Latent Variable Models and the EM Algorithm (Contd)

CS772A: Probabilistic Machine Learning
Piyush Rai
Announcement

- Mid-sem exam on Sept 19, 1800-2000
- Venue: L17, ERES seating scheme
- Syllabus: Up to today’s lecture
- Closed book exam
  - Necessary formulae/equations will be provided in the question paper
Plan Today

- Expectation Maximization (EM) algorithm for param-est/inference in LVMs

- An example of EM
  - Parameter estimation for Gaussian mixture models
Parameter Estimation in Latent Variable Models

- Assume each observation $x_n$ to be associated with a “local” latent variable $z_n$

  $p(z_n|\phi)$: A suitable prior distribution based on the nature of $z_n$
  $p(x_n|z_n, \theta)$: A suitable likelihood based on the nature of $x_n$

- Although we can do fully Bayesian inference for all the unknowns, suppose we only want a point estimate of the “global” parameters $\Theta = (\theta, \phi)$ via MLE/MAP

- Such MLE/MAP problems in LVMs are difficult to solve in a “clean” way
  - Can do gradient based opt on log-lik but usually won’t get closed form updates for $\Theta$
  - However, EM algo gives a clean way to obtain closed form updates for $\Theta$ for such LVMs
Why MLE/MAP of Params is Hard for LVMs?

- Suppose we want to estimate parameters $\Theta$ via MLE. If we knew $z_n$, we could solve

$$
\Theta_{MLE} = \arg \max_{\Theta} \sum_{n=1}^{N} \log p(x_n, z_n|\Theta) = \arg \max_{\Theta} \sum_{n=1}^{N} [\log p(z_n|\phi) + \log p(x_n|z_n, \theta)]
$$

- Easy. Usually closed form if $p(z_n|\phi)$ and $p(x_n|z_n, \theta)$ have simple forms

- However, since in LVMs, $z_n$ is hidden, the MLE problem for $\Theta$ will be the following

$$
\Theta_{MLE} = \arg \max_{\Theta} \sum_{n=1}^{N} \log p(x_n|\Theta) = \arg \max_{\Theta} \log p(X|\Theta)
$$

- $\log p(x_n|\Theta)$ will not have a simple expression since $p(x_n|\Theta)$ requires sum/integral

$$
p(x_n|\Theta) = \sum_{z_n} p(x_n, z_n|\Theta) \quad \text{... or if } z_n \text{ is continuous: } p(x_n|\Theta) = \int p(x_n, z_n|\Theta) dz_n
$$

- MLE now becomes difficult, no closed form expression for $\Theta$

- Can we maximize some other quantity instead of $\log p(x_n|\Theta)$ for this MLE?

Easy to solve

In particular, if they are exp-fam distributions

Can still use gradient based methods but won’t get clean, closed-form updates for $\Theta$
An Important Identity

- Assume \( p_z = p(Z|X, \Theta) \) and \( q(Z) \) to be some prob distribution over \( Z \), then for all \( q \)
  \[
  \log p(X|\Theta) = \mathcal{L}(q, \Theta) + KL(q||p_z)
  \]
- In the above \( \mathcal{L}(q, \Theta) = \sum_z q(Z) \log \frac{p(X,Z|\Theta)}{q(Z)} \)
- \( KL(q||p_z) = -\sum_z q(Z) \log \frac{p(Z|X, \Theta)}{q(Z)} \)
- \( KL \) is always non-negative, so \( \log p(X|\Theta) \geq \mathcal{L}(q, \Theta) \)
- Thus \( \mathcal{L}(q, \Theta) \) is a lower-bound on \( \log p(X|\Theta) \)
- Thus if we maximize \( \mathcal{L}(q, \Theta) \), it will also improve \( \log p(X|\Theta) \)
- Also, as we'll see, it's easier to maximize \( \mathcal{L}(q, \Theta) \)
Maximizing $\mathcal{L}(q, \Theta)$

- $\mathcal{L}(q, \Theta)$ depends on $q$ and $\Theta$. We’ll use ALT-OPT to maximize it.
- Let’s maximize $\mathcal{L}(q, \Theta)$ w.r.t. $q$ with $\Theta$ fixed at some $\Theta^{\text{old}}$

