Deep Generative Models

Lecture 3

Roman Isachenko

Moscow Institute of Physics and Technology
Yandex School of Data Analysis

2025, Autumn

Recap of previous lecture

Jacobian matrix

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

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

Change of variables theorem (CoV)

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

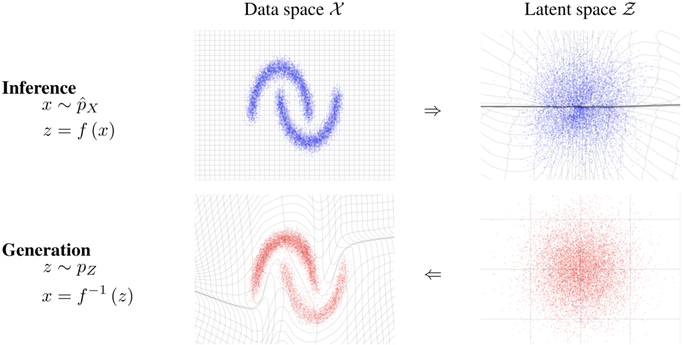
$$p(\mathbf{x}) = p(\mathbf{z}) |\det(\mathbf{J}_\mathbf{f})| = p(\mathbf{z}) \left| \det \left(\frac{\partial \mathbf{z}}{\partial \mathbf{x}} \right) \right| = p(\mathbf{f}(\mathbf{x})) \left| \det \left(\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right) \right|$$

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

Recap of previous lecture

Definition

A normalizing flow is a *differentiable, invertible* transformation from data $\mathbf{x}$ to noise $\mathbf{z}$.



Log likelihood

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

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

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 the computation of the Jacobian determinant.

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 needs to be performed only once at initialization. Then the decomposed matrices ($\mathbf{P}/\mathbf{L}/\mathbf{U}$ or $\mathbf{Q}/\mathbf{R}$) are optimized.

Kingma D. P., et al. 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:

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

Gaussian autoregressive NF

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

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

- This transformation is both **invertible** and **differentiable**, mapping $p(\mathbf{z})$ to $p(\mathbf{x}|\boldsymbol{\theta})$.
- The Jacobian matrix of this transformation is triangular.

The generative function $\mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z})$ is **sequential**, while the inference function $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})$ is **not sequential**.

Papamakarios G., Pavlakou T., Murray I. Masked Autoregressive Flow for Density Estimation, 2017

Recap of previous lecture

Let us split $\mathbf{x}$ and $\mathbf{z}$ into 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

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

Both density estimation and sampling require only a single pass!

Jacobian

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

A coupling layer is a special case of an autoregressive normalizing flow.

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

Outline

1. Forward and Reverse KL for NF

2. Latent variable models (LVM)

3. Variational lower bound (ELBO)

4. EM-algorithm

Outline

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

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

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

- We must be able to compute $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})$ and its Jacobian.
- We need access to the density $p(\mathbf{z})$.
- Computing the inverse function $\mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z}) = \mathbf{f}_{\boldsymbol{\theta}}^{-1}(\mathbf{z})$ is only necessary when sampling from the normalizing flow.

Forward KL vs Reverse KL

Reverse KL

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

Reverse KL for NF model (LOTUS trick)

$$
\log p(\mathbf{x}|\boldsymbol{\theta}) = \log p(\mathbf{z}) + \log |\det(\mathbf{J_f})| = \log p(\mathbf{z}) - \log |\det(\mathbf{J_g})|
$$

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

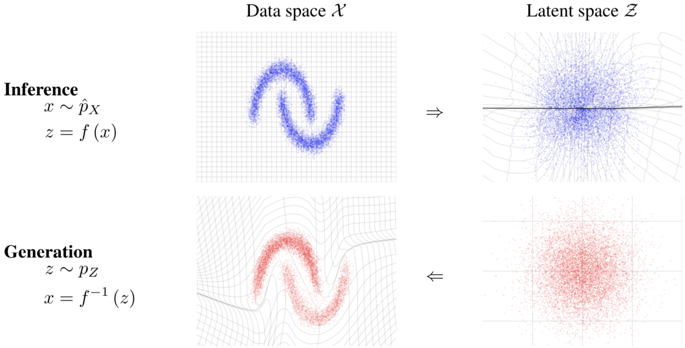
- We must be able to compute $\mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z})$ and its Jacobian.
- Sampling from $p(\mathbf{z})$ is required (but not its explicit evaluation), and we also need to evaluate $\pi(\mathbf{x})$.
- Computing $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})$ is unnecessary.

Normalizing flows KL duality

Theorem

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

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



Papamakarios G. et al. Normalizing flows for probabilistic modeling and inference, 2019

Normalizing flows KL duality

Theorem

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

Proof

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

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

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

Papamakarios G., Pavlakou T., Murray I. Masked Autoregressive Flow for Density Estimation, 2017

Outline

Bayesian framework

Bayes theorem

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

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

Interpretation

- We begin with unknown variables $\boldsymbol{\theta}$ and prior beliefs about them $p(\boldsymbol{\theta})$.
- Once data $\mathbf{x}$ has been observed, the posterior $p(\boldsymbol{\theta}|\mathbf{x})$ combines our prior knowledge and the observed data.

