

# 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

$$\begin{cases} \mathbf{x}_1 = \mathbf{z}_1; \\ \mathbf{x}_2 = \mathbf{z}_2 \odot \sigma_{\theta}(\mathbf{z}_1) + \mu_{\theta}(\mathbf{z}_1). \end{cases} \quad \begin{cases} \mathbf{z}_1 = \mathbf{x}_1; \\ \mathbf{z}_2 = (\mathbf{x}_2 - \mu_{\theta}(\mathbf{x}_1)) \odot \frac{1}{\sigma_{\theta}(\mathbf{x}_1)}. \end{cases}$$

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.

## Recap of previous lecture

Consider Ordinary Differential Equation

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

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

### Euler update step

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

### Residual block

$$\mathbf{z}_{t+1} = \mathbf{z}_t + f_{\theta}(\mathbf{z}_t)$$

It is equivalent to Euler update step for solving ODE with  $\Delta t = 1$ !

In the limit of adding more layers and taking smaller steps we get:

$$\frac{d\mathbf{z}(t)}{dt} = f_{\theta}(\mathbf{z}(t), t); \quad \mathbf{z}(t_0) = \mathbf{x}; \quad \mathbf{z}(t_1) = \mathbf{y}.$$

## Recap of previous lecture

### Forward pass (loss function)

$$\begin{aligned} L(\mathbf{y}) &= L(\mathbf{z}(t_1)) = L\left(\mathbf{z}(t_0) + \int_{t_0}^{t_1} f_{\theta}(\mathbf{z}(t), t) dt\right) \\ &= L(\text{ODESolve}(\mathbf{z}(t_0), f_{\theta}, t_0, t_1)) \end{aligned}$$

**Note:** ODESolve could be any method (Euler step, Runge-Kutta methods).

### Backward pass (gradients computation)

For fitting parameters we need gradients:

$$\mathbf{a}_{\mathbf{z}}(t) = \frac{\partial L(\mathbf{y})}{\partial \mathbf{z}(t)}; \quad \mathbf{a}_{\theta}(t) = \frac{\partial L(\mathbf{y})}{\partial \theta(t)}.$$

In theory of optimal control these functions called **adjoint** functions. They show how the gradient of the loss depends on the hidden state  $\mathbf{z}(t)$  and parameters  $\theta$ .

## Recap of previous lecture

$$\mathbf{a}_z(t) = \frac{\partial L(\mathbf{y})}{\partial \mathbf{z}(t)}; \quad \mathbf{a}_\theta(t) = \frac{\partial L(\mathbf{y})}{\partial \theta(t)} - \text{adjoint functions.}$$

### Theorem (Pontryagin)

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

### Forward pass

$$\mathbf{z}(t_1) = \int_{t_0}^{t_1} f_\theta(\mathbf{z}(t), t) dt + \mathbf{z}_0 \quad \Rightarrow \quad \text{ODE Solver}$$

### Backward pass

$$\left. \begin{aligned} \frac{\partial L}{\partial \theta(t_0)} &= \mathbf{a}_\theta(t_0) = - \int_{t_1}^{t_0} \mathbf{a}_z(t)^T \frac{\partial f_\theta(\mathbf{z}(t), t)}{\partial \theta(t)} dt + 0 \\ \frac{\partial L}{\partial \mathbf{z}(t_0)} &= \mathbf{a}_z(t_0) = - \int_{t_1}^{t_0} \mathbf{a}_z(t)^T \frac{\partial f_\theta(\mathbf{z}(t), t)}{\partial \mathbf{z}(t)} dt + \frac{\partial L}{\partial \mathbf{z}(t_1)} \\ \mathbf{z}(t_0) &= - \int_{t_1}^{t_0} f_\theta(\mathbf{z}(t), t) dt + \mathbf{z}_1. \end{aligned} \right\} \Rightarrow \text{ODE Solver}$$