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

- Now let’s maximize $\mathcal{L}(q, \Theta)$ w.r.t. $\Theta$ with $q$ fixed at $\hat{q} = p_z = p(Z \| X, \Theta^{\text{old}})$

$$\Theta^{\text{new}} = \arg\max_{\Theta} \mathcal{L}(\hat{q}, \Theta) = \arg\max_{\Theta} \sum_Z p(Z \| X, \Theta^{\text{old}}) \log \left\{ \frac{p(X, Z \| \Theta)}{p(Z \| X, \Theta^{\text{old}})} \right\}$$

$$= \arg\max_{\Theta} \sum_Z p(Z \| X, \Theta^{\text{old}}) \log p(X, Z \| \Theta)$$

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

$$= \arg\max_{\Theta} Q(\Theta, \Theta^{\text{old}})$$

- $\log p(X \| \Theta)$ is called Incomplete-Data Log Likelihood (ILL)
- Since $\log p(X \| \Theta) = \mathcal{L}(q, \Theta) + KL(q \| p_z)$ is constant when $\Theta$ is held fixed at $\Theta^{\text{old}}$.
- The posterior distribution of $Z$ given current parameters $\Theta^{\text{old}}$

Maximization of expected CLL where the expectation is w.r.t. the posterior distribution of $Z$ given current parameters $\Theta^{\text{old}}$

Much easier than maximizing ILL since CLL will have simple expressions (since it is kind of akin to knowing $Z$)
The Expectation-Maximization (EM) Algorithm

- ALT-OPT of $\mathcal{L}(q, \Theta)$ w.r.t. $q$ and $\Theta$ gives the EM algorithm (Dempster, Laird, Rubin, 1977)

### The EM Algorithm

1. **Initialize** $\Theta$ as $\Theta^{(0)}$, set $t = 1$

2. **Step 1:** Compute posterior of latent variables given current parameters $\Theta^{(t-1)}$

   \[ p(z_n^{(t)}|x_n, \Theta^{(t-1)}) = \frac{p(z_n^{(t)}|\Theta^{(t-1)})p(x_n|z_n^{(t)}, \Theta^{(t-1)})}{p(x_n|\Theta^{(t-1)})} \propto \text{prior} \times \text{likelihood} \]

   Latent variables also assumed indep. a priori

3. **Step 2:** Now maximize the expected complete data log-likelihood w.r.t. $\Theta$

   \[ \Theta^{(t)} = \arg \max_{\Theta} Q(\Theta, \Theta^{(t-1)}) = \arg \max_{\Theta} \sum_{n=1}^{N} \mathbb{E}_{p(z_n^{(t)}|x_n, \Theta^{(t-1)})} [\log p(x_n, z_n^{(t)}|\Theta)] \]

   Conditional posterior of each latent variable $z_n$

   Usually computing CP + expected CLL is referred to as the E step, and max. of exp-CLL w.r.t. $\Theta$ as the M step

4. **If not yet converged**, set $t = t + 1$ and go to step 2.

   Assuming the (expected) CLL $\mathbb{E}_{p(z|x, \Theta^{old})} [\log p(X, Z|\Theta)]$ factorizes over all observations

- Note: If we can take the MAP estimate $\hat{z}_n$ of $z_n$ (not the CP) in Step 1 and maximize the CLL in Step 2 using that, i.e., do $\arg \max_{\Theta} \sum_{n=1}^{N} [\log p(x_n, \hat{z}_n^{(t)}|\Theta)]$ then this will be std ALT-OPT
The Expected CLL

- Expected CLL in EM is given by (assume observations are i.i.d.)

\[
Q(\Theta, \Theta^{old}) = \sum_{n=1}^{N} \mathbb{E}_{p(z_n|x_n, \Theta^{old})}[\log p(x_n, z_n|\Theta)]
\]

\[
= \sum_{n=1}^{N} \mathbb{E}_{p(z_n|x_n, \Theta^{old})}[\log p(x_n|z_n, \Theta) + \log p(z_n|\Theta)]
\]

- If \( p(z_n|\Theta) \) and \( p(x_n|z_n, \Theta) \) are exp-family distributions, \( Q(\Theta, \Theta^{old}) \) has a very simple form.

- In resulting expressions, replace terms containing \( z_n \)'s by their respective expectations, e.g.,
  - \( z_n \) replaced by \( \mathbb{E}_{p(z_n|x_n, \hat{\Theta})}[z_n] \)
  - \( z_n z_n^T \) replaced by \( \mathbb{E}_{p(z_n|x_n, \hat{\Theta})}[z_n z_n^T] \)

- However, in some LVMs, these expectations are intractable to compute and need to be approximated (will see some examples later), e.g., using MC integral approximation.
What’s Going On In EM?

