Modern Computational Statistics

Lecture 10: Expectation Maximization

Cheng Zhang

School of Mathematical Sciences, Peking University

October 23, 2019

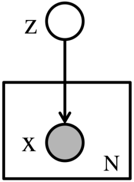
Introduction

- In this lecture, we discuss Expectation-Maximization (EM), which is an iterative optimization method dealing with missing or latent data.
- In such cases, we may assume the observed data x are generated from random variable X along with missing or unobserved data z from random variable Z. We envision complete data would have been $y = (x, z)$.
- Very often, the inclusion of the observed data z is a *data augmentation* strategy to ease computation. In this case, Z is often referred to as *latent* variable.

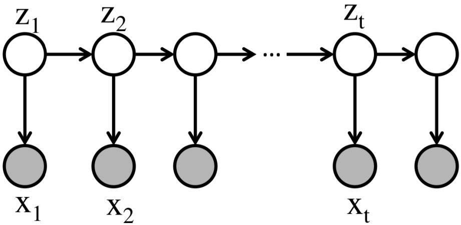
Latent Variable Model

- Some of the variables in the model are not observed.
- Examples: mixture model, hidden Markov model (HMM), latent Dirichlet allocation (LDA), etc.
- We consider the learning problem of latent variable models

Mixture Model



Hidden Markov Model



Marginal Likelihood

- complete data likelihood $p(x, z|\theta)$, θ is model parameter
- When z is missing, we need to marginalize out z and use the marginal log-likelihood for learning

$$\log p(x|\theta) = \log \sum_z p(x, z|\theta)$$

- Examples: Gaussian mixture model. $z \sim \mathrm{Discrete}(\pi)$, $\theta = (\pi, \mu, \Sigma)$

$$\begin{aligned} p(x|\theta) &= \sum_k p(z = k|\theta) p(x|z = k, \theta) \\ &= \sum_k \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \\ &= \sum_k \pi_k \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)\right) \end{aligned}$$

Learning in Latent Variable Model

- For most of these latent variable models, when the missing components z are observed, the complete data likelihood often factorizes, and the maximum likelihood estimates hence have closed-form solutions.
- When z are not observed, marginalization destroys the factorizible structure and makes learning much more difficult.
- How to learn in this scenario?
  - **Idea 1**: simply take derivative and use gradient ascent directly
  - **Idea 2**: find appropriate estimates of z (e.g., using the current conditional distribution $p(z|x, \theta)$), fill them in and do complete data learning – This is EM!

Expectation Maximization

- At each iteration, the EM algorithm involves two steps
  - based on the current $\theta^{(t)}$, fill in unobserved z to get *complete data* (x, z')
  - Update θ to maximize the complete data log-likelihood $\ell(x, z'|\theta) = \log p(x, z'|\theta)$
- How to choose z'?
  - Use conditional distribution $p(z|x, \theta^{(t)})$
  - Take full advantage of the current estimates $\theta^{(t)}$

$$\mathbb{E}_{p(z|x,\theta^{(t)})}\ell(x, z|\theta) = \sum_{z} p(z|x, \theta^{(t)})\ell(x, z|\theta)$$

In some sense, this is our best guess (as shown later).

EM Algorithm

More specifically, we start from some initial $\theta^{(0)}$. In each iteration, we follow the two steps below

- **Expectation (E-step)**: compute $p(z|x, \theta^{(t)})$ and form the expectation using the current estimate $\theta^{(t)}$

$$Q^{(t)}(\theta) = \mathbb{E}_{p(z|x,\theta^{(t)})} \ell(x, z|\theta)$$

- **Maximization (M-step)**: Find θ that maximizes the expected complete data log-likelihood

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

In many cases, the expectation is easier to handle than the marginal log-likelihood.

How does EM Work?

