Training Latent Variable Models with Auto-encoding Variational Bayes: A Tutorial

Outline

- Why this tutorial
- Expectation Maximization (EM)
- Approximate E step as variational inference
- Approximate M step
- AEVB
- General recipe for applying AEVB to almost any latent variable models
- Example derivations: Factor Analysis, VAE

Try MLE for latent variable models:

$$
\begin{aligned}
\boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(D) \\
&= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i) \\
&= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{N} \log \int p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)\, d\boldsymbol{z}_i
\end{aligned}
$$

Generally the integral is intractable: no closed-form solution if $\boldsymbol{z}_i$ is continuous; even if $\boldsymbol{z}_i$ is discrete there are additional problems that make this less appealing (CAUTION)

One can derive a lower bound to log likelihood using Jensen's inequality:

$$
\begin{aligned}
\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i) &= \sum_{i=1}^{N} \log \int q_i(\boldsymbol{z}_i) \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)}{q_i(\boldsymbol{z}_i)}\, d\boldsymbol{z}_i \\
&\geq \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)}{q_i(\boldsymbol{z}_i)}\right] \quad \text{(equality if } q_i(\boldsymbol{z}_i) = p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \boldsymbol{x}_i)) \\
&= \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)] + \sum_{i=1}^{N} \mathbb{H}(q_i)
\end{aligned}
$$

EM algorithm: derive $p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \boldsymbol{x}_i)$ using Bayes rule set $q_i(\boldsymbol{z}_i) = \underline{p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \boldsymbol{x}_i)}$ so that bound is tight, then maximize $\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)]$. At first sight, it might not be clear why maximizing $\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)]$ is easiser, but it's indeed easier in practice for plenty of simple models.

Alternative derivation to EM that gives new insights

$$
\begin{aligned}
\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i) &= \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i)] \\
&= \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)}{p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \boldsymbol{x}_i)}\right] \\
&= \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}\left[\log \left(\frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)}{q(\boldsymbol{z}_i)} \cdot \frac{q(\boldsymbol{z}_i)}{p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \boldsymbol{x}_i)}\right)\right] \\
&= \underbrace{\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q(\boldsymbol{z}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q(\boldsymbol{z}_i)]}_{\text{lower bound we had on previously slide}} + D_{\mathbb{KL}}(q(\boldsymbol{z}_i) || \, p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \boldsymbol{x}_i))
\end{aligned}
$$

We see that the gap between the lower bound and the log likelihood is the sum of all KL-divergences between the chosen distributions $q(\boldsymbol{z}_i)$ and the true posteriors!

Goal of E-step: make the lower bound tight

$$\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i) = \underbrace{\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q_i(\boldsymbol{z}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_i(\boldsymbol{z}_i)]}_{\text{lower bound/ELBO}} + \underbrace{\sum_{i=1}^{N} D_{\mathbb{KL}}(q_i(\boldsymbol{z}_i) || \, p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \, \boldsymbol{x}_i))}_{\text{gap}}$$

One way to think about this is to minimize $D_{\mathbb{KL}}(q_i(\boldsymbol{z}_i) || \, p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \, \boldsymbol{x}_i))$ for $i = 1, \ldots, N$.

But the problem is simply the variational inference problem:

$$q_i^* = \arg\max_{q_i \in Q} D_{\mathbb{KL}}(q_i(\boldsymbol{z}_i) || \, p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \, \boldsymbol{x}_i))$$

How to solve this problem since $p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \, \boldsymbol{x}_i)$ is **intractable** (or it is tractable but we just want a more general solution that does not require explicit derivations)? There's a solution but two perspectives:

- Since the LHS does not contain q_i, minimizing the gap with respect to q_i's is equivalent to maximizing the ELBO with respect to q_i. Fortunately, the terms in ELBO are easy to evaluate, and the expectation can be sidestepped with a technique we'll later discuss.
- Apply the "textbook" VI solution:

$$D_{\mathbb{KL}}(q_i(\boldsymbol{z}_i) || \, \underline{p_{\boldsymbol{\theta}}(\boldsymbol{z}_i | \, \boldsymbol{x}_i)}) = D_{\mathbb{KL}}(q_i(\boldsymbol{z}_i) || \, \underline{p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i)}) + \underbrace{\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i)}_{\text{can be dropped}}$$

It is convenient to define Q as a parametrized family of distributions.