# Recap of previous lecture

## Continuous-in-time normalizing flows

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

## Theorem (Picard)

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

## Forward transform + log-density

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

## Hutchinson's trace estimator

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

# Outline

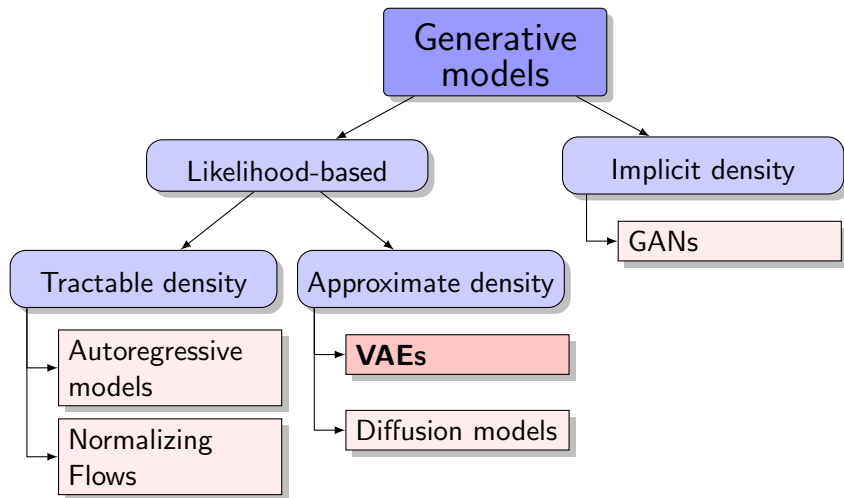
1. Latent variable models (LVM)
2. Variational lower bound (ELBO)
3. EM-algorithm, amortized inference

# Outline

1. Latent variable models (LVM)
2. Variational lower bound (ELBO)
3. EM-algorithm, amortized inference



# 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  $\mathbf{t}$  is our model parameters  $\theta$ .

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

### Posterior distribution

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

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

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

# Latent variable models (LVM)

## MLE problem

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

The distribution  $p(\mathbf{x}|\theta)$  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  $\mathbf{x}$

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

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

## Motivation

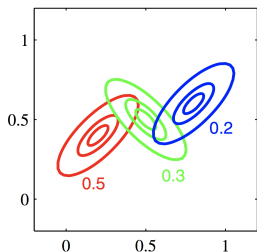
The distributions  $p(\mathbf{x}|\mathbf{z}, \theta)$  and  $p(\mathbf{z})$  could be quite simple.

# Latent variable models (LVM)

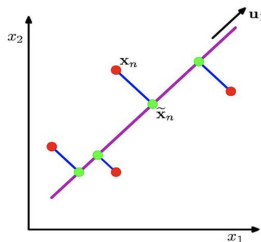
$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \log \int p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z})d\mathbf{z} \rightarrow \max_{\boldsymbol{\theta}}$$

## Examples

*Mixture of gaussians*



*PCA model*

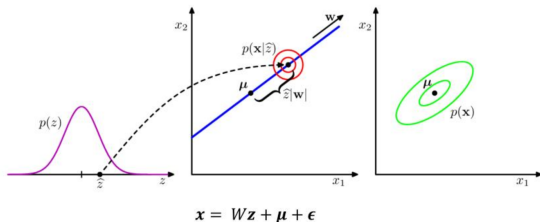


- ▶  $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$
- ▶  $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I})$
- ▶  $p(\mathbf{z}) = \text{Categorical}(\boldsymbol{\pi})$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

# Latent variable models (LVM)

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

**PCA** projects original data  $\mathbf{X}$  onto a low dimensional latent space while maximizing the variance of the projected data.



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

# Maximum likelihood estimation for LVM

## MLE for extended problem

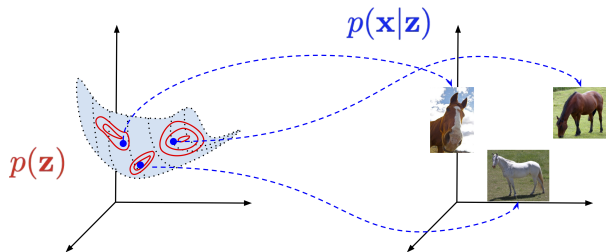
$$\begin{aligned}\theta^* &= \arg \max_{\theta} p(\mathbf{X}, \mathbf{Z} | \theta) = \arg \max_{\theta} \prod_{i=1}^n p(\mathbf{x}_i, \mathbf{z}_i | \theta) = \\ &= \arg \max_{\theta} \sum_{i=1}^n \log p(\mathbf{x}_i, \mathbf{z}_i | \theta).\end{aligned}$$

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

## MLE for original problem

$$\begin{aligned}\theta^* &= \arg \max_{\theta} \log p(\mathbf{X} | \theta) = \arg \max_{\theta} \sum_{i=1}^n \log p(\mathbf{x}_i | \theta) = \\ &= \arg \max_{\theta} \sum_{i=1}^n \log \int p(\mathbf{x}_i, \mathbf{z}_i | \theta) d\mathbf{z}_i = \\ &= \arg \max_{\theta} \log \sum_{i=1}^n \int p(\mathbf{x}_i | \mathbf{z}_i, \theta) p(\mathbf{z}_i) d\mathbf{z}_i.\end{aligned}$$

# Naive approach



## Monte-Carlo estimation

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

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  $\mathbf{z}$ .



