

Deep Generative Models

Lecture 3

Roman Isachenko

Moscow Institute of Physics and Technology
Yandex School of Data Analysis

2024, Autumn

Recap of previous lecture

Jacobian matrix

Let $\mathbf{f} : \mathbb{R}^m \rightarrow \mathbb{R}^m$ be a differentiable function.

$$\mathbf{z} = \mathbf{f}(\mathbf{x}), \quad \mathbf{J} = \frac{\partial \mathbf{z}}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial z_1}{\partial x_1} & \cdots & \frac{\partial z_1}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_m}{\partial x_1} & \cdots & \frac{\partial z_m}{\partial x_m} \end{pmatrix} \in \mathbb{R}^{m \times m}$$

Change of variable theorem (CoV)

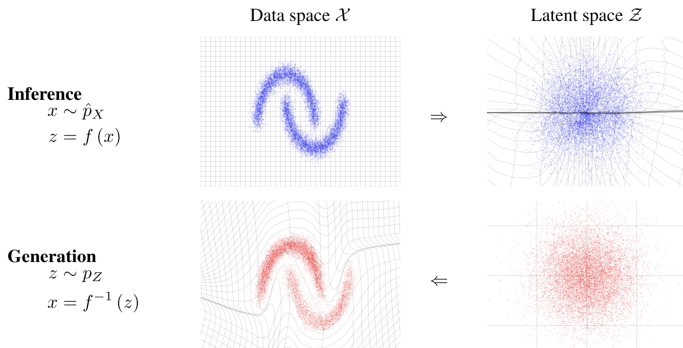
Let \mathbf{x} be a random variable with density function $p(\mathbf{x})$ and $\mathbf{f} : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is a differentiable, invertible function. If $\mathbf{z} = \mathbf{f}(\mathbf{x})$, $\mathbf{x} = \mathbf{f}^{-1}(\mathbf{z}) = \mathbf{g}(\mathbf{z})$, then

$$p(\mathbf{x}) = p(\mathbf{z}) |\det(\mathbf{J}_{\mathbf{f}})| = p(\mathbf{z}) \left| \det \left(\frac{\partial \mathbf{z}}{\partial \mathbf{x}} \right) \right| = p(\mathbf{f}(\mathbf{x})) \left| \det \left(\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right) \right|$$
$$p(\mathbf{z}) = p(\mathbf{x}) |\det(\mathbf{J}_{\mathbf{g}})| = p(\mathbf{x}) \left| \det \left(\frac{\partial \mathbf{x}}{\partial \mathbf{z}} \right) \right| = p(\mathbf{g}(\mathbf{z})) \left| \det \left(\frac{\partial \mathbf{g}(\mathbf{z})}{\partial \mathbf{z}} \right) \right|.$$

Recap of previous lecture

Definition

Normalizing flow is a *differentiable, invertible* mapping from data \mathbf{x} to the noise \mathbf{z} .



Log likelihood

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \log p(\mathbf{f}_K \circ \dots \circ \mathbf{f}_1(\mathbf{x})) + \sum_{k=1}^K \log |\det(\mathbf{J}_{\mathbf{f}_k})|$$

Recap of previous lecture

Flow log-likelihood

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \log p(\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})) + \log |\det(\mathbf{J}_{\mathbf{f}})|$$

The main challenge is a determinant of the Jacobian.

Linear flows

$$\mathbf{z} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}) = \mathbf{W}\mathbf{x}, \quad \mathbf{W} \in \mathbb{R}^{m \times m}, \quad \boldsymbol{\theta} = \mathbf{W}, \quad \mathbf{J}_{\mathbf{f}} = \mathbf{W}^T$$

- ▶ LU-decomposition

$$\mathbf{W} = \mathbf{P}\mathbf{L}\mathbf{U}.$$

- ▶ QR-decomposition

$$\mathbf{W} = \mathbf{Q}\mathbf{R}.$$

Decomposition should be done only once in the beginning. Next, we fit decomposed matrices (**P/L/U** or **Q/R**).

