Deep Generative Models

Lecture 4

Roman Isachenko

AI Masters

2024, Spring

Recap of previous lecture

Let split $\mathbf{x}$ and $\mathbf{z}$ in two parts:

$$\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2] = [\mathbf{x}_{1:d}, \mathbf{x}_{d+1:m}]; \quad \mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2] = [\mathbf{z}_{1:d}, \mathbf{z}_{d+1:m}].$$

Coupling layer

$$\begin{cases} \mathbf{x}_1 = \mathbf{z}_1; \\ \mathbf{x}_2 = \mathbf{z}_2 \odot \boldsymbol{\sigma}_{\boldsymbol{\theta}}(\mathbf{z}_1) + \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{z}_1). \end{cases} \qquad \begin{cases} \mathbf{z}_1 = \mathbf{x}_1; \\ \mathbf{z}_2 = (\mathbf{x}_2 - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{x}_1)) \odot \frac{1}{\boldsymbol{\sigma}_{\boldsymbol{\theta}}(\mathbf{x}_1)}. \end{cases}$$

Estimating the density takes 1 pass, sampling takes 1 pass!

Jacobian

$$\det\left(\frac{\partial \mathbf{z}}{\partial \mathbf{x}}\right) = \det\begin{pmatrix} \mathbf{I}_d & 0_{d \times m-d} \\ \frac{\partial \mathbf{z}_2}{\partial \mathbf{x}_1} & \frac{\partial \mathbf{z}_2}{\partial \mathbf{x}_2} \end{pmatrix} = \prod_{j=1}^{m-d} \frac{1}{\sigma_j(\mathbf{x}_1)}.$$

Coupling layer is a special case of autoregressive NF.

Dinh L., Sohl-Dickstein J., Bengio S. Density estimation using Real NVP, 2016

Recap of previous lecture

Continuous-in-time dynamics

$$\frac{d\mathbf{z}(t)}{dt} = f_{\boldsymbol{\theta}}(\mathbf{z}(t), t); \quad \text{with initial condition } \mathbf{z}(t_0) = \mathbf{z}_0.$$

$$\mathbf{z}(t_1) = \int_{t_0}^{t_1} f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)dt + \mathbf{z}_0 = \text{ODESolve}(\mathbf{z}(t_0), f_{\boldsymbol{\theta}}, t_0, t_1).$$

Euler update step

$$\frac{\mathbf{z}(t + \Delta t) - \mathbf{z}(t)}{\Delta t} = f_{\boldsymbol{\theta}}(\mathbf{z}(t), t) \implies \mathbf{z}(t + \Delta t) = \mathbf{z}(t) + \Delta t \cdot f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)$$

Theorem (Picard)

If f is uniformly Lipschitz continuous in $\mathbf{z}$ and continuous in t, then the ODE has a **unique** solution.

$$\mathbf{x} = \mathbf{z}(t_1) = \mathbf{z}(t_0) + \int_{t_0}^{t_1} f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)dt$$

$$\mathbf{z} = \mathbf{z}(t_0) = \mathbf{z}(t_1) + \int_{t_1}^{t_0} f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)dt$$

Chen R. T. Q. et al. Neural Ordinary Differential Equations, 2018

Recap of previous lecture

Theorem (Kolmogorov-Fokker-Planck: special case)

If f is uniformly Lipschitz continuous in $\mathbf{z}$ and continuous in t, then

$$\frac{d \log p(\mathbf{z}(t), t)}{dt} = -\text{tr}\left(\frac{\partial f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)}{\partial \mathbf{z}(t)}\right).$$

$$\log p(\mathbf{z}(t_1), t_1) = \log p(\mathbf{z}(t_0), t_0) - \int_{t_0}^{t_1} \text{tr}\left(\frac{\partial f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)}{\partial \mathbf{z}(t)}\right) dt.$$

▶ **Discrete-in-time NF**: evaluation of determinant of the Jacobian costs $O(m^3)$ (we need invertible f).

▶ **Continuous-in-time NF**: getting the trace of the Jacobian costs $O(m^2)$ (we need smooth f).

