Variational Autoencoder - Derivations & Proofs

Paul F. Roysdon, Ph.D.

Contents

1 Mathematical Derivations & Proofs **1**
- 1.1 Introduction . . . 1
- 1.2 Data and Notation . . . 1
- 1.3 Model Formulation and MLE Objective . . . 2
- 1.4 ELBO Derivations (Two Equivalent Proofs) . . . 3
- 1.5 Gaussian Encoder: Closed-Form KL and Reparameterization . . . 3
- 1.6 Likelihood Choices and Reconstruction Losses . . . 4
- 1.7 Gradients and Proof of Correctness (SGVB) . . . 4
- 1.8 Relation to MLE and Tightness . . . 5
- 1.9 Algorithm (VAE Training via SGVB) . . . 5
- 1.10 Extensions (Brief) . . . 5
- 1.11 Summary of Variables and Their Dimensions . . . 6
- 1.12 Summary . . . 6

1 Mathematical Derivations & Proofs

1.1 Introduction

A Variational Autoencoder (VAE) is a latent-variable generative model trained by maximizing a tractable lower bound on the log-evidence (marginal likelihood) of the data. The model posits a prior over latent codes $\mathbf{z}$ and a conditional likelihood (decoder) $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$; inference of the intractable posterior $p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x})$ is *amortized* by a recognition (encoder) distribution $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})$. We derive the *Evidence Lower BOund* (ELBO) from first principles, present complete proofs via Jensen's inequality and KL identity, give closed forms for the Gaussian case, and show how the *reparameterization trick* yields low-variance unbiased stochastic gradients.

1.2 Data and Notation

We observe a dataset

$$\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^{n}, \qquad \mathbf{x}_i \in \mathbb{R}^d.$$

The VAE specifies:

- A latent prior $p(\mathbf{z})$ on $\mathbb{R}^r$ (typically $\mathcal{N}(\mathbf{0}, \mathbf{I}_r)$).
- A decoder (generative model) $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$ with parameters $\boldsymbol{\theta}$.
- An encoder (inference model) $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})$ with parameters $\boldsymbol{\phi}$.

Neural parameterizations (one-hidden-layer written for concreteness):

$$\text{Encoder:} \quad q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) = \mathcal{N}\big(\mathbf{z};\ \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}),\ \mathrm{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{x}))\big),$$

$$\text{Decoder:} \quad p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z}) = \begin{cases} \mathcal{N}\big(\mathbf{x};\ \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{z}),\ \sigma_x^2 \mathbf{I}_d\big), & \text{Gaussian output,} \\ \text{Bernoulli}\big(\mathbf{x};\ \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{z})\big), & \text{binary output.} \end{cases}$$

Here r is the latent (bottleneck) dimension; d is the data dimension.

Generative Model

We assume that the joint distribution over data and latent variables is given by:

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

The typical choices are:

- **Prior:** $p(\mathbf{z})$ is usually chosen as a standard Gaussian

  $$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}_d),$$

  where $\mathbf{I}_d$ is the $d \times d$ identity matrix.

- **Likelihood (Decoder):** $p_\theta(\mathbf{x}|\mathbf{z})$ is a distribution over $\mathbb{R}^D$. For real-valued data one may choose a Gaussian

  $$p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}\Big(\mathbf{x}; \boldsymbol{\mu}_\theta(\mathbf{z}), \operatorname{diag}(\boldsymbol{\sigma}_\theta^2(\mathbf{z}))\Big),$$

  where $\boldsymbol{\mu}_\theta(\mathbf{z}) \in \mathbb{R}^D$ and $\boldsymbol{\sigma}_\theta^2(\mathbf{z}) \in \mathbb{R}^D_{>0}$. For binary data the likelihood may be modeled as a product of Bernoulli distributions.

Approximate Posterior (Encoder)

Since the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$ is typically intractable, we introduce an approximate posterior:

$$q_\phi(\mathbf{z}|\mathbf{x}),$$

which is also commonly chosen to be Gaussian:

$$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\Big(\mathbf{z}; \boldsymbol{\mu}_\phi(\mathbf{x}), \operatorname{diag}(\boldsymbol{\sigma}_\phi^2(\mathbf{x}))\Big).$$

Here:

- The encoder network takes $\mathbf{x} \in \mathbb{R}^D$ as input and outputs:
  - A mean vector $\boldsymbol{\mu}_\phi(\mathbf{x}) \in \mathbb{R}^d$,
  - A variance vector $\boldsymbol{\sigma}_\phi^2(\mathbf{x}) \in \mathbb{R}^d_{>0}$ (or its standard deviation $\boldsymbol{\sigma}_\phi(\mathbf{x}) \in \mathbb{R}^d_{>0}$).