Kingma D. P., Dhariwal P. Glow: Generative Flow with Invertible 1x1 Convolutions, 2018

Hoogeboom E., et al. Emerging convolutions for generative normalizing flows, 2019

Recap of previous lecture

Consider an autoregressive model

$$p(\mathbf{x}|\theta) = \prod_{j=1}^m p(x_j|\mathbf{x}_{1:j-1}, \theta), \quad p(x_j|\mathbf{x}_{1:j-1}, \theta) = \mathcal{N}(\mu_{j,\theta}(\mathbf{x}_{1:j-1}), \sigma_{j,\theta}^2(\mathbf{x}_{1:j-1})).$$

Gaussian autoregressive NF

$$\mathbf{x} = \mathbf{g}_{\theta}(\mathbf{z}) \quad \Rightarrow \quad x_j = \sigma_{j,\theta}(\mathbf{x}_{1:j-1}) \cdot z_j + \mu_{j,\theta}(\mathbf{x}_{1:j-1}).$$

$$\mathbf{z} = \mathbf{f}_{\theta}(\mathbf{x}) \quad \Rightarrow \quad z_j = (x_j - \mu_{j,\theta}(\mathbf{x}_{1:j-1})) \cdot \frac{1}{\sigma_{j,\theta}(\mathbf{x}_{1:j-1})}.$$

- ▶ We have an **invertible** and **differentiable** transformation from $p(\mathbf{z})$ to $p(\mathbf{x}|\theta)$.
- ▶ Jacobian of such transformation is triangular!

Generation function $\mathbf{g}_{\theta}(\mathbf{z})$ is **sequential**.

Inference function $\mathbf{f}_{\theta}(\mathbf{x})$ is **not sequential**.

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

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

Coupling layer is a special case of autoregressive NF.

Outline

1. Forward and Reverse KL for NF
2. Latent variable models (LVM)
3. Variational lower bound (ELBO)
4. EM-algorithm
5. Amortized inference

Outline

1. Forward and Reverse KL for NF
2. Latent variable models (LVM)
3. Variational lower bound (ELBO)
4. EM-algorithm
5. Amortized inference

Forward KL vs Reverse KL

Forward KL \equiv MLE

$$\begin{aligned} KL(\pi||p) &= \int \pi(\mathbf{x}) \log \frac{\pi(\mathbf{x})}{p(\mathbf{x}|\boldsymbol{\theta})} d\mathbf{x} \\ &= -\mathbb{E}_{\pi(\mathbf{x})} \log p(\mathbf{x}|\boldsymbol{\theta}) + \text{const} \rightarrow \min_{\boldsymbol{\theta}} \end{aligned}$$

Forward KL for NF model

$$\begin{aligned} \log p(\mathbf{x}|\boldsymbol{\theta}) &= \log p(\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})) + \log |\det(\mathbf{J}_{\mathbf{f}})| \\ KL(\pi||p) &= -\mathbb{E}_{\pi(\mathbf{x})} [\log p(\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})) + \log |\det(\mathbf{J}_{\mathbf{f}})|] + \text{const} \end{aligned}$$

- ▶ We need to be able to compute $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})$ and its Jacobian.
- ▶ We need to be able to compute the density $p(\mathbf{z})$.
- ▶ We don't need to think about computing the function $\mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z}) = \mathbf{f}_{\boldsymbol{\theta}}^{-1}(\mathbf{z})$ until we want to sample from the NF.

Forward KL vs Reverse KL

Reverse KL

$$\begin{aligned}KL(p||\pi) &= \int p(\mathbf{x}|\boldsymbol{\theta}) \log \frac{p(\mathbf{x}|\boldsymbol{\theta})}{\pi(\mathbf{x})} d\mathbf{x} \\ &= \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} [\log p(\mathbf{x}|\boldsymbol{\theta}) - \log \pi(\mathbf{x})] \rightarrow \min_{\boldsymbol{\theta}}\end{aligned}$$

Reverse KL for NF model (LOTUS trick)

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \log p(\mathbf{z}) + \log |\det(\mathbf{J}_{\mathbf{f}})| = \log p(\mathbf{z}) - \log |\det(\mathbf{J}_{\mathbf{g}})|$$

$$KL(p||\pi) = \mathbb{E}_{p(\mathbf{z})} [\log p(\mathbf{z}) - \log |\det(\mathbf{J}_{\mathbf{g}})| - \log \pi(\mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z}))]$$

- ▶ We need to be able to compute $\mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z})$ and its Jacobian.
- ▶ We need to be able to sample from the density $p(\mathbf{z})$ (do not need to evaluate it) and to evaluate(!) $\pi(\mathbf{x})$.
- ▶ We don't need to think about computing the function $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})$.