- EM algorithm can be viewed as optimizing a lower bound on the marginal log-likelihood $\mathcal{L}(\theta) = \log p(x|\theta)$
- A class of lower bounds

$$
\begin{aligned}
\mathcal{L}(\theta) = \log \sum_z p(x, z|\theta) &= \log \sum_z q(z) \frac{p(x, z|\theta)}{q(z)} \\
&\geq \sum_z q(z) \log \frac{p(x, z|\theta)}{q(z)} \qquad \text{- Jensen's inequality} \\
&= \sum_z q(z) \log p(x, z|\theta) - \sum_z q(z) \log q(z), \quad \forall q(z)
\end{aligned}
$$

- The term in the last equation is often called *Free-energy*

$$
\mathcal{F}(q, \theta) = \sum_z q(z) \log p(x, z|\theta) - \sum_z q(z) \log q(z)
$$

Lower Bound Maximization

- Free-energy is a lower bound of the true log-likelihood

$$\mathcal{L}(\theta) \geq \mathcal{F}(q, \theta)$$

- EM is simply doing coordinate ascent on $\mathcal{F}(q, \theta)$
  - E-step: Find $q^{(t)}$ that maximizes $\mathcal{F}(q, \theta^{(t)})$
  - M-step: Find $\theta^{(t+1)}$ that maximizes $\mathcal{F}(q^{(t)}, \theta)$
- Properties:
  - Each iteration improves $\mathcal{F}$

$$\mathcal{F}(q^{(t+1)}, \theta^{(t+1)}) \geq \mathcal{F}(q^{(t)}, \theta^{(t)})$$

  - Each iteration improves $\mathcal{L}$ as well

$$\mathcal{L}(\theta^{(t+1)}) \geq \mathcal{L}(\theta^{(t)})$$

will show later

E-step

- Find q that maximizes $\mathcal{F}(q, \theta^{(t)})$

$$
\begin{aligned}
\mathcal{F}(q, \theta) &= \sum_z q(z) \log p(x, z|\theta) - \sum_z q(z) \log q(z) \\
&= \sum_z q(z) \log \frac{p(z|x, \theta) p(x|\theta)}{q(z)} \\
&= \sum_z q(z) \log \frac{p(z|x, \theta)}{q(z)} + \log p(x|\theta) \\
&= \mathcal{L}(\theta) - D_{\mathrm{KL}} \left(q(z) \| p(z|x, \theta) \right) \\
&\leq \mathcal{L}(\theta)
\end{aligned}
$$

E-step

$$\mathcal{F}(q, \theta^{(t)}) = \mathcal{L}(\theta^{(t)}) - D_{\mathrm{KL}}(q(z) \| p(z|x, \theta^{(t)}))$$

- KL divergence is non-negative and is minimized (equals to 0) iff the two distributions are identical.
- Therefore, $\mathcal{F}(q, \theta^{(t)})$ is maximized at $q^{(t)}(z) = p(z|x, \theta^{(t)})$.
- So when we are computing $p(z|x, \theta^{(t)})$, we are actually computing $\arg\max_q \mathcal{F}(q, \theta^{(t)})$
- Moreover,

  $$\mathcal{F}(q^{(t)}, \theta^{(t)}) = \mathcal{L}(\theta^{(t)})$$

  this means the lower bound matches the true log-likelihood at $\theta^{(t)}$, which is crucial for the improvement on $\mathcal{L}$.

M-step

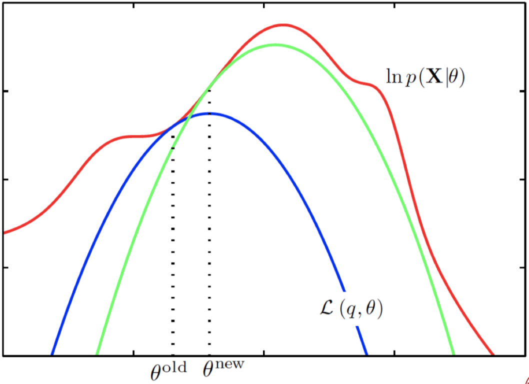
- Find $\theta^{(t+1)}$ that maximizes $\mathcal{F}(q^{(t)}, \theta)$

$$\begin{aligned}
\theta^{(t+1)} &= \arg\max_{\theta} \mathcal{F}(q^{(t)}, \theta) \\
&= \arg\max_{\theta} \sum_{z} p(z|x, \theta^{(t)}) \log p(x, z|\theta) + H(p(z|x, \theta^{(t)})) \\
&= \arg\max_{\theta} \mathbb{E}_{p(z|x,\theta^{(t)})} \ell(x, z|\theta)
\end{aligned}$$

- The expected complete data log-likelihood usually can be solved in the same manner (closed-form solutions) as the fully-observed model.