Hutchinson's trace estimator

$$\log p(\mathbf{z}(t_1)) = \log p(\mathbf{z}(t_0)) - \mathbb{E}_{p(\boldsymbol{\epsilon})} \int_{t_0}^{t_1} \left[\boldsymbol{\epsilon}^T \frac{\partial f}{\partial \mathbf{z}} \boldsymbol{\epsilon}\right] dt.$$

Grathwohl W. et al. FFJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models, 2018

Recap of previous lecture

Forward pass (Loss function)

$$\mathbf{z} = \mathbf{x} + \int_{t_1}^{t_0} f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)dt, \quad L(\mathbf{z}) = \log p(\mathbf{z})$$

$$L(\mathbf{z}) = L\left(\mathbf{x} + \int_{t_1}^{t_0} f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)dt\right) = L\big(\text{ODESolve}(\mathbf{x}, f_{\boldsymbol{\theta}}, t_1, t_0)\big)$$

Adjoint functions

$$\mathbf{a_z}(t) = \frac{\partial L}{\partial \mathbf{z}(t)}; \quad \mathbf{a_\theta}(t) = \frac{\partial L}{\partial \boldsymbol{\theta}(t)}.$$

These functions show how the gradient of the loss depends on the hidden state $\mathbf{z}(t)$ and parameters $\boldsymbol{\theta}$.

Theorem (Pontryagin)

$$\frac{d\mathbf{a_z}(t)}{dt} = -\mathbf{a_z}(t)^T \cdot \frac{\partial f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)}{\partial \mathbf{z}}; \quad \frac{d\mathbf{a_\theta}(t)}{dt} = -\mathbf{a_z}(t)^T \cdot \frac{\partial f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)}{\partial \boldsymbol{\theta}}.$$

Chen R. T. Q. et al. Neural Ordinary Differential Equations, 2018

Recap of previous lecture

Forward pass

$$\mathbf{z} = \mathbf{z}(t_0) = \int_{t_0}^{t_1} f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)dt + \mathbf{x} \quad \Rightarrow \quad \text{ODE Solver}$$

Backward pass

$$\left.\begin{array}{l} \dfrac{\partial L}{\partial \boldsymbol{\theta}(t_1)} = \mathbf{a}_{\boldsymbol{\theta}}(t_1) = -\int_{t_0}^{t_1} \mathbf{a_z}(t)^T \dfrac{\partial f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)}{\partial \boldsymbol{\theta}(t)}dt + 0 \\[3mm] \dfrac{\partial L}{\partial \mathbf{z}(t_1)} = \mathbf{a_z}(t_1) = -\int_{t_0}^{t_1} \mathbf{a_z}(t)^T \dfrac{\partial f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)}{\partial \mathbf{z}(t)}dt + \dfrac{\partial L}{\partial \mathbf{z}(t_0)} \\[3mm] \mathbf{z}(t_1) = -\int_{t_1}^{t_0} f_{\boldsymbol{\theta}}(\mathbf{z}(t), t)dt + \mathbf{z}_0. \end{array}\right\} \Rightarrow \text{ODE Solver}$$

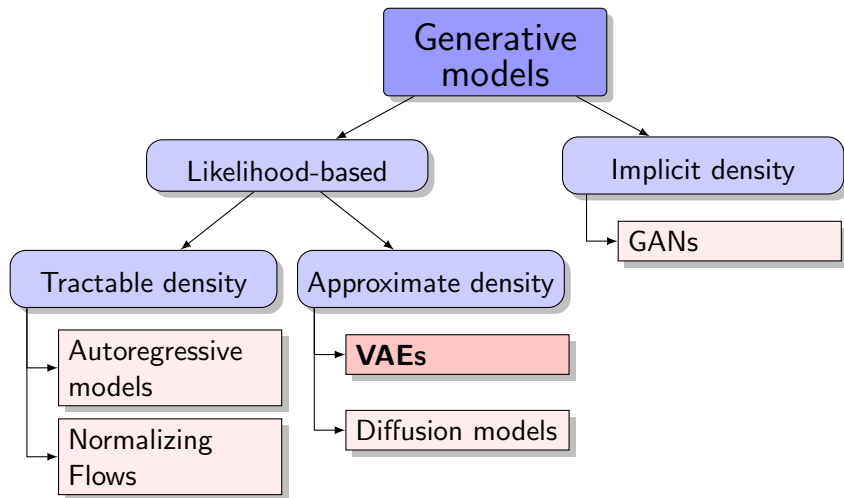**Note:** These scary formulas are the standard backprop in the discrete case.

