The EM Algorithm (Contd.) and Some Examples

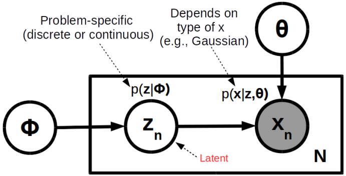
Piyush Rai

Topics in Probabilistic Modeling and Inference (CS698X)

Feb 13, 2018

Recap: Parameter Estimation in LVM

- A simple LVM: Assume each observation $\boldsymbol{x}_n$ to be associated with a "local" latent variable $\boldsymbol{z}_n$



- Denote all the unknown parameters (not including the latent variables) as $\Theta = (\theta, \phi)$
- The MLE for Θ would be $\hat{\Theta} = \arg\max_\Theta \sum_{n=1}^N \log p(\boldsymbol{x}_n|\Theta)$ where

$$
\begin{aligned}
\boldsymbol{z}_n \text{ discrete:} \quad \log p(\boldsymbol{x}_n|\Theta) &= \log \sum_{\boldsymbol{z}_n} p(\boldsymbol{x}_n, \boldsymbol{z}_n|\Theta) = \log \sum_{k=1}^{K} p(\boldsymbol{x}_n|\boldsymbol{z}_n = k, \Theta) p(\boldsymbol{z}_n = k|\Theta) \\
\boldsymbol{z}_n \text{ continuous:} \quad \log p(\boldsymbol{x}_n|\Theta) &= \log \int p(\boldsymbol{x}_n, \boldsymbol{z}_n|\Theta) d\boldsymbol{z}_n = \log \int p(\boldsymbol{x}_n|\boldsymbol{z}_n, \Theta) p(\boldsymbol{z}_n|\Theta) d\boldsymbol{z}_n
\end{aligned}
$$

- $\log p(\boldsymbol{x}_n|\Theta)$ usually doesn't have a simple form so directly doing MLE is not easy

Recap: EM for Parameter Estimation in LVM

- Instead of maximizing $\log p(\mathbf{X}|\Theta)$ (which usually doesn't have a simple form), EM solves

$$\hat{\Theta} = \arg\max_{\Theta} \mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] \qquad \text{(expected complete data log-lik)}$$

- Note: If $p(\mathbf{X}|\mathbf{Z})$ and $p(\mathbf{Z})$ are exp-fam distributions, $\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$ usually has a simple form
- EM operates in the following iterative fashion until convergence
  1. Initialize Θ as Θ^{old}
  2. **E Step:** Compute posterior $p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$ and compute $\mathcal{Q}(\Theta, \Theta^{old}) = \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$
  3. **M Step:** Compute a point estimate of Θ by solving the following problem

$$\Theta^{new} = \arg\max_{\Theta} \mathcal{Q}(\Theta, \Theta^{old}) = \arg\max_{\Theta} \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)]$$

  4. If not converged, set $\Theta^{old} = \Theta^{new}$ and go to step 2
- Note: EM infers the posterior over latent variables Θ and point estimate for parameters Θ

Recap: Justification for EM

- Based on the following identity $\log p(\mathbf{X}|\Theta) = \mathcal{L}(q, \Theta) + \mathrm{KL}(q||p_z)$ where

$$\mathcal{L}(q, \Theta) = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \left\{ \frac{p(\mathbf{X}, \mathbf{Z}|\Theta)}{q(\mathbf{Z})} \right\} \quad \text{and} \quad \mathrm{KL}(q||p_z) = -\sum_{\mathbf{Z}} q(\mathbf{Z}) \log \left\{ \frac{p(\mathbf{Z}|\mathbf{X}, \Theta)}{q(\mathbf{Z})} \right\}$$

- $\mathcal{L}(q, \Theta)$ is a lower bound on $\log p(\mathbf{X}|\Theta)$ (since $\mathrm{KL}(q||p_z) \geq 0$)
- EM does MLE for Θ by maximizing $\mathcal{L}(q, \Theta)$ using an alternating scheme
  - Step 1: Maximize $\mathcal{L}(q, \Theta)$ w.r.t q with Θ fixed at Θ^{old}