Monotonicity of EM

$$\begin{aligned}\mathcal{L}(\theta^{(t+1)}) &\geq \mathcal{F}(q^{(t)}, \theta^{(t+1)}) \\ &\geq \mathcal{F}(q^{(t)}, \theta^{(t)}) = \mathcal{L}(\theta^{(t)})\end{aligned}$$



EM for Exponential Families

- When the complete data follow an exponential family distribution (in canonical form), the density is

$$p(x, z|\theta) = h(x, z) \exp(\theta \cdot T(x, z) - A(\theta))$$

- E-step

$$\begin{aligned} Q^{(t)}(\theta) &= \mathbb{E}_{p(z|x,\theta^{(t)})} \log p(x, z|\theta) \\ &= \theta \cdot \mathbb{E}_{p(z|x,\theta^{(t)})} T(x, z) - A(\theta) + \mathrm{Const} \end{aligned}$$

- M-step

$$\nabla_\theta Q^{(t)}(\theta) = 0 \Rightarrow \mathbb{E}_{p(z|x,\theta^{(t)})} T(x, z) = \nabla_\theta A(\theta) = \mathbb{E}_{p(x,z|\theta)} T(x, z)$$

Examples: Censored Survival Times

- In survival analyses, we often have to terminate our study before observing the real survival times, leading to censored survival data.
- Suppose the observed data are $Y = \{(t_1, \delta_1), \dots, (t_n, \delta_n)\}$, where $T_j \sim \mathrm{Exp}(\mu)$ and δ_j is the indicator of a censored sample. WLOG, assume $\delta_i = 0, i \le r, \quad \delta_i = 1, i > r$
- The log-likelihood function is

$$\begin{aligned}\log p(Y|\mu) &= \sum_{i=1}^{r} \log p(t_i|\mu) + \sum_{i>r} \log p(T_i > t_i|\mu) \\ &= -r \log \mu - \sum_{i=1}^{n} t_i/\mu\end{aligned}$$

- The MLE of μ: $\hat{\mu} = \sum_{i=1}^{n} t_i/r$

Examples: Censored Survival Times

- Let us see how EM works in this simple case.
- Let $t = (T_1, \ldots, T_n) = (T_1, \ldots, T_r, z)$ be the complete data vector, where $z = (T_{r+1}, \ldots, T_n)$ are the unobserved $n - r$ censored random variables.
- Natural parameter $1/\mu$, sufficient statistics $\sum_{i=1}^n T_i$, and $\mathbb{E}_\mu \sum_{i=1}^n T_i = n\mu$
- By the lack of memory, $T_i | T_i > t_i \sim t_i + \mathrm{Exp}(\mu)$, $\forall i > r$.

$$\mathbb{E}_{p(z|Y,\mu^{(k)})} \sum_{i=1}^n T_i = \sum_{i=1}^r t_i + \sum_{i>r} t_i + (n-r)\mu^{(k)}$$

- Update formula

$$\mu^{(k+1)} = \frac{\sum_{i=1}^n t_i + (n-r)\mu^{(k)}}{n}$$

Gaussian Mixture Model

- Consider clustering of data $X = \{x_1, \ldots, x_N\}$ using a finite mixture of Gaussians.

$$z \sim \mathrm{Discrete}(\pi), \quad x|z = k \sim \mathcal{N}(\mu_k, \Sigma_k)$$

  $\theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$ are model parameters

- Complete data log-likelihood

$$\begin{aligned}\log p(x, z|\theta) &= \log \prod_{k=1}^{K} (p(z = k)p(x|z = k))^{1_{z=k}} \\ &= \sum_{k=1}^{K} 1_{z=k}(\log \pi_k + \log \mathcal{N}(x|\mu_k, \Sigma_k))\end{aligned}$$

E-step

- Compute the conditional probability $p(z_n|x_n, \theta^{(t)})$ via Bayes' theorem

$$p(z_n|x_n, \theta) = \frac{p(z_n, x_n|\theta)}{\sum_{z_n} p(z_n, x_n|\theta)}$$

$$p(z_n = k|x_n, \theta^{(t)}) = \frac{\pi_k^{(t)} \mathcal{N}(x_n|\mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_k \pi_k^{(t)} \mathcal{N}(x_n|\mu_k^{(t)}, \Sigma_k^{(t)})}$$