But as N increases, the number of parameters it takes to define all the q_i's grow linearly.

It is convenient to represent $\{q_i\}$ by a neural network $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)$ such that $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) = q_i(\boldsymbol{z}_i)$.

The changes are then reflected in our lower bound:

$$\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i) \;\geq\; \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)]$$

Approximate E-step as variational inference: stochastic optimization of ELBO

We can maximize ELBO with respect to ϕ with minibatch gradient ascent until convergence:

$$
\begin{aligned}
\boldsymbol{\phi}^{t+1} &\leftarrow \boldsymbol{\phi}^{t} + \eta \nabla_{\boldsymbol{\phi}} \Bigg\{ \underbrace{\frac{1}{N_B} \sum_{i=1}^{N_B} \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)]}_{\text{ELBO over a minibatch}} \Bigg\}_{\boldsymbol{\phi}=\boldsymbol{\phi}^t} \\
&= \boldsymbol{\phi}^{t} + \eta \frac{1}{N_B} \sum_{i=1}^{N_B} \nabla_{\boldsymbol{\phi}} \{ \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)] \}_{\boldsymbol{\phi}=\boldsymbol{\phi}^t} \\
&= \boldsymbol{\phi}^{t} + \eta \frac{1}{N_B} \sum_{i=1}^{N_B} \nabla_{\boldsymbol{\phi}} \{ \mathbb{E}_{\boldsymbol{\varepsilon}_i \sim q(\boldsymbol{\varepsilon}_i)} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)] \}_{\boldsymbol{\phi}=\boldsymbol{\phi}^t} \quad (\boldsymbol{z}_i = r(\boldsymbol{\varepsilon}_i, \boldsymbol{\phi}, \boldsymbol{x}_i)) \\
&= \boldsymbol{\phi}^{t} + \eta \frac{1}{N_B} \sum_{i=1}^{N_B} \cancel{\mathbb{E}_{\boldsymbol{\varepsilon}_i \sim q(\boldsymbol{\varepsilon}_i)}} [\nabla_{\boldsymbol{\phi}} \{ \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) \}_{\boldsymbol{\phi}=\boldsymbol{\phi}^t}] \\
&\simeq \boldsymbol{\phi}^{t} + \eta \frac{1}{N_B} \sum_{i=1}^{N_B} \nabla_{\boldsymbol{\phi}} \{ \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) \}_{\boldsymbol{\phi}=\boldsymbol{\phi}^t} \\
&= \boldsymbol{\phi}^{t} + \eta \underline{\nabla_{\boldsymbol{\phi}}} \Bigg\{ \underbrace{\frac{1}{N_B} \sum_{i=1}^{N_B} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)}_{\text{just evaluate this expression in pytorch!}} \Bigg\}_{\boldsymbol{\phi}=\boldsymbol{\phi}^t}
\end{aligned}
$$

where we have applied the reparametrization trick in line 3. It makes the assumption that sampling from $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)$ can be made equivalent to (i) first sampling from some base distribution $\boldsymbol{\varepsilon}_i \sim q(\boldsymbol{\varepsilon}_i)$ that's free of $\boldsymbol{\phi}$ and then (ii) transforming the same with a deterministic and differentiable function $r(\boldsymbol{\varepsilon}_i, \boldsymbol{\phi}, \boldsymbol{x}_i)$.