Normalizing flows KL duality

Theorem

Fitting NF model $p(\mathbf{x}|\boldsymbol{\theta})$ to the target distribution $\pi(\mathbf{x})$ using forward KL (MLE) is equivalent to fitting the induced distribution $p(\mathbf{z}|\boldsymbol{\theta})$ to the base $p(\mathbf{z})$ using reverse KL:

$$\arg \min_{\boldsymbol{\theta}} KL(\pi(\mathbf{x})||p(\mathbf{x}|\boldsymbol{\theta})) = \arg \min_{\boldsymbol{\theta}} KL(p(\mathbf{z}|\boldsymbol{\theta})||p(\mathbf{z})).$$



Normalizing flows KL duality

Theorem

$$\arg \min_{\theta} KL(\pi(\mathbf{x})||p(\mathbf{x}|\theta)) = \arg \min_{\theta} KL(p(\mathbf{z}|\theta)||p(\mathbf{z})).$$

Proof

- ▶ $\mathbf{z} \sim p(\mathbf{z}), \mathbf{x} = \mathbf{g}_{\theta}(\mathbf{z}), \mathbf{x} \sim p(\mathbf{x}|\theta);$
- ▶ $\mathbf{x} \sim \pi(\mathbf{x}), \mathbf{z} = \mathbf{f}_{\theta}(\mathbf{x}), \mathbf{z} \sim p(\mathbf{z}|\theta);$

$$\log p(\mathbf{z}|\theta) = \log \pi(\mathbf{g}_{\theta}(\mathbf{z})) + \log |\det(\mathbf{J}_{\mathbf{g}})|;$$

$$\log p(\mathbf{x}|\theta) = \log p(\mathbf{f}_{\theta}(\mathbf{x})) + \log |\det(\mathbf{J}_{\mathbf{f}})|.$$

$$\begin{aligned} KL(p(\mathbf{z}|\theta)||p(\mathbf{z})) &= \mathbb{E}_{p(\mathbf{z}|\theta)} [\log p(\mathbf{z}|\theta) - \log p(\mathbf{z})] = \\ &= \mathbb{E}_{p(\mathbf{z}|\theta)} [\log \pi(\mathbf{g}_{\theta}(\mathbf{z})) + \log |\det(\mathbf{J}_{\mathbf{g}})| - \log p(\mathbf{z})] = \\ &= \mathbb{E}_{\pi(\mathbf{x})} [\log \pi(\mathbf{x}) - \log |\det(\mathbf{J}_{\mathbf{f}})| - \log p(\mathbf{f}_{\theta}(\mathbf{x}))] = \\ &= \mathbb{E}_{\pi(\mathbf{x})} [\log \pi(\mathbf{x}) - \log p(\mathbf{x}|\theta)] = KL(\pi(\mathbf{x})||p(\mathbf{x}|\theta)). \end{aligned}$$

Outline

1. Forward and Reverse KL for NF
2. Latent variable models (LVM)
3. Variational lower bound (ELBO)
4. EM-algorithm
5. Amortized inference

Bayesian framework

Bayes theorem

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

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

Meaning

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

Bayesian framework

Let consider the case, where the unobserved variables θ are the model parameters (we treat model parameters θ as random variables).

- ▶ $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$ – observed samples;
- ▶ $p(\theta)$ – prior distribution.

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.

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

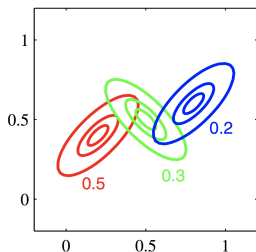
The distributions $p(\mathbf{x}|\mathbf{z}, \theta)$ and $p(\mathbf{z})$ could be quite simple.

Latent variable models (LVM)

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

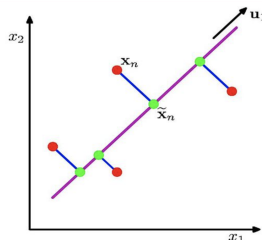
Examples

Mixture of gaussians



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

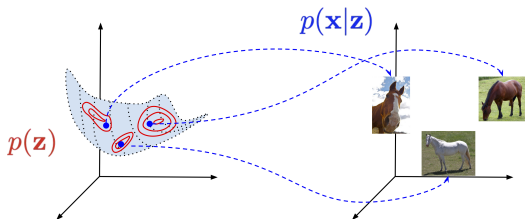
PCA model



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

MLE for LVM

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



Naive approach

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

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. Forward and Reverse KL for NF
2. Latent variable models (LVM)
3. Variational lower bound (ELBO)
4. EM-algorithm
5. Amortized inference

ELBO derivation I

Inequality derivation

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

Here $q(\mathbf{z})$ is an arbitrary distribution ($\int q(\mathbf{z}) d\mathbf{z} = 1$).

Variational lower bound (ELBO)

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

This inequality holds for every distribution $q(\mathbf{z})$.