- Denote $\gamma_{n,k}^{(t)} \triangleq p(z_n = k|x_n, \theta^{(t)})$, which can be viewed as a *soft clustering* of x_n

$$\sum_k \gamma_{n,k}^{(t)} = 1$$

E-step

- Expected complete-data log-likelihood

$$
\begin{aligned}
Q^{(t)}(\theta) &= \sum_n \sum_{z_n} p(z_n|x_n, \theta^{(t)}) \log p(x_n, z_n|\theta) \\
&= \sum_n \sum_k \gamma_{n,k}^{(t)} (\log \pi_k + \log \mathcal{N}(x_n|\mu_k, \Sigma_k)) \\
&= \sum_k \sum_n \gamma_{n,k}^{(t)} (\log \pi_k + \log \mathcal{N}(x_n|\mu_k, \Sigma_k))
\end{aligned}
$$

Substitute $\mathcal{N}(x_n|\mu_k, \Sigma_k)$ in

$$
\begin{aligned}
Q^{(t)}(\theta) = \sum_k \sum_n \gamma_{n,k}^{(t)} \Big(\log \pi_k - \frac{d}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma_k| \\
- \frac{1}{2}(x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k) \Big)
\end{aligned}
$$

M-step

- Maximize $Q^{(t)}(\theta)$ with respect to π using Lagrange multipliers

$$\pi_k^{(t+1)} \propto \sum_n \gamma_{n,k}^{(t)}$$

Therefore

$$\pi_k^{(t+1)} = \frac{\sum_n \gamma_{n,k}^{(t)}}{\sum_k \sum_n \gamma_{n,k}^{(t)}} = \frac{\sum_n \gamma_{n,k}^{(t)}}{\sum_n \sum_k \gamma_{n,k}^{(t)}} = \frac{\sum_n \gamma_{n,k}^{(t)}}{N}$$

- Note that $\sum_n \gamma_{n,k}^{(t)}$ can be viewed as the weighted number of data points in mixture component k, and $\pi_k^{(t+1)}$ is the fraction of data the belongs to mixture component k.

M-step

- Compute the derivative w.r.t μ_k

$$\frac{\partial Q^{(t)}(\theta)}{\partial \mu_k} = \sum_n \gamma_{n,k}^{(t)} \Sigma_k^{-1}(x_n - \mu_k) = \Sigma_k^{-1} \sum_n \gamma_{n,k}^{(t)}(x_n - \mu_k)$$

- Therefore,

$$\mu_k^{(t+1)} = \frac{\sum_n \gamma_{n,k}^{(t)} x_n}{\sum_n \gamma_{n,k}^{(t)}}$$

$\mu_k^{(t+1)}$ is the weighted mean of data points assigned to mixture component k

- Similarly, we can get

$$\Sigma_k^{(t+1)} = \frac{\sum_n \gamma_{n,k}^{(t)}(x_n - \mu_k^{(t+1)})(x_n - \mu_k^{(t+1)})^T}{\sum_n \gamma_{n,k}^{(t)}}$$

EM algorithm for Gaussian Mixture Models

- **E-step**: Compute the soft clustering probabilities

$$\gamma_{n,k}^{(t)} = \frac{\pi_k^{(t)} \mathcal{N}(x_n | \mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_k \pi_k^{(t)} \mathcal{N}(x_n | \mu_k^{(t)}, \Sigma_k^{(t)})}$$