1.3 Model Formulation and MLE Objective

The joint model is $p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) = p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})\, p(\mathbf{z})$. Maximum likelihood learning seeks

$$\max_{\boldsymbol{\theta}} \ \frac{1}{n} \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(\mathbf{x}_i), \qquad \log p_{\boldsymbol{\theta}}(\mathbf{x}) = \log \int p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})\, p(\mathbf{z})\, d\mathbf{z}, \tag{1}$$

but the integral is generally intractable. VAEs introduce $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})$ to approximate $p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x})$ and derive a tractable lower bound.

1.4 ELBO Derivations (Two Equivalent Proofs)

(A) Jensen's inequality (importance trick). *Proof.* Our goal is to maximize the marginal likelihood (or evidence) of the data:

$$p_\theta(\mathbf{x}) = \int_{\mathbb{R}^d} p_\theta(\mathbf{x}, \mathbf{z})\, d\mathbf{z}.$$

Because this integral is generally intractable, we derive a lower bound on $\log p_\theta(\mathbf{x})$.

We begin with:

$$\log p_\theta(\mathbf{x}) = \log \int_{\mathbb{R}^d} p_\theta(\mathbf{x}, \mathbf{z})\, d\mathbf{z}.$$

Introduce the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ (which satisfies $q_\phi(\mathbf{z}|\mathbf{x}) > 0$ whenever $p_\theta(\mathbf{x}, \mathbf{z}) > 0$):

$$\log p_\theta(\mathbf{x}) = \log \int_{\mathbb{R}^d} q_\phi(\mathbf{z}|\mathbf{x})\, \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\, d\mathbf{z}.$$

Since the logarithm is a concave function, Jensen's inequality yields:

$$\begin{aligned} \log p_{\boldsymbol{\theta}}(\mathbf{x}) &= \log \int q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})\, \frac{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}{q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})}\, d\mathbf{z} = \log \mathbb{E}_{q_{\boldsymbol{\phi}}}\left[\frac{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})}{q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})}\right] \\ &\geq \mathbb{E}_{q_{\boldsymbol{\phi}}}[\log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) - \log q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})] \;\triangleq\; \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}), \end{aligned} \tag{2}$$

where Eqn. (2) is the *Evidence Lower Bound* (ELBO). All expectations are taken with respect to the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$, where $\mathbf{z} \in \mathbb{R}^d$, and the inequality uses concavity of $\log(\cdot)$. ∎

(B) KL identity decomposition. *Proof.* Add and subtract $\log q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})$ inside the marginal:

$$\begin{aligned} \log p_{\boldsymbol{\theta}}(\mathbf{x}) &= \mathbb{E}_{q_{\boldsymbol{\phi}}}[\log p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) - \log q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})] \;+\; \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}}[\log q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) - \log p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x})]}_{\mathrm{KL}\left(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \,\|\, p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})\right)} \\ &= \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) \;+\; \mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) \,\|\, p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}))\,. \end{aligned} \tag{3}$$

Since the KL is nonnegative, $\mathcal{L}$ is a lower bound tight iff $q_{\boldsymbol{\phi}} = p_{\boldsymbol{\theta}}$. ∎

Canonical ELBO form. Using $p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}) = p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})\, p(\mathbf{z})$ in Eqn. (2):

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}\big[\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})\big] - \mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) \,\|\, p(\mathbf{z}))\,. \tag{4}$$

The first term is a *reconstruction* term; the second regularizes the encoder toward the prior.

Dataset ELBO / training objective.

$$\max_{\boldsymbol{\theta}, \boldsymbol{\phi}} \;\; \widehat{\mathcal{L}} = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}_i). \tag{5}$$

1.5 Gaussian Encoder: Closed-Form KL and Reparameterization

A key challenge in optimizing the ELBO is that the expectation is taken with respect to a distribution $q_\phi(\mathbf{z}|\mathbf{x})$ that depends on the parameters ϕ. To allow gradient backpropagation through $\mathbf{z}$, we reparameterize the sampling process.

Assume $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) = \mathcal{N}\big(\mathbf{z};\; \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}),\; \mathrm{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{x}))\big)$ and $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I}_r)$.