Chen R. T. Q. et al. Neural Ordinary Differential Equations, 2018

Outline

1. Latent variable models (LVM)

2. Variational lower bound (ELBO)

3. EM-algorithm, amortized inference

Outline

Generative models zoo

Bayesian framework

Bayes theorem

$$p(\mathbf{t}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathbf{t})p(\mathbf{t})}{p(\mathbf{x})} = \frac{p(\mathbf{x}|\mathbf{t})p(\mathbf{t})}{\int p(\mathbf{x}|\mathbf{t})p(\mathbf{t})d\mathbf{t}}$$

- $\mathbf{x}$ – observed variables, $\mathbf{t}$ – unobserved variables (latent variables/parameters);
- $p(\mathbf{x}|\mathbf{t})$ – likelihood;
- $p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{t})p(\mathbf{t})d\mathbf{t}$ – evidence;
- $p(\mathbf{t})$ – prior distribution, $p(\mathbf{t}|\mathbf{x})$ – posterior distribution.

Meaning

We have unobserved variables $\mathbf{t}$ and some prior knowledge about them $p(\mathbf{t})$. Then, the data $\mathbf{x}$ has been observed. Posterior distribution $p(\mathbf{t}|\mathbf{x})$ summarizes the knowledge after the observations.

Bayesian framework

Let consider the case, where the unobserved variables **t** is our model parameters $\boldsymbol{\theta}$.

- ▶ $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$ – observed samples;
- ▶ $p(\boldsymbol{\theta})$ – prior parameters distribution (we treat model parameters $\boldsymbol{\theta}$ as random variables).

Posterior distribution

$$p(\boldsymbol{\theta}|\mathbf{X}) = \frac{p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{X})} = \frac{p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}$$

If evidence $p(\mathbf{X})$ is intractable (due to multidimensional integration), we can't get posterior distribution and perform the exact inference.

Maximum a posteriori (MAP) estimation

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathbf{X}) = \arg\max_{\boldsymbol{\theta}} \big(\log p(\mathbf{X}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta})\big)$$

Latent variable models (LVM)

MLE problem

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} p(\mathbf{X}|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{n} p(\mathbf{x}_i|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\mathbf{x}_i|\boldsymbol{\theta}).$$

The distribution $p(\mathbf{x}|\boldsymbol{\theta})$ could be very complex and intractable (as well as real distribution $\pi(\mathbf{x})$).

Extended probabilistic model

Introduce latent variable $\mathbf{z}$ for each sample $\mathbf{x}$

$$p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z}); \quad \log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) + \log p(\mathbf{z}).$$

$$p(\mathbf{x}|\boldsymbol{\theta}) = \int p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})d\mathbf{z} = \int p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z})d\mathbf{z}.$$
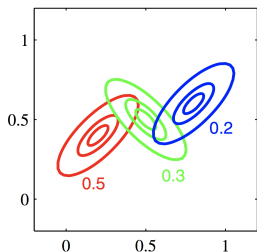
Motivation

The distributions $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$ and $p(\mathbf{z})$ could be quite simple.
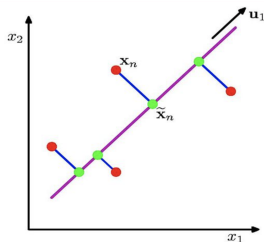
Latent variable models (LVM)