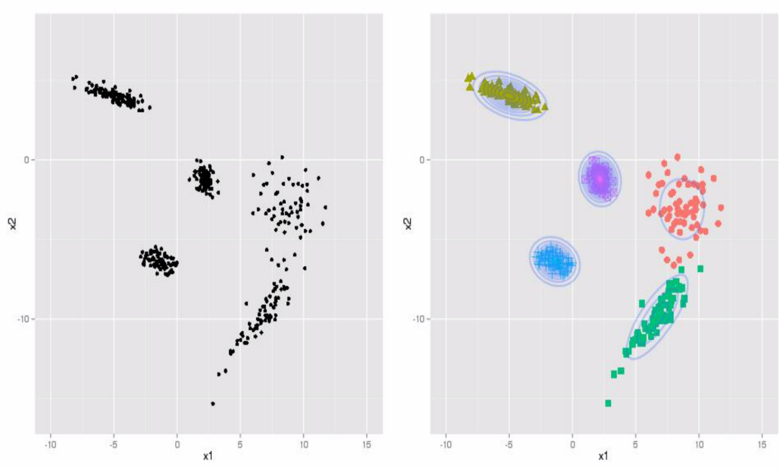
- **M-step**: Update parameters

$$\pi_k^{(t+1)} = \frac{\sum_n \gamma_{n,k}^{(t)}}{N}$$

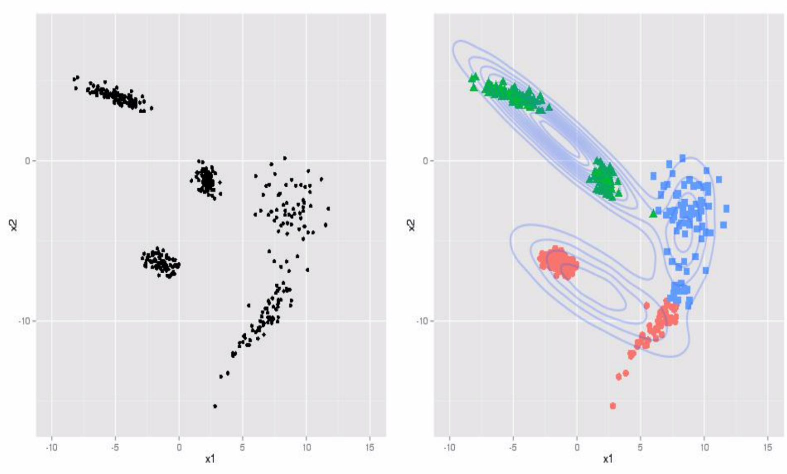
$$\mu_k^{(t+1)} = \frac{\sum_n \gamma_{n,k}^{(t)} x_n}{\sum_n \gamma_{n,k}^{(t)}}$$

$$\Sigma_k^{(t+1)} = \frac{\sum_n \gamma_{n,k}^{(t)} (x_n - \mu_k^{(t+1)})(x_n - \mu_k^{(t+1)})^T}{\sum_n \gamma_{n,k}^{(t)}}$$

Examples: Mixture of 5 Gaussians



Examples: Mixture of 3 Gaussians



Connection to k-means

- The k-means algorithm follows two steps
  - Assignment step: assign data to the nearest cluster

$$\gamma_{n,k} = \begin{cases} 1, & k = \arg\min_{k'} \|x_n - \mu_{k'}\| \\ 0, & \text{otherwise} \end{cases}$$

  - Update step: set μ_k to the mean of data points assigned to the k-th cluster

$$\mu_k = \frac{\sum_n \gamma_{n,k}^{(t)} x_n}{\sum_n \gamma_{n,k}^{(t)}} = \frac{1}{N_k} \sum_{n:\gamma_{n,k}=1} x_n$$

  N_k is the number of data points assigned to the k-th cluster.

- Therefore, k-means can be viewed as a special case of EM for Gaussian mixture models where $\Sigma_k = I$ and $\gamma_{n,k}$ are hard assignments instead of soft clustering probabilities.

Hidden Markov Model

- Sequence data $x_1, x_2, \ldots, x_T$, each $x_n \in \mathbb{R}^d$
- Hidden variables $z_1, z_2, \ldots, z_T$, each $z_t \in \{1, 2, \ldots, K\}$
- Joint probability

$$p(x, z) = p(z_1) \prod_{t=1}^{T-1} p(z_{t+1}|z_t) \prod_{t=1}^{T} p(x_t|z_t)$$

- $p(x_t|z_t)$ is the *emission probability*, could be a Gaussian

$$p(x_t|z_t = k) = \mathcal{N}(x_t|\mu_k, \Sigma_k)$$

- $p(z_{t+1}|z_t)$ is the *transition probability*, a $K \times K$ matrix $a_{ij} = p(z_{t+1} = j|z_t = i)$, $\sum_j a_{ij} = 1$
- $p(z_1) \sim \text{Discrete}(\pi)$ is the prior for the first hidden state

