} + \ln p(\mathbf{y})$$

Outline

1 EM algorithm

2 Variational EM

3 **Approximate inference for Neural Networks**

Approx. NNs - Problem definition

Let's assume we have a dataset $\mathcal{D} = \{\mathbf{X}, \mathbf{Y}\}$, with $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ and $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$

$$\mathrm{KL}\left(q_\theta(\boldsymbol{\omega}) \| p(\boldsymbol{\omega}|\mathcal{D})\right) \propto -\int q_\theta(\boldsymbol{\omega}) \log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega} + \mathrm{KL}\left(q_\theta(\boldsymbol{\omega}) \| p(\boldsymbol{\omega})\right),$$

$$= -\sum_{i=1}^{N} \int q_\theta(\boldsymbol{\omega}) \log p\left(\mathbf{y}_i | \mathbf{f}^{\omega}\left(\mathbf{x}_i\right)\right) \mathrm{d}\boldsymbol{\omega} + \mathrm{KL}\left(q_\theta(\boldsymbol{\omega}) \| p(\boldsymbol{\omega})\right),$$

where $\boldsymbol{\omega}$ is a vector composed by the stacking of all weights.

Approx. NNs - Problem definition

1. The summed-over terms log-likelihood are not tractable for BNNs with more than a single hidden layer.
2. This objective requires us to perform computations over the entire dataset, which can be too costly for large N.

Approx. NNs - Solutions

We can reduce the data size problem by data sub-sampling (also referred to as mini-batch optimization).

$$\widehat{\mathcal{L}}_{\mathrm{VI}}(\theta) := -\frac{N}{M}\sum_{i\in S}\int q_\theta(\boldsymbol{\omega})\log p\left(\mathbf{y}_i|\mathbf{f}^{\boldsymbol{\omega}}\left(\mathbf{x}_i\right)\right)\mathrm{d}\boldsymbol{\omega} + \mathrm{KL}\left(q_\theta(\boldsymbol{\omega})||p(\boldsymbol{\omega})\right)$$

with a random index set S of size M.

Approx. NNs - Data size solution

We can reduce the data size problem by data sub-sampling (also referred to as mini-batch optimization).

$$\widehat{\mathcal{L}}_{\mathrm{VI}}(\theta) := -\frac{N}{M} \sum_{i \in S} \int q_\theta(\boldsymbol{\omega}) \log p\left(\mathbf{y}_i | \mathbf{f}^{\boldsymbol{\omega}}\left(\mathbf{x}_i\right)\right) \mathrm{d}\boldsymbol{\omega} + \mathrm{KL}\left(q_\theta(\boldsymbol{\omega}) \| p(\boldsymbol{\omega})\right)$$

with a random index set S of size M.

Approx. NNs - Expected value solution

We can reduce the data size problem by data sub-sampling (also referred to as mini-batch optimization).

$$\mathbb{E}_{q_\theta(\boldsymbol{\omega})}[\log p\left(\mathbf{y}_i|\mathbf{f}^{\boldsymbol{\omega}}\left(\mathbf{x}_i\right)\right)] = \int q_\theta(\boldsymbol{\omega}) \log p\left(\mathbf{y}_i|\mathbf{f}^{\boldsymbol{\omega}}\left(\mathbf{x}_i\right)\right) \mathrm{d}\boldsymbol{\omega}$$

$$q_\theta(\boldsymbol{\omega}) = \prod_{i=1}^{L} q_\theta\left(\mathbf{W}_i\right) = \prod_{i=1}^{L}\prod_{j=1}^{K_i}\prod_{k=1}^{K_{i+1}} q_{m_{ijk},\sigma_{ijk}}\left(w_{ijk}\right) = \prod_{i,j,k} \mathcal{N}\left(w_{ijk}; m_{ijk}, \sigma_{ijk}^2\right)$$

$$\mathbb{E}_{q_\theta(\boldsymbol{\omega})}[\log p\left(\mathbf{y}_i|\mathbf{f}^{\boldsymbol{\omega}}\left(\mathbf{x}_i\right)\right)] \approx \frac{1}{M}\sum_{m=1}^{M} q_\theta(\boldsymbol{\omega}_j) \log p\left(\mathbf{y}_i|\mathbf{f}^{\boldsymbol{\omega}}\left(\mathbf{x}_i\right)\right)$$

Approx. NNs - Expected value solution

A better way is to use Pathwise Gradient Estimator

$$\mathbb{E}_{q_{\boldsymbol{\theta}}(\boldsymbol{\omega})}\left[f_{\boldsymbol{\theta}}(\boldsymbol{\omega})\right] \longrightarrow \mathbb{E}_{q_0(\boldsymbol{\epsilon})}\left[f_{\boldsymbol{\theta}}(g(\boldsymbol{\theta};\boldsymbol{\epsilon}))\right]$$

$$q_{ijk}\left(w_{ijk}\right) = g(\theta_{ijk}, \epsilon_{ijk}) = m_{ijk} + \sigma_{ijk}\epsilon_{ijk}$$