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \log \int p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) p(\mathbf{z}) d\mathbf{z} \to \max_{\boldsymbol{\theta}}$$
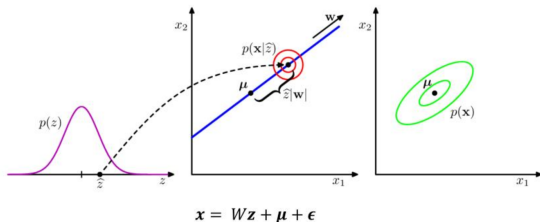
Examples

Mixture of gaussians   *PCA model*

- $p(\mathbf{x}|z, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$
- $p(z) = \text{Categorical}(\boldsymbol{\pi})$

- $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I})$
- $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|0, \mathbf{I})$

Bishop C. Pattern Recognition and Machine Learning, 2006

Latent variable models (LVM)

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \log \int p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) p(\mathbf{z}) d\mathbf{z} \to \max_{\boldsymbol{\theta}}$$

PCA projects original data **X** onto a low dimensional latent space while maximizing the variance of the projected data.

$$x = Wz + \mu + \epsilon$$

- ▶ $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I})$
- ▶ $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|0, \mathbf{I})$
- ▶ $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I})$
- ▶ $p(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{M}^{-1}\mathbf{W}^T(\mathbf{x} - \boldsymbol{\mu}), \sigma^2\mathbf{M}\right)$, where $\mathbf{M} = \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}$

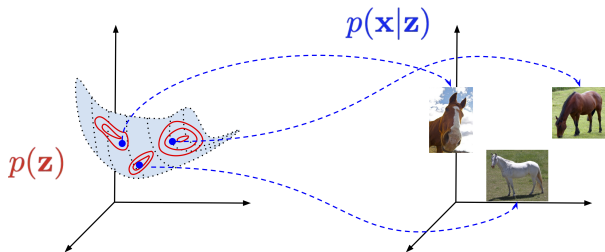# Maximum likelihood estimation for LVM

MLE for extended problem

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{n} p(\mathbf{x}_i, \mathbf{z}_i|\boldsymbol{\theta}) =$$

$$= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\mathbf{x}_i, \mathbf{z}_i|\boldsymbol{\theta}).$$

However, $\mathbf{Z}$ is unknown.

MLE for original problem

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \log p(\mathbf{X}|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\mathbf{x}_i|\boldsymbol{\theta}) =$$

$$= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log \int p(\mathbf{x}_i, \mathbf{z}_i|\boldsymbol{\theta}) d\mathbf{z}_i =$$

$$= \arg\max_{\boldsymbol{\theta}} \log \sum_{i=1}^{n} \int p(\mathbf{x}_i|\mathbf{z}_i, \boldsymbol{\theta}) p(\mathbf{z}_i) d\mathbf{z}_i.$$

Naive approach

Monte-Carlo estimation

$$p(\mathbf{x}|\boldsymbol{\theta}) = \int p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) p(\mathbf{z}) d\mathbf{z} = \mathbb{E}_{p(\mathbf{z})} p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) \approx \frac{1}{K} \sum_{k=1}^{K} p(\mathbf{x}|\mathbf{z}_k, \boldsymbol{\theta}),$$

where $\mathbf{z}_k \sim p(\mathbf{z})$.
Challenge: to cover the space properly, the number of samples grows exponentially with respect to dimensionality of $\mathbf{z}$.

---

image credit: https://jmtomczak.github.io/blog/4/4_VAE.html

Outline

Variational lower bound (ELBO)

Derivation 1 (inequality)

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \log \int p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} = \log \int \frac{q(\mathbf{z})}{q(\mathbf{z})} p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} =$$

$$= \log \mathbb{E}_q \left[\frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} \right] \geq \mathbb{E}_q \log \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} = \mathcal{L}(q, \boldsymbol{\theta})$$

Derivation 2 (equality)

$$\mathcal{L}(q, \boldsymbol{\theta}) = \int q(\mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} d\mathbf{z} = \int q(\mathbf{z}) \log \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}) p(\mathbf{x}|\boldsymbol{\theta})}{q(\mathbf{z})} d\mathbf{z} =$$

$$= \int q(\mathbf{z}) \log p(\mathbf{x}|\boldsymbol{\theta}) d\mathbf{z} + \int q(\mathbf{z}) \log \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})}{q(\mathbf{z})} d\mathbf{z} =$$

$$= \log p(\mathbf{x}|\boldsymbol{\theta}) - KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}))$$