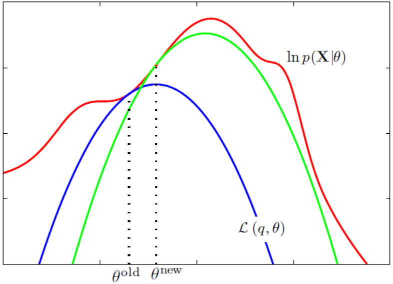
$$\hat{q} = \arg\max_q \mathcal{L}(q, \Theta^{old}) = \arg\min_q \mathrm{KL}(q||p_z) = p_z = p(\mathbf{Z}|\mathbf{X}, \Theta^{old})$$

  - Step 2: Maximize $\mathcal{L}(q, \Theta)$ w.r.t Θ using q obtained from E step, i.e.,

$$\begin{aligned} \hat{\Theta} = \arg\max_\Theta \mathcal{L}(\hat{q}, \Theta) &= \arg\max_\Theta \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \Theta^{old}) \log p(\mathbf{X}, \mathbf{Z}|\Theta) \\ &= \arg\max_\Theta \mathbb{E}_{p(\mathbf{Z}|\mathbf{X}, \Theta^{old})}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] \end{aligned}$$

- The above two steps are essentially the E and M steps, respestively, in the EM algorithm!

Recap: Convergence of EM (Pictorially)



The E step updates q such that $\mathcal{L}(q, \Theta)$ touches $\log p(\mathbf{X}|\Theta)$ at Θ^{old} (lower bound becomes tight)

The M step finds the maxima Θ^{new} of this tight lower bound

Next E step again updates q such that $\mathcal{L}(q, \Theta)$ touches $\log p(\mathbf{X}|\Theta)$ at Θ^{new} (bound is tight again)

Next M step finds the new maximia of this new tight lower bound

The process continues until we reach a local optima of $\log p(\mathbf{X}|\Theta)$

EM via Some Examples

The Basic Recipe for Deriving EM for any LVM

- Need mainly two things: The posterior $p(\boldsymbol{z}_n|\boldsymbol{x}_n, \Theta)$ over latent variables and CLL $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$
- The posterior over the latent variables, i.e., $p(\boldsymbol{z}_n|\boldsymbol{x}_n, \Theta)$

$$p(\boldsymbol{z}_n|\boldsymbol{x}_n, \Theta) \propto p(\boldsymbol{z}_n|\phi)p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

- The CLL for a simple LVM with one latent variable per data point, $\Theta = (\theta, \phi)$, and i.i.d. structure

$$\log p(\mathbf{X}, \mathbf{Z}|\Theta) = \sum_{n=1}^{N} \log p(\boldsymbol{x}_n, \boldsymbol{z}_n|\Theta) = \sum_{n=1}^{N} [\log p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta) + \log p(\boldsymbol{z}_n|\phi)]$$

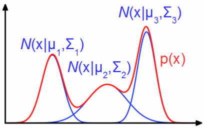
- In the M step, we do MLE on the expected CLL

$$\mathcal{Q}(\Theta, \Theta^{old}) = \sum_{n=1}^{N} \mathbb{E}_{p(\boldsymbol{z}_n|\boldsymbol{x}_n, \Theta^{old})}[\log p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta) + \log p(\boldsymbol{z}_n|\phi)$$

- Note: Computing $p(\boldsymbol{z}_n|\boldsymbol{x}_n, \Theta)$ and/or the expected CLL may require approximations
- Note: MLE on $\mathcal{Q}(\Theta, \Theta^{old})$ may or may not be possible in closed form (but for many models, it is)

EM for Gaussian Mixture Model (GMM)

- GMM: A model for data clustering and density estimation
- Assumes data generated from a mixture of K Gaussians with mixing proportions $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_K]$