- As we saw, the maximization of lower bound $\mathcal{L}(q, \Theta)$ had two steps
- Step 1 finds the optimal $q$ (call it $\hat{q}$) by setting it as the posterior of $Z$ given current $\Theta$
- Step 2 maximizes $\mathcal{L}(\hat{q}, \Theta)$ w.r.t. $\Theta$ which gives a new $\Theta$.

KL becomes zero and $\mathcal{L}(q, \Theta)$ becomes equal to $\log p(X|\Theta)$; thus their curves touch at current $\Theta$

Good initialization matters; otherwise would converge to a poor local optima

Note that $\Theta$ only changes in Step 2 so the objective $\log p(X|\Theta)$ can only change in Step 2

Green curve: $\mathcal{L}(\hat{q}, \Theta)$ after setting $q$ to $\hat{q}$

Local optima found for $\Theta_{MLE}$

$\log p(X|\Theta)$

$\Theta(0), \Theta(1), \Theta(2), \Theta(3)$

$\Theta(MLE)$

Also kind of similar to Newton’s method (and has second order like convergence behavior in some cases)

Unlike Newton’s method, we don’t construct and optimize a quadratic approximation, but a lower bound

Even though original MLE problem $\arg\max_{\Theta} \log p(X|\Theta)$ could be solved using gradient methods, EM often works faster and has cleaner updates

Alternating between them until convergence to some local optima
Some Applications of EM

- Mixture Models (each data-point comes from one of $K$ mixture components)
  - Examples: Mixture of Gaussians, Mixture of Experts, etc
- Latent variable models for dimensionality reduction or representation learning
  - Probabilistic PCA, Factor Analysis, Variational Autoencoders, etc
- Problems models with missing features/labels (treated as latent variables)
  - An example of problem with missing labels: Semi-supervised learning
- Hyperparameter estimation in probabilistic models (an alternative to MLE-II)
  - MLE-II estimates hyperparams by maximizing the marginal likelihood, e.g.,
    $\{\hat{\lambda}, \hat{\beta}\} = \text{argmax}_{\lambda, \beta} p(y|X, \lambda, \beta) = \text{argmax}_{\lambda, \beta} \int p(y|w, X, \beta)p(w|\lambda)dw$
  - With EM, can treat $w$ as latent var, and $\lambda, \beta$ as “parameters”
    - E step will estimate the CP of $w$ given current estimates of $\lambda, \beta$
    - M step will re-estimate $\lambda, \beta$ by maximizing the expected CLL
      \[
      E[\log p(y, w|X, \beta, \lambda)] = E[\log p(y|w, X, \beta) + \log p(w|\lambda)]
      \]
An Example: Mixture Models

- Assume $K$ probability distributions (e.g., Gaussians), one for each cluster

$$p(z_n | \phi) = \text{multinoulli}(\pi)$$
(also means $p(z_n = k | \phi) = \pi_k$)

Discrete latent variable (with $K$ possible values) or a one-hot vector of length $K$.
Modeled by a multinoulli distribution as prior

The parameter vector $\pi = [\pi_1, \pi_2, ..., \pi_K]$ of the multinoulli distribution

- Parameters of the $K$ distributions, e.g., $\theta = (\mu_k, \Sigma_k)_{k=1}^K$

The likelihood distributions

If the $z_n$ were known, it just becomes generative classification, for which which we know how to estimate $\theta$ and $\phi$, given training data

$p(x)$ is a Gaussian mixture model (GMM)

$p(x)$ is a Gaussian mixture model (GMM)

The likelihood $p(x_n | z_n = k, \theta) = \mathcal{N}(\mu_k, \Sigma_k)$

- The log-likelihood will be

$$\log p(x_n | \theta) = \log \sum_{k=1}^K p(x_n, z_n = k | \theta)$$

$$= \log \sum_{k=1}^K p(z_n = k | \phi) p(x_n | z_n = k, \theta) = \log \sum_{k=1}^K \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)$$

Assumed generated from one of the $K$ distributions depending on the true (but unknown) value of $z_n$ (which clustering will find))

MLE on this objective won’t give closed form solution for the parameters
Detour: MLE for Generative Classification