Variational decomposition

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})) \geq \mathcal{L}(q, \boldsymbol{\theta}).$$

Variational lower bound (ELBO)

$$\mathcal{L}(q, \boldsymbol{\theta}) = \int q(\mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} d\mathbf{z} =$$

$$= \int q(\mathbf{z}) \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) d\mathbf{z} + \int q(\mathbf{z}) \log \frac{p(\mathbf{z})}{q(\mathbf{z})} d\mathbf{z}$$

$$= \mathbb{E}_q \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) - KL(q(\mathbf{z})||p(\mathbf{z}))$$

Log-likelihood decomposition

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}))$$
$$= \mathbb{E}_q \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) - KL(q(\mathbf{z})||p(\mathbf{z})) + KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})).$$

▶ Instead of maximizing incomplete likelihood, maximize ELBO

$$\max_{\boldsymbol{\theta}} p(\mathbf{x}|\boldsymbol{\theta}) \quad \rightarrow \quad \max_{q, \boldsymbol{\theta}} \mathcal{L}(q, \boldsymbol{\theta})$$

▶ Maximization of ELBO by **variational** distribution q is equivalent to minimization of KL

$$\arg\max_q \mathcal{L}(q, \boldsymbol{\theta}) \equiv \arg\min_q KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})).$$

Outline

EM-algorithm

$$\mathcal{L}(q, \boldsymbol{\theta}) = \mathbb{E}_q \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) - KL(q(\mathbf{z})||p(\mathbf{z})) =$$

$$= \mathbb{E}_q \left[\log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) - \log \frac{q(\mathbf{z})}{p(\mathbf{z})} \right] d\mathbf{z} \rightarrow \max_{q, \boldsymbol{\theta}}.$$

Block-coordinate optimization

▶ Initialize $\boldsymbol{\theta}^*$;

▶ **E-step** $(\mathcal{L}(q, \boldsymbol{\theta}) \rightarrow \max_q)$

$$q^*(\mathbf{z}) = \arg\max_q \mathcal{L}(q, \boldsymbol{\theta}^*) =$$

$$= \arg\min_q KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*)) = p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*);$$

▶ **M-step** $(\mathcal{L}(q, \boldsymbol{\theta}) \rightarrow \max_{\boldsymbol{\theta}})$

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(q^*, \boldsymbol{\theta});$$

▶ Repeat E-step and M-step until convergence.

EM-algorithm illustration

Bishop C. Pattern Recognition and Machine Learning, 2006
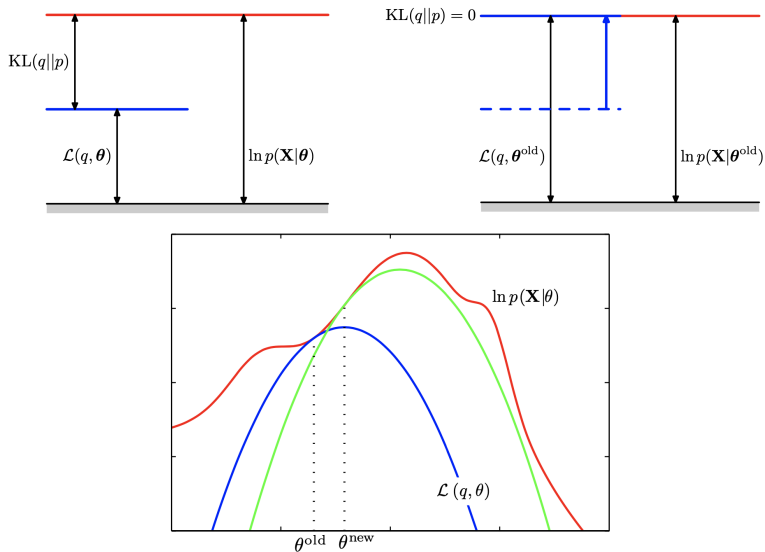
Amortized variational inference