- Assume the K Gaussians have mean and covariance matrices $\{\mu_k, \Sigma_k\}_{k=1}^K$
- The prior probability of observation $\boldsymbol{x}_n$ generated from the k-th Gaussian

$$p(\boldsymbol{z}_n = k) = \pi_k$$

- It's basically multinoulli prior on $\boldsymbol{z}$, i.e., $p(\boldsymbol{z}_n) = \prod_{k=1}^K \pi_k^{z_{nk}}$ (note $z_{nk} = 1$ if $\boldsymbol{z}_n = k$; 0 otherwise)
- If $\boldsymbol{z}_n = k$, we generate $\boldsymbol{x}_n$ from the k-th Gaussian. Thus $p(\boldsymbol{x}_n|\boldsymbol{z}_n = k) = \mathcal{N}(\boldsymbol{x}_n|\mu_k, \Sigma_k)$
- In this LVM, $\mathbf{Z} = [\boldsymbol{z}_n, \ldots, \boldsymbol{z}_N]$ and parameters $\Theta = \{\boldsymbol{\pi}, \{\mu_k, \Sigma_k\}_{k=1}^K\}$

EM for GMM

- We know the basic recipe already. Let's look at the relevant quantities we need to compute
- We need the posterior $p(\boldsymbol{z}_n|\boldsymbol{x}_n) \propto p(\boldsymbol{z}_n)p(\boldsymbol{x}_n|\boldsymbol{z}_n)$. Can compute it easily

$$p(\boldsymbol{z}_n = k|\boldsymbol{x}_n) \propto p(\boldsymbol{z}_n = k)p(\boldsymbol{x}_n|\boldsymbol{z}_n = k) = \pi_k \mathcal{N}(\boldsymbol{x}_n|\mu_k, \Sigma_k) \quad \text{(same as writing } p(z_{nk} = 1|\boldsymbol{x}_n))$$

- We need the CLL $\log p(\mathbf{X}, \mathbf{Z}|\Theta) = \sum_{n=1}^{N}[\log p(\boldsymbol{x}_n|\boldsymbol{z}_n) + \log p(\boldsymbol{z}_n)]$
- Note that $p(\boldsymbol{x}_n|\boldsymbol{z}_n) = \prod_{k=1}^{K}[p(\boldsymbol{x}_n|\boldsymbol{z}_n = k)]^{z_{nk}}$ and $p(\boldsymbol{z}_n) = \prod_{k=1}^{K} \pi_k^{z_{nk}}$
- Therefore the CLL would be

$$\log p(\mathbf{X}, \mathbf{Z}|\Theta) = \sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}[\log \pi_k + \log p(\boldsymbol{x}_n|\boldsymbol{z}_n = k)] = \sum_{n=1}^{N}\sum_{k=1}^{K} z_{nk}[\log \pi_k + \log \mathcal{N}(\boldsymbol{x}_n|\mu_k, \Sigma_k)]$$

- The expected CLL will be the above quantity with z_{nk} replaced by $\mathbb{E}_{p(\boldsymbol{z}_n|\boldsymbol{x}_n,\Theta)}[z_{nk}]$ where

$$\mathbb{E}[z_{nk}] \quad = \quad 0 \times p(z_{nk} = 0|\boldsymbol{x}_n) + 1 \times p(z_{nk} = 1|\boldsymbol{x}_n) = p(z_{nk} = 1|\boldsymbol{x}_n)$$

- Once we have expected CLL, we can do MLE on CLL by maximizing w.r.t. μ_k, Σ_k, π_k

EM for GMM: The Algorithm

- Initialize the parameters $\Theta^{(0)} = \{\mu_k^{(0)}, \Sigma_k^{(0)}, \pi_k^{(0)}\}_{k=1}^K$
- For $t = 1, \ldots, T$ (or until convergence)
  - For each observation $n = 1, \ldots, N$, compute posterior over latent variables

$$p(\boldsymbol{z}_n = k | \boldsymbol{x}_n) \propto \pi_k^{(t-1)} \mathcal{N}(\boldsymbol{x}_n | \mu_k^{(t-1)}, \Sigma_k^{(t-1)}), \quad k = 1, \ldots, K$$

  - Solve MLE on expected CLL to obtain $\{\mu_k^{(t)}, \Sigma_k^{(t)}, \pi_k^{(t)}\}_{k=1}^K$, where expected CLL