Bayesian framework

Consider the situation where the unobserved variables $\boldsymbol{\theta}$ correspond to model parameters (i.e., $\boldsymbol{\theta}$ are treated as random variables).

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

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 the evidence $p(\mathbf{X})$ is intractable (due to high-dimensional integration), it is not possible to compute the posterior distribution exactly.

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 $p(\mathbf{x}|\boldsymbol{\theta})$ can be highly complex and often intractable (similarly to the true data distribution $\pi(\mathbf{x})$).

Extended probabilistic model

Introduce a 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

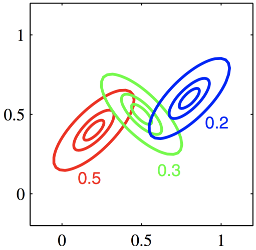
The conditional $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$ and the prior $p(\mathbf{z})$ are typically much simpler distributions.

Latent variable models (LVM)

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

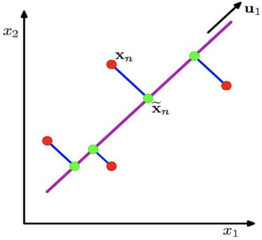
Examples

Mixture of Gaussians



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

PCA model

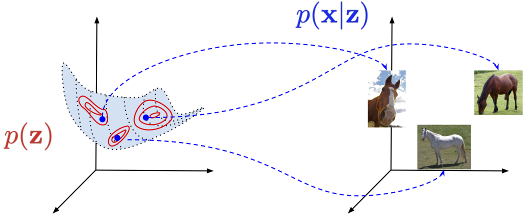


- $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$
- $p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$

Bishop C. Pattern Recognition and Machine Learning, 2006

MLE for LVM

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



Naive approach

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

where $\mathbf{z}_k \sim p(\mathbf{z})$.

Challenge: To adequately explore the latent space, the required number of samples grows exponentially with the dimensionality of $\mathbf{z}$.

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

Outline

ELBO derivation I

Inequality derivation

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

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

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

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 any choice of $q(\mathbf{z})$.

ELBO derivation II

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

Equality derivation

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

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

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

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

Variational decomposition

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

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

Variational lower bound (ELBO)

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

Log-likelihood decomposition

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

- Instead of maximizing the likelihood, maximize the ELBO:

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

- Maximizing the ELBO with respect to the **variational** distribution q is equivalent to minimizing the KL divergence

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

Outline

EM-algorithm

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

Block-coordinate optimization

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

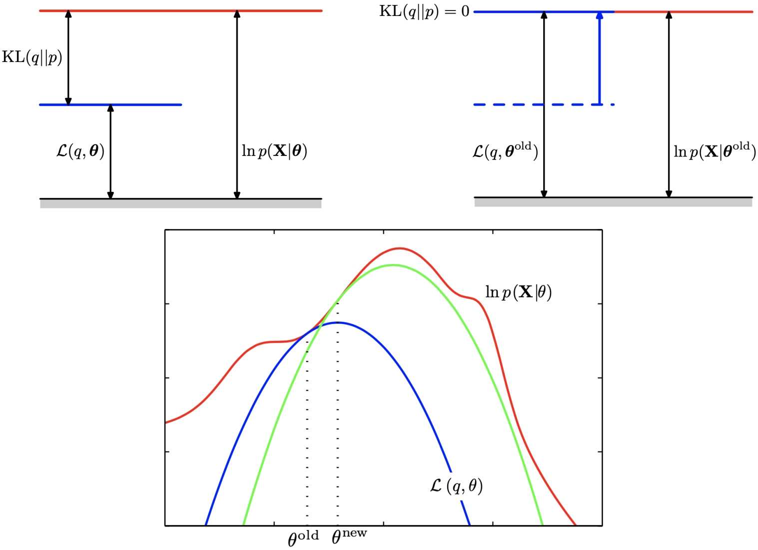
$$
\begin{aligned}
q^*(\mathbf{z}) = \arg\max_q \mathcal{L}_{q,\boldsymbol{\theta}^*}(\mathbf{x}) = \\
= \arg\min_q KL(q(\mathbf{z})||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*)) = p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*);
\end{aligned}
$$

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

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

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

EM-algorithm illustration



Bishop C. Pattern Recognition and Machine Learning, 2006

Summary

- Flow duality establishes a connection between data space and latent space using forward and reverse KL formulations.
- The Bayesian framework generalizes most standard machine learning methodologies.
- LVMs introduce latent representations of observed samples, providing more interpretable models.
- LVMs maximize the variational evidence lower bound (ELBO) to obtain maximum likelihood estimates for the parameters.
- The general variational EM algorithm maximizes the ELBO objective within LVMs to obtain the MLE for parameters $\boldsymbol{\theta}$.