Now that ELBO is a tight bound to log likelihood, we can maximize ELBO with respect to generative parameters $\boldsymbol{\theta}$ with minibatch gradient ascent until convergence:

$$\begin{aligned}
\boldsymbol{\theta}^{t+1} &\leftarrow \boldsymbol{\theta}^t + \eta \nabla_{\boldsymbol{\theta}} \left\{ \underbrace{\frac{1}{N_B} \sum_{i=1}^{N_B} \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)]}_{\text{ELBO over a minibatch}} \right\}_{\boldsymbol{\theta}=\boldsymbol{\theta}^t} \\
&= \boldsymbol{\theta}^t + \eta \frac{1}{N_B} \sum_{i=1}^{N_B} \nabla_{\boldsymbol{\theta}} \{ \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)} [\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)] \}_{\boldsymbol{\theta}=\boldsymbol{\theta}^t} \\
&= \boldsymbol{\theta}^t + \eta \frac{1}{N_B} \sum_{i=1}^{N_B} \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)} [\nabla_{\boldsymbol{\theta}} \{ \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) \}_{\boldsymbol{\theta}=\boldsymbol{\theta}^t}] \quad \text{(no reparametrization)} \\
&\simeq \boldsymbol{\theta}^t + \eta \frac{1}{N_B} \sum_{i=1}^{N_B} \nabla_{\boldsymbol{\theta}} \{ \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) \}_{\boldsymbol{\theta}=\boldsymbol{\theta}^t} \quad \text{where} \quad \boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) \\
&= \boldsymbol{\theta}^t + \eta \nabla_{\boldsymbol{\theta}} \left\{ \underbrace{\frac{1}{N_B} \sum_{i=1}^{N_B} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)}_{\text{just evaluate this expression in pytorch!}} \right\}_{\boldsymbol{\theta}=\boldsymbol{\theta}^t}
\end{aligned}$$

1. Define the graphical model; write down the distributions for $p_{\boldsymbol{\theta}}(\mathbf{z}_i)$ and $p_{\boldsymbol{\theta}}(\mathbf{x}_i|\mathbf{z}_i)$.
2. Decide on the distribution for $q_{\boldsymbol{\phi}}(\mathbf{z}_i|\mathbf{x}_i)$.
3. Derive how to "sample" from the approximate posterior via the reparametrization trick.
4. For that model, implement f.
5. Alternate between E-steps and M-steps, or just do them simultaneously until convergence:

$$(\boldsymbol{\phi}^{t+1}, \boldsymbol{\theta}^{t+1}) \leftarrow (\boldsymbol{\phi}^{t}, \boldsymbol{\theta}^{t}) + \eta \nabla_{(\boldsymbol{\phi},\boldsymbol{\theta})} \left\{ \frac{1}{N_B} \sum_{i=1}^{N_B} f(\mathbf{x}_i, \mathbf{z}_i, \boldsymbol{\theta}, \boldsymbol{\phi}) \right\}_{(\boldsymbol{\phi},\boldsymbol{\theta})=(\boldsymbol{\phi}^t,\boldsymbol{\theta}^t)} \tag{8}$$

where we've defined f to be the shorthand for

$$f(\boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\theta}, \boldsymbol{\phi}) = \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)$$

Further variance reduction (the reconstruction-KL interpretation):

$$\begin{aligned}
& \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)] \\
= \; & \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i | \boldsymbol{z}_i) + \log p_{\boldsymbol{\theta}}(\boldsymbol{z}_i) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i)] \\
= \; & \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i | \boldsymbol{z}_i)] - \nabla_{\boldsymbol{\phi}} \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i)}[\log q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) - \log p_{\boldsymbol{\theta}}(\boldsymbol{z}_i)] \\
= \; & \mathbb{E}_{\boldsymbol{z}_i \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_i)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i | \boldsymbol{z}_i)] - D_{\mathbb{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_i | \boldsymbol{x}_i) || p_{\boldsymbol{\theta}}(\boldsymbol{z}_i))
\end{aligned}$$

For certain choices of $q_{\boldsymbol{\phi}}$ and $p_{\boldsymbol{\theta}}$, the KL can be evaluated analytically, no need to sample the second term.

Using AEVB to train Factor Analysis Model

(See PDF for code snippets and plots)

(See PDF for code snippets and plots)