ELBO derivation II

$$p(\mathbf{z}|\mathbf{x}, \theta) = \frac{p(\mathbf{x}, \mathbf{z}|\theta)}{p(\mathbf{x}|\theta)}$$

Equality derivation

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

Variational decomposition

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

Here $KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \theta)) \geq 0$.

Variational lower bound (ELBO)

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

Log-likelihood decomposition

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

- Instead of maximizing incomplete likelihood, maximize ELBO

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

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

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

Outline

1. Forward and Reverse KL for NF
2. Latent variable models (LVM)
3. Variational lower bound (ELBO)
4. EM-algorithm
5. Amortized inference

EM-algorithm

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

Block-coordinate optimization

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

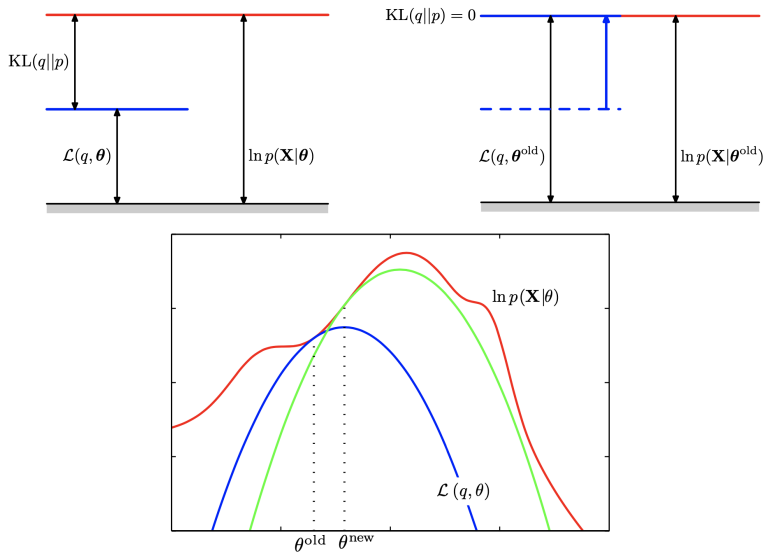
$$\begin{aligned}q^*(\mathbf{z}) &= \arg \max_q \mathcal{L}_{q,\theta^*}(\mathbf{x}) = \\ &= \arg \min_q KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \theta^*)) = p(\mathbf{z}|\mathbf{x}, \theta^*);\end{aligned}$$

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

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

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

EM-algorithm illustration



Outline

1. Forward and Reverse KL for NF
2. Latent variable models (LVM)
3. Variational lower bound (ELBO)
4. EM-algorithm
5. Amortized inference

Amortized variational inference

E-step

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

- ▶ $q(\mathbf{z})$ 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 ϕ .

Variational Bayes

- ▶ E-step

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

- ▶ M-step

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

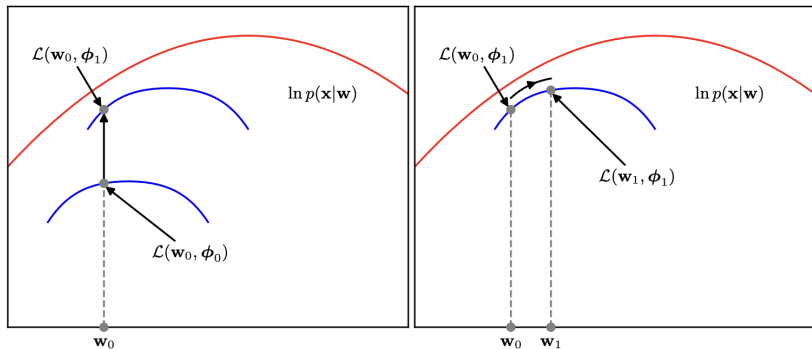
Variational EM illustration

► E-step

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

► M-step

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



Variational EM-algorithm

ELBO

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

► E-step

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

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

► M-step

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

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

Now all that is left is to obtain gradients: $\nabla_{\phi} \mathcal{L}_{\phi,\boldsymbol{\theta}}(\mathbf{x})$, $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\phi,\boldsymbol{\theta}}(\mathbf{x})$.

Challenge: Number of samples n could be huge (we need derive the **unbiased** stochastic gradients).

Summary

- ▶ Flow duality connects data space and latent space via forward and reverse KL formulations.
- ▶ Bayesian framework is a generalization of most common machine learning tasks.
- ▶ LVM introduces latent representation of observed samples to make model more interpretative.
- ▶ 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 θ .
- ▶ Amortized variational inference allows to efficiently compute the stochastic gradients for ELBO using Monte-Carlo estimation.