# Outline

1. Latent variable models (LVM)
2. Variational lower bound (ELBO)
3. EM-algorithm, amortized inference

# Variational lower bound (ELBO)

## Derivation 1 (inequality)

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

## Derivation 2 (equality)

$$\begin{aligned}\mathcal{L}(q, \boldsymbol{\theta}) &= \int q(\mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} d\mathbf{z} = \int q(\mathbf{z}) \log \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}) p(\mathbf{x}|\boldsymbol{\theta})}{q(\mathbf{z})} d\mathbf{z} = \\ &= \int q(\mathbf{z}) \log p(\mathbf{x}|\boldsymbol{\theta}) d\mathbf{z} + \int q(\mathbf{z}) \log \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})}{q(\mathbf{z})} d\mathbf{z} = \\ &= \log p(\mathbf{x}|\boldsymbol{\theta}) - KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}))\end{aligned}$$

## 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)

$$\begin{aligned}\mathcal{L}(q, \theta) &= \int q(\mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{z} | \theta)}{q(\mathbf{z})} d\mathbf{z} = \\ &= \int q(\mathbf{z}) \log p(\mathbf{x} | \mathbf{z}, \theta) d\mathbf{z} + \int q(\mathbf{z}) \log \frac{p(\mathbf{z})}{q(\mathbf{z})} d\mathbf{z} \\ &= \mathbb{E}_q \log p(\mathbf{x} | \mathbf{z}, \theta) - KL(q(\mathbf{z}) || p(\mathbf{z}))\end{aligned}$$

### Log-likelihood decomposition

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

- Instead of maximizing incomplete likelihood, maximize ELBO

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

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

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

# Outline

1. Latent variable models (LVM)
2. Variational lower bound (ELBO)
3. EM-algorithm, amortized inference

# EM-algorithm

$$\begin{aligned}\mathcal{L}(q, \theta) &= \mathbb{E}_q \log p(\mathbf{x}|\mathbf{z}, \theta) - KL(q(\mathbf{z})||p(\mathbf{z})) = \\ &= \mathbb{E}_q \left[ \log p(\mathbf{x}|\mathbf{z}, \theta) - \log \frac{q(\mathbf{z})}{p(\mathbf{z})} \right] d\mathbf{z} \rightarrow \max_{q, \theta}.\end{aligned}$$

## Block-coordinate optimization

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

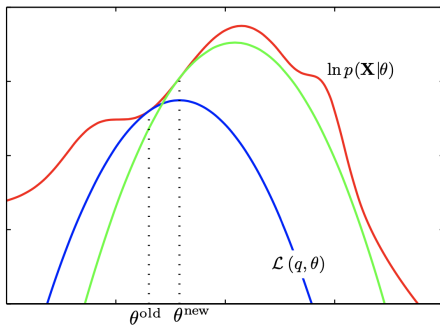
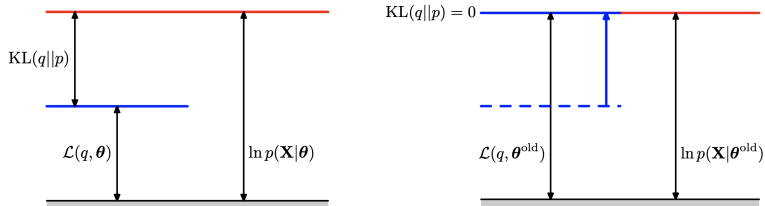
$$\begin{aligned}q^*(\mathbf{z}) &= \arg \max_q \mathcal{L}(q, \theta^*) = \\ &= \arg \min_q KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \theta^*)) = p(\mathbf{z}|\mathbf{x}, \theta^*);\end{aligned}$$

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

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

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

# EM-algorithm illustration



# Amortized variational inference

## E-step

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

- ▶  $q(\mathbf{z})$  approximates true posterior distribution  $p(\mathbf{z}|\mathbf{x}, \theta^*)$ , that is why it is called **variational posterior**;
- ▶  $p(\mathbf{z}|\mathbf{x}, \theta^*)$  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  $\phi$ .

## Variational Bayes

- ▶ E-step

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

- ▶ M-step

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

# Variational EM-algorithm

## ELBO

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

### ► E-step

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

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

### ► M-step

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

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  $\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta})$ ,  $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta})$ .

**Challenge:** Number of samples  $n$  could be huge (we need 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  $\theta$ .