- Assume a $K$ class generative classification model with Gaussian class-conditionals
- Assume class $k = 1, 2, \ldots, K$ is modeled by a Gaussian with mean $\mu_k$ and cov matrix $\Sigma_k$
- The labels $z_n$ (known) are one-hot vecs. Also, $z_{nk} = 1$ if $z_n = k$, and $z_{nk} = 0$, o/w
- Assuming class prior as $p(z_n = k) = \pi_k$, the model has params $\Theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$
- Given training data $\{x_n, z_n\}_{n=1}^N$, the MLE solution will be

\[
\hat{\pi}_k = \frac{1}{N} \sum_{n=1}^N z_{nk} \quad \text{Same as } \frac{N_k}{N} \text{ where } N_k \text{ is } \# \text{ of training ex. for which } y_n = k
\]

\[
\hat{\mu}_k = \frac{1}{N_k} \sum_{n=1}^N z_{nk} x_n \quad \text{Same as } \frac{1}{N_k} \sum_{n:z_n=k}^N x_n
\]

\[
\hat{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^N z_{nk} (x_n - \hat{\mu}_k)(x_n - \hat{\mu}_k)^T \quad \text{Same as } \frac{1}{N_k} \sum_{n:z_n=k}^N (x_n - \hat{\mu}_k)(x_n - \hat{\mu}_k)^T
\]
Here is a formal derivation of the MLE solution for $\Theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$

$$\hat{\Theta} = \arg\max_{\Theta} p(X, Z|\Theta) = \arg\max_{\Theta} \prod_{n=1}^N p(x_n, z_n|\Theta)$$

$$= \arg\max_{\Theta} \prod_{n=1}^N p(z_n|\Theta) p(x_n|z_n, \Theta)$$

$$= \arg\max_{\Theta} \prod_{n=1}^N \prod_{k=1}^K \pi_k^{z_{nk}} \prod_{k=1}^K p(x_n|z_n = k, \Theta)^{z_{nk}}$$

$$= \arg\max_{\Theta} \log \prod_{n=1}^N \prod_{k=1}^K \left[\pi_k p(x_n|z_n = k, \Theta)\right]^{z_{nk}}$$

$$= \arg\max_{\Theta} \sum_{n=1}^N \sum_{k=1}^K z_{nk} \left[\log \pi_k + \log N(x_n|\mu_k, \Sigma_k)\right]$$

In general, in models with probability distributions from the exponential family, the MLE problem will usually have a simple analytic form.

Also, due to the form of the likelihood (Gaussian) and prior (multinoulli), the MLE problem had a nice separable structure after taking the log.

Can see that, when estimating the parameters of the $k^{th}$ Gaussian ($\pi_k, \mu_k, \Sigma_k$), we only will only need training examples from the $k^{th}$ class, i.e., examples for which $z_{nk} = 1$.

The form of this expression is important; will encounter this in GMM too.
EM for Mixture Models

So how do we estimate the parameters of a GMM where \( z_n \)'s are unknown?

- The guess about \( z_n \) can be in one of the two forms:
  - A “hard” guess – a single best value \( \hat{z}_n \) (some “optimal” value of the random variable \( z_n \))
  - The “expected” value \( \mathbb{E}[z_n] \) of the random variable \( z_n \)

- Using the hard guess \( \hat{z}_n \) of \( z_n \) will result in an ALT-OPT like algorithm.
- Using the expected value of \( z_n \) will give the so-called Expectation-Maximization (EM) algo.

Well, you kind of already know how to do this. 😊 Remember generative classification?

Yes, exactly. 😊 However, just like in gen-class, you will need to repeat the guess and estimate them a few times until you converge.

Hmmmm.. So can we make a guess what the value of each \( z_n \) and then estimate \( \theta \) and \( \phi \) as we do in case of generative classification??

EM is pretty much like ALT-OPT but with soft/expected values of the latent variables.
**EM for Gaussian Mixture Model (GMM)**

- EM finds $\Theta_{MLE}$ by maximizing $\mathbb{E}[\log p(X, Z|\Theta)]$ rather than $\log p(X, \hat{Z} |\Theta)$
- Note: Expectation will be w.r.t. the conditional posterior distribution of $Z$, i.e., $p(Z|X, \Theta)$
- The EM algorithm for GMM operates as follows
  - Initialize $\Theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$ as $\tilde{\Theta}$
  - Repeat until convergence
    - Compute conditional posterior $p(Z|X, \tilde{\Theta})$. Since obs are i.i.d, compute separately for each $n$ (and for $k = 1, 2, \ldots K$)
    $$p(z_n = k|x_n, \tilde{\Theta}) \propto p(z_n = k|\tilde{\Theta}) p(x_n|z_n = k, \tilde{\Theta}) = \hat{\pi}_k \mathcal{N}(x_n|\hat{\mu}_k, \hat{\Sigma}_k)$$
    - Update $\Theta$ by maximizing the expected complete data log-likelihood
      $$\hat{\Theta} = \arg\max_\Theta \mathbb{E}_{p(z|X, \tilde{\Theta})}[\log p(X, Z|\Theta)] = \sum_{n=1}^N \mathbb{E}_{p(z_n|x_n, \tilde{\Theta})}[\log p(x_n, z_n|\Theta)]$$

Solution has a similar form as ALT-OPT (or gen. class.), except we now have the expectation of $z_{nk}$ being used.