then

$$\widehat{\mathcal{L}}_{\mathrm{VI}}(\theta) := -\frac{N}{M}\sum_{i\in S}\int q_\theta(\boldsymbol{\omega}) \log p\left(\mathbf{y}_i|\mathbf{f}^{\boldsymbol{\omega}}\left(\mathbf{x}_i\right)\right) \mathrm{d}\boldsymbol{\omega} + \mathrm{KL}\left(q_\theta(\boldsymbol{\omega})\|p(\boldsymbol{\omega})\right)$$

becomes

$$\widehat{\mathcal{L}}_{\mathrm{MC}}(\theta) = -\frac{N}{M}\sum_{i\in S}\log p\left(\mathbf{y}_i|\mathbf{f}^{g(\theta,\epsilon)}\left(\mathbf{x}_i\right)\right) + \mathrm{KL}\left(q_\theta(\boldsymbol{\omega})\|p(\boldsymbol{\omega})\right)$$

where $\mathbb{E}_{S,\epsilon}\left[\hat{\mathcal{L}}_{\mathrm{MC}}(\theta)\right] = \mathcal{L}_{\mathrm{VI}}(\theta)$.

Algorithm 1 Minimise divergence between $q_\theta(\boldsymbol{\omega})$ and $p(\boldsymbol{\omega}|X, Y)$

1: Given dataset $\mathbf{X}, \mathbf{Y}$,
2: Define learning rate schedule η,
3: Initialise parameters θ randomly.
4: **repeat**
5: Sample M random variables $\widehat{\boldsymbol{\epsilon}}_i \sim p(\boldsymbol{\epsilon})$, S a random subset of $\{1, .., N\}$ of size M.
6: Calculate stochastic derivative estimator w.r.t. θ:

$$\widehat{\Delta\theta} \leftarrow -\frac{N}{M}\sum_{i\in S}\frac{\partial}{\partial\theta}\log p(\mathbf{y}_i|\mathbf{f}^{g(\theta,\widehat{\boldsymbol{\epsilon}_i})}(\mathbf{x}_i)) + \frac{\partial}{\partial\theta}\mathrm{KL}(q_\theta(\boldsymbol{\omega})||p(\boldsymbol{\omega})).$$

7: Update θ:

$$\theta \leftarrow \theta + \eta\widehat{\Delta\theta}.$$

8: **until** θ has converged.

Approx. NNs - Dropout

$$
\begin{aligned}
\hat{\mathbf{y}} &= \hat{\mathbf{h}}\mathbf{M}_2 \\
&= (\mathbf{h} \odot \hat{\boldsymbol{\epsilon}}_2)\mathbf{M}_2 \\
&= (\mathbf{h} \cdot \mathrm{diag}(\hat{\boldsymbol{\epsilon}}_2))\mathbf{M}_2 \\
&= \mathbf{h}(\mathrm{diag}(\hat{\boldsymbol{\epsilon}}_2)\mathbf{M}_2) \\
&= \sigma\Big(\hat{\mathbf{x}}\mathbf{M}_1 + \mathbf{b}\Big)(\mathrm{diag}(\hat{\boldsymbol{\epsilon}}_2)\mathbf{M}_2) \\
&= \sigma\Big((\mathbf{x} \odot \hat{\boldsymbol{\epsilon}}_1)\mathbf{M}_1 + \mathbf{b}\Big)(\mathrm{diag}(\hat{\boldsymbol{\epsilon}}_2)\mathbf{M}_2) \\
&= \sigma\Big(\mathbf{x}(\mathrm{diag}(\hat{\boldsymbol{\epsilon}}_1)\mathbf{M}_1) + \mathbf{b}\Big)(\mathrm{diag}(\hat{\boldsymbol{\epsilon}}_2)\mathbf{M}_2)
\end{aligned}
$$

Ountline

1. EM algorithm
2. Variational EM
3. Approximate inference for Neural Networks

Approx. NNs - Laplace approximation

The standard Laplace approximation is obtained by taking the second-order Taylor expansion around a mode of a distribution.
If we approximate the log posterior over the weights of a network given some data $\mathcal{D}$ around a MAP estimate $\boldsymbol{\omega}^*$, we obtain

$$\log p(\boldsymbol{\omega}|\mathcal{D}) \approx \log p\left(\boldsymbol{\omega}^*|\mathcal{D}\right) - \frac{1}{2}\left(\boldsymbol{\omega} - \boldsymbol{\omega}^*\right)^\top \overline{H}\left(\boldsymbol{\omega} - \boldsymbol{\omega}^*\right),$$

where $\overline{H} = \mathbb{E}[H]$ is the average Hessian of the negative log posterior. The posterior over the weights is then approximated as Gaussian:

$$\boldsymbol{\omega} \sim \mathcal{N}\left(\boldsymbol{\omega}^*, \overline{H}^{-1}\right)$$

References I


Bishop, C. M. (2006).
Pattern Recognition and Machine Learning (Information Science and Statistics).
Springer-Verlag, Berlin, Heidelberg.


Gal, Y. (2016).
Uncertainty in Deep Learning.
PhD thesis, University of Cambridge.


Ritter, H., Botev, A., and Barber, D. (2018).
A Scalable Laplace Approximation for Neural Networks.
In *International Conference on Learning Representations*.