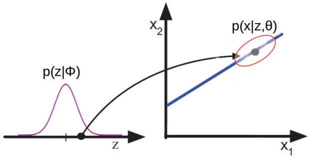
$$\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = \sum_{n=1}^N \sum_{k=1}^K \mathbb{E}[z_{nk}][\log \pi_k + \log \mathcal{N}(\boldsymbol{x}_n | \mu_k, \Sigma_k)]$$

- Note: Easy to see that the estimation of μ_k and Σ_k is a weighted MLE for a multivariate Gaussian

Probabilistic Principal Component Analysis (PPCA)

- Assume data $\boldsymbol{x}_n$ as a linear transformation of a latent variable $\boldsymbol{z}_n$, plus Gaussian noise and write

$$\boldsymbol{x}_n = \mathbf{W}\boldsymbol{z}_n + \epsilon_n \qquad \text{with} \qquad \epsilon_n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_D)$$



- The distribution of $\boldsymbol{x}_n$, conditioned on $\boldsymbol{z}_n$ will be

$$p(\boldsymbol{x}_n|\boldsymbol{z}_n) = \mathcal{N}(\mathbf{W}\boldsymbol{z}_n, \sigma^2 \mathbf{I}_D)$$

- Assume $\boldsymbol{z}_n$ to have a Gaussian prior e.g., $p(\boldsymbol{z}_n) = \mathcal{N}(\mathbf{0}, \mathbf{I}_K)$
- In this LVM, $\mathbf{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N]$ and $\Theta = (\mathbf{W}, \sigma^2)$

EM for PPCA

- As usual, need two things: The posterior $p(\boldsymbol{z}_n|\boldsymbol{x}_n)$ and the CLL $\log p(\mathbf{X}, \mathbf{Z}|\Theta)$
- The posterior $p(\boldsymbol{z}_n|\boldsymbol{x}_n)$ can be easily computed and will be (from linear Gaussian model properties)

$$p(\boldsymbol{z}_n|\boldsymbol{x}_n) \propto p(\boldsymbol{z}_n)p(\boldsymbol{x}_n|\boldsymbol{z}_n) = \mathcal{N}(\mathbf{M}^{-1}\mathbf{W}^\top\boldsymbol{x}_n, \sigma^2\mathbf{M}^{-1}) \qquad (\text{where } \mathbf{M} = \mathbf{W}^\top\mathbf{W} + \sigma^2\mathbf{I}_K)$$

- The CLL also has a simple expression and is given by

$$\log p(\mathbf{X}, \mathbf{Z}|\mathbf{W}, \sigma^2) = \log \prod_{n=1}^N p(\boldsymbol{x}_n, \boldsymbol{z}_n|\mathbf{W}, \sigma^2) = \log \prod_{n=1}^N p(\boldsymbol{x}_n|\boldsymbol{z}_n, \mathbf{W}, \sigma^2)p(\boldsymbol{z}_n) = \sum_{n=1}^N \{\log p(\boldsymbol{x}_n|\boldsymbol{z}_n, \mathbf{W}, \sigma^2) + \log p(\boldsymbol{z}_n)\}$$

- Using $p(\boldsymbol{x}_n|\boldsymbol{z}_n) = \frac{1}{(2\pi\sigma^2)^{D/2}} \exp\left[-\frac{(\boldsymbol{x}_n - \mathbf{W}\boldsymbol{z}_n)^\top(\boldsymbol{x}_n - \mathbf{W}\boldsymbol{z}_n)}{2\sigma^2}\right]$ and $p(\boldsymbol{z}_n) \propto \exp\left[-\frac{\boldsymbol{z}_n^\top\boldsymbol{z}_n}{2}\right]$ and simplifying

$$\text{CLL} = -\sum_{n=1}^N \left\{\frac{D}{2}\log\sigma^2 + \frac{1}{2\sigma^2}||\boldsymbol{x}_n||^2 - \frac{1}{\sigma^2}\boldsymbol{z}_n^\top\mathbf{W}^\top\boldsymbol{x}_n + \frac{1}{2\sigma^2}\text{tr}(\boldsymbol{z}_n\boldsymbol{z}_n^\top\mathbf{W}^\top\mathbf{W}) + \frac{1}{2}\text{tr}(\boldsymbol{z}_n\boldsymbol{z}_n^\top)\right\} \quad (\text{Exercise: Verify})$$