$\hat{\pi}_k = \frac{1}{N} \sum_{n=1}^N \mathbb{E}[z_{nk}]$  $\hat{\mu}_k = \frac{1}{N_k} \sum_{n=1}^N \mathbb{E}[z_{nk}]x_n$

$N_k : $ Effective number of points in cluster $k$

$\hat{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^N \mathbb{E}[z_{nk}] (x_n - \hat{\mu}_k)(x_n - \hat{\mu}_k)^T$
EM for GMM (Contd)

- The EM algo for GMM required $\mathbb{E}[z_{nk}]$. Note $z_{nk} \in \{0,1\}$.

\[ \mathbb{E}[z_{nk}] = \gamma_{nk} = 0 \times p(z_{nk} = 0|x_n, \bar{\theta}) + 1 \times p(z_{nk} = 1|x_n, \bar{\theta}) = p(z_{nk} = 1|x_n, \bar{\theta}) \propto \hat{\pi}_k \mathcal{N}(x_n | \hat{\mu}_k, \hat{\Sigma}_k) \]

**EM for Gaussian Mixture Model**

1. Initialize $\Theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$ as $\Theta^{(0)}$, set $t = 1$

2. **E step**: compute the expectation of each $z_n$ (we need it in M step)

\[ \mathbb{E}[z_{nk}^{(t)}] = \gamma_{nk}^{(t)} = \frac{\pi_k^{(t-1)} \mathcal{N}(x_n | \mu_k^{(t-1)}, \Sigma_k^{(t-1)})}{\sum_{\ell=1}^K \pi_\ell^{(t-1)} \mathcal{N}(x_n | \mu_\ell^{(t-1)}, \Sigma_\ell^{(t-1)})} \quad \forall n, k \]

3. Given “responsibilities” $\gamma_{nk} = \mathbb{E}[z_{nk}]$, and $N_k = \sum_{n=1}^N \gamma_{nk}$, re-estimate $\Theta$ via MLE

\[ \mu_k^{(t)} = \frac{1}{N_k} \sum_{n=1}^N \gamma_{nk}^{(t)} x_n \]

\[ \Sigma_k^{(t)} = \frac{1}{N_k} \sum_{n=1}^N \gamma_{nk}^{(t)} (x_n - \mu_k^{(t)})(x_n - \mu_k^{(t)})^T \]

\[ \pi_k^{(t)} = \frac{N_k}{N} \]

4. Set $t = t + 1$ and go to step 2 if not yet converged
EM vs Gradient-based Methods

- Can also estimate params using gradient-based optimization instead of EM
  - We can usually explicitly sum over or integrate out the latent variables $Z$, e.g.,
    \[
    \mathcal{L}(\Theta) = \log p(X|\Theta) = \log \sum_Z p(X, Z|\Theta)
    \]
  - Now we can optimize $\mathcal{L}(\Theta)$ using first/second order optimization to find the optimal $\Theta$

- EM is usually preferred over this approach because
  - The M step has often simple closed-form updates for the parameters $\Theta$
  - Often constraints (e.g., PSD matrices) are automatically satisfied due to form of updates
  - In some cases\(^\dagger\), EM usually converges faster (and often like second-order methods)
    - E.g., Example: Mixture of Gaussians with when the data is reasonably well-clustered
  - EM also provides the conditional posterior over the latent variables $Z$ (from E step)

EM: Some Final Comments

- The E and M steps may not always be possible to perform exactly. Some reasons:
  - The conditional posterior of latent variables $p(Z|X, \Theta)$ may not be easy to compute.
    - Will need to approximate $p(Z|X, \Theta)$ using methods such as MCMC or variational inference.
  - Even if $p(Z|X, \Theta)$ is easy, the expected CLL may not be easy to compute.
    
    $$E[\log p(X, Z|\Theta)] = \int \log p(X, Z|\Theta)p(Z|X, \Theta)dZ$$

  - Maximization of the expected CLL may not be possible in closed form.

- EM works even if the M step is only solved approximately (Generalized EM).

- If M step has multiple parameters whose updates depend on each other, they are updated in an alternating fashion - called Expectation Conditional Maximization (ECM).

- Other advanced probabilistic inference algos are based on ideas similar to EM:
  - E.g., Variational EM, Variational Bayes (VB) inference, a.k.a. Variational Inference (VI).