E-step

$$q(\mathbf{z}) = \arg\max_q \mathcal{L}(q, \boldsymbol{\theta}^*) = \arg\min_q KL(q||p) = p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*).$$

- ▶ $q(\mathbf{z})$ approximates true posterior distribution $p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*)$, that is why it is called **variational posterior**;
- ▶ $p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*)$ could be **intractable**;
- ▶ $q(\mathbf{z})$ is different for each object $\mathbf{x}$.

Idea

Restrict a family of all possible distributions $q(\mathbf{z})$ to a parametric class $q(\mathbf{z}|\mathbf{x}, \phi)$ conditioned on samples $\mathbf{x}$ with parameters ϕ.

Variational Bayes

- ▶ E-step

$$\phi_k = \phi_{k-1} + \eta \nabla_\phi \mathcal{L}(\phi, \boldsymbol{\theta}_{k-1})|_{\phi = \phi_{k-1}}$$

- ▶ M-step

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\phi_k, \boldsymbol{\theta})|_{\boldsymbol{\theta} = \boldsymbol{\theta}_{k-1}}$$
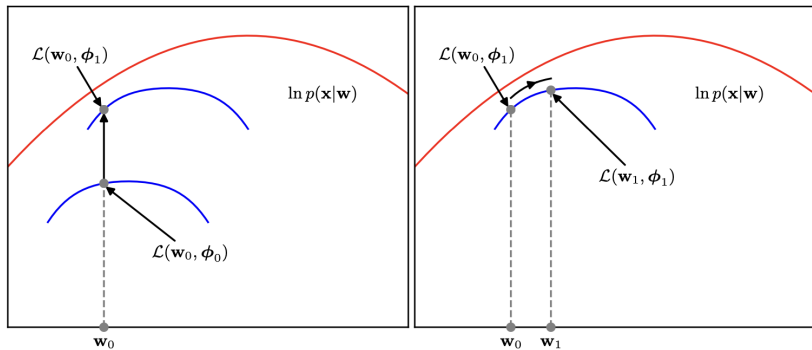
Variational EM illustration

- E-step

$$\phi_k = \phi_{k-1} + \eta \nabla_\phi \mathcal{L}(\phi, \boldsymbol{\theta}_{k-1})|_{\phi=\phi_{k-1}}$$

- M-step

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\phi_k, \boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{k-1}}$$

Bishop C. Deep Learning: Foundations and Concepts, 2024

Variational EM-algorithm

ELBO

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{L}(\phi, \boldsymbol{\theta}) + KL(q(\mathbf{z}|\mathbf{x}, \phi)||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})) \geq \mathcal{L}(\phi, \boldsymbol{\theta}).$$

- ▶ **E-step**

$$\phi_k = \phi_{k-1} + \eta \nabla_\phi \mathcal{L}(\phi, \boldsymbol{\theta}_{k-1})|_{\phi=\phi_{k-1}},$$

  where ϕ – parameters of variational posterior distribution $q(\mathbf{z}|\mathbf{x}, \phi)$.

- ▶ **M-step**

$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\phi_k, \boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{k-1}},$$

  where $\boldsymbol{\theta}$ – parameters of the generative distribution $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$.

Now all we have to do is to obtain two gradients $\nabla_\phi \mathcal{L}(\phi, \boldsymbol{\theta})$, $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\phi, \boldsymbol{\theta})$.

Challenge: Number of samples n could be huge (we heed to derive unbiased stochastic gradients).

Summary

- ▶ Bayesian framework is a generalization of most common machine learning tasks.

- ▶ LVM introduces latent representation of observed samples to make model more interpretable.

- ▶ LVM maximizes variational evidence lower bound (ELBO) to find MLE for the parameters.

- ▶ The general variational EM algorithm maximizes ELBO objective for LVM model to find MLE for parameters $\boldsymbol{\theta}$.