- To do MLE on expected CLL, we need $\mathbb{E}[\boldsymbol{z}_n]$ and $\mathbb{E}[\boldsymbol{z}_n\boldsymbol{z}_n^\top]$. Using $p(\boldsymbol{z}_n|\boldsymbol{x}_n) = \mathcal{N}(\mathbf{M}^{-1}\mathbf{W}^\top\boldsymbol{x}_n, \sigma^2\mathbf{M}^{-1})$

$$\begin{aligned}\mathbb{E}[\boldsymbol{z}_n] &= \mathbf{M}^{-1}\mathbf{W}^\top\boldsymbol{x}_n \\ \mathbb{E}[\boldsymbol{z}_n\boldsymbol{z}_n^\top] &= \mathbb{E}[\boldsymbol{z}_n]\mathbb{E}[\boldsymbol{z}_n]^\top + \text{cov}(\boldsymbol{z}_n) = \mathbb{E}[\boldsymbol{z}_n]\mathbb{E}[\boldsymbol{z}_n]^\top + \sigma^2\mathbf{M}^{-1}\end{aligned}$$

- Once we have expected CLL, we can do MLE on CLL by maximizing w.r.t. $\mathbf{W}, \sigma^2$

EM for PPCA: The Algorithm

- Initialize the parameters $\Theta^{(0)} = \{\mathbf{W}^{(0)}, \sigma^{2(0)}\}$
- For $t = 1, \ldots, T$ (or until convergence)
  - For each observation $n = 1, \ldots, N$, compute posterior over latent variables

$$p(\boldsymbol{z}_n|\boldsymbol{x}_n) = \mathcal{N}(\mathbf{M}^{-1}\mathbf{W}^{(t-1)^\top}\boldsymbol{x}_n, \sigma^{2(t-1)}\mathbf{M}^{-1}) \qquad \text{where } \mathbf{M} = \mathbf{W}^\top\mathbf{W} + \sigma^{2(t-1)}\mathbf{I}_K$$

  - Solve MLE on expected CLL to obtain $\{\mathbf{W}^{(t)}, \sigma^{2(t)}\}$, where expected CLL

$$\mathbb{E}[\log p(\mathbf{X}, \mathbf{Z}|\Theta)] = -\sum_{n=1}^{N}\left\{\frac{D}{2}\log\sigma^2 + \frac{1}{2\sigma^2}||\boldsymbol{x}_n||^2 - \frac{1}{\sigma^2}\mathbb{E}[\boldsymbol{z}_n]^\top\mathbf{W}^\top\boldsymbol{x}_n + \frac{1}{2\sigma^2}\text{tr}(\mathbb{E}[\boldsymbol{z}_n\boldsymbol{z}_n^\top]\mathbf{W}^\top\mathbf{W}) + \frac{1}{2}\text{tr}(\mathbb{E}[\boldsymbol{z}_n\boldsymbol{z}_n^\top])\right\}$$

- Can show that the MLE for $\mathbf{W}$ is

$$\mathbf{W} = \left[\sum_{n=1}^{N}\boldsymbol{x}_n\mathbb{E}[\boldsymbol{z}_n]^\top\right]\left[\sum_{n=1}^{N}\mathbb{E}[\boldsymbol{z}_n\boldsymbol{z}_n^\top]\right]^{-1} \qquad \text{(verify and also try it for } \sigma^2\text{)}$$

Parameters vs Latent Variables: The Difference?

- Both are unknowns of the models
- If we are finding a posterior over an unknown, we usually refer to it as latent variable
- If we are finding a point estimate over an unknown, we refer to it as parameter
- If we are inferring posteriors over all unknowns then there is no such distinction
- EM distinguishes unknowns as latent variables vs parameters based on how it learns them
  - EM infers posterior over some unknowns which we call latent variables
  - EM computes point estimates over other unknowns which we call parameters
- A subtle but important thing to keep in mind