Expected Complete Data Log-likelihood

- The expected complete data log-likelihood is

$$
\begin{aligned}
Q &= \mathbb{E}_{p(z|x)} \log p(x, z) \\
&= \sum_{z} p(z|x) \left(\log p(z_1) + \sum_{t=1}^{T-1} \log p(z_{t+1}|z_t) + \sum_{t=1}^{T} \log p(x_t|z_t) \right) \\
&= \sum_{z_1} p(z_1|x) \log p(z_1) + \sum_{t=1}^{T-1} \sum_{z_t, z_{t+1}} p(z_t, z_{t+1}|x) \log p(z_{t+1}|z_t) \\
&\qquad + \sum_{t=1}^{T} \sum_{z_t} p(z_t|x) \log p(x_t|z_t)
\end{aligned}
$$

- Therefore, in the E-step, we need to compute unary and pairwise marginal probabilities $p(z_t|x)$ and $p(z_t, z_{t+1}|x)$.

E-step: Forward-Backward Algorithm

- Using the sequential structure of HMM, we can compute these marginal probabilities via **dynamic programming**.
- The **forward algorithm**

$$\begin{aligned}
\alpha_{t+1}(j) &= p(z_{t+1} = j, x_1, \dots, x_{t+1}) \\
&= \sum_i p(z_{t+1} = j, z_t = i, x_1, \dots, x_{t+1}) \\
&= p(x_{t+1} | z_{t+1} = j) \sum_i p(z_{t+1} = j | z_t = i) p(z_t, x_1, \dots, x_t) \\
&= p(x_{t+1} | z_{t+1} = j) \sum_i a_{ij} p(z_t, x_1, \dots, x_t) \\
&= p(x_{t+1} | z_{t+1} = j) \sum_i a_{ij} \alpha_t(i)
\end{aligned}$$

E-step: Forward-Backward Algorithm

- The **backward algorithm**

$$\begin{aligned}\beta_t(i) &= p(x_{t+1},\ldots,x_T|z_t=i)\\ &= \sum\nolimits_j p(x_{t+1},\ldots,x_T,z_{t+1}=j|z_t=i)\\ &= \sum\nolimits_j a_{ij}p(x_{t+1}|z_{t+1}=j)\beta_{t+1}(j)\end{aligned}$$

- Unary marginal probability

$$p(z_t=j|x)\propto p(z_t=j,x)=\alpha_t(j)\beta_t(j)$$

- Pairwise marginal probability

$$\begin{aligned}p(z_{t+1}=j,z_t=i|x)\propto p(z_{t+1}=j,z_t=i,x)\\ =\alpha_t(i)a_{ij}p(x_{t+1}|z_{t+1}=j)\beta_{t+1}(j)\end{aligned}$$

M-step

- From the E-step, we have

$$\gamma_{t,k} = p(z_t = k|x) = \frac{\alpha_t(k)\beta_t(k)}{\sum_k \alpha_t(k)\beta_t(k)}$$

$$\xi_t(i,j) = p(z_{t+1} = j, z_t = i|x) = \frac{\alpha_t(i)a_{ij}p(x_{t+1}|z_{t+1} = j)\beta_{t+1}(j)}{\sum_k \alpha_t(k)\beta_t(k)}$$

- The expected complete data log-likelihood is

$$\begin{aligned} Q = \sum_k \gamma_{1,k} \log \pi_k + \sum_{t=1}^{T-1} \sum_{i,j} \xi_t(i,j) \log a_{ij} \\ + \sum_{t=1}^{T} \sum_k \gamma_{t,k} \log \mathcal{N}(x_t|\mu_k, \Sigma_k) \end{aligned}$$

- Closed form solution for M-step – just like in the Gaussian mixture model

Summary

EM algorithm finds MLE for models with missing/latent variables. Applicable if the following pieces are easy to solve

- Estimating missing data from observed data using current parameters (E-step)
- Find complete data MLE (M-step)

Pros

- No need for gradients, learning rates, etc.
- Fast convergence
- Monotonicity. Guaranteed to improve $\mathcal{L}$ at every iteration

Cons

- Can get stuck at local optimal
- Requires conditional distribution $p(z|x, \theta)$ to be tractable

References

- A. P. Dempster, N. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, Series B, 39:1–38, 1977.
- R. M. Neal and G. E. Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. Learning in Graphical Models, 89:355–368, 1998.
- Baum, L. E. (1972). An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. In Shisha, O. (Ed.), Inequalities III: Proceedings of the 3rd Symposium on Inequalities, 1–8. Academic Press.