Closed-form KL. For diagonal Gaussians $q = \mathcal{N}(\boldsymbol{\mu}, \operatorname{diag}(\boldsymbol{\sigma}^2))$ vs. $p = \mathcal{N}(\mathbf{0}, \mathbf{I})$,

$$\mathrm{KL}(q\|p) = \frac{1}{2}\sum_{j=1}^{r}\left(\mu_j^2 + \sigma_j^2 - \log\sigma_j^2 - 1\right). \tag{6}$$

If the network outputs $\boldsymbol{\ell} = \log\boldsymbol{\sigma}^2$, then $\sigma_j^2 = \exp(\ell_j)$, where the log-variance vector $\boldsymbol{\ell}(\cdot)$ has the identity: $\boldsymbol{\ell} \triangleq \log\boldsymbol{\sigma}^2$.

Reparameterization trick. Let $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_r)$ be an auxiliary random variable and define the deterministic transform and reparameterize $\mathbf{z}$ as:

$$\mathbf{z} \;=\; \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}) \;+\; \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\mathbf{x}) \;\odot\; \boldsymbol{\epsilon}, \tag{7}$$

where $\odot$ is the Hadamard product. Then for any integrable f,

$$\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[f(\mathbf{z})] \;=\; \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,I)}[f(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}) + \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\mathbf{x}) \odot \boldsymbol{\epsilon})].$$

This reformulation permits gradients to be computed with respect to ϕ using standard backpropagation.

Proof. The pushforward of $\boldsymbol{\epsilon}$ through the affine map Eqn. (7) is $\mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\phi}}, \operatorname{diag}(\boldsymbol{\sigma}_{\boldsymbol{\phi}}^2))$; equality of expectations follows by change of variables. ■

Stochastic ELBO estimator. With a single Monte Carlo sample $\boldsymbol{\epsilon}^{(1)}$,

$$\widehat{\mathcal{L}}(\mathbf{x}) \approx \underbrace{\log p_{\boldsymbol{\theta}}\Big(\mathbf{x} \mid \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}) + \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\mathbf{x}) \odot \boldsymbol{\epsilon}^{(1)}\Big)}_{\text{reconstruction term}} \;-\; \underbrace{\mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) \parallel p(\mathbf{z}))}_{\text{regularization term}}. \tag{8}$$

Gradients w.r.t. $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ can be backpropagated through the deterministic path Eqn. (7), yielding unbiased, low-variance estimates (SGVB).

1.6 Likelihood Choices and Reconstruction Losses

Gaussian decoder. If $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z}) = \mathcal{N}(\mathbf{x};\ \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{z}),\ \sigma_x^2\mathbf{I})$ with fixed $\sigma_x^2 > 0$, then

$$\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{Z}) = -\frac{1}{2\sigma_x^2}\|\mathbf{x} - \boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{Z})\|_2^2 - \frac{d}{2}\log(2\pi\sigma_x^2),$$

so maximizing the ELBO is equivalent (up to constants) to minimizing MSE of the decoder mean plus the KL.

Bernoulli decoder. For binary $\mathbf{x}$ and $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{Z}) = \mathrm{Bernoulli}(\mathbf{x};\ \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{Z}))$ with $\boldsymbol{\pi} = \sigma(\cdot)$,

$$\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{Z}) = -\sum_{j=1}^{d}\Big[x_j\log\pi_j + (1 - x_j)\log(1 - \pi_j)\Big],$$

the (negative) cross-entropy reconstruction term.

1.7 Gradients and Proof of Correctness (SGVB)

Proof. Write the per-sample ELBO as

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,I)}\Big[\underbrace{\log p_{\boldsymbol{\theta}}\big(\mathbf{x} \mid \mathbf{z}(\mathbf{x}, \boldsymbol{\epsilon})\big)}_{R(\boldsymbol{\theta},\boldsymbol{\phi};\mathbf{x},\boldsymbol{\epsilon})}\Big] - \mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) \parallel p(\mathbf{z})).$$

Decoder gradient. Since R depends on $\boldsymbol{\theta}$ only through $\log p_{\boldsymbol{\theta}}$,

$$\nabla_{\boldsymbol{\theta}}\mathcal{L} = \mathbb{E}_{\boldsymbol{\epsilon}}\big[\nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}\big(\mathbf{x} \mid \mathbf{z}(\mathbf{x}, \boldsymbol{\epsilon})\big)\big] \,.$$

Encoder gradient. The KL has closed-form gradient via Eqn. (6); for the reconstruction term,

$$\nabla_{\boldsymbol{\phi}}\, \mathbb{E}_{\boldsymbol{\epsilon}}\big[\log p_{\boldsymbol{\theta}}\big(\mathbf{x} \mid \mathbf{z}(\mathbf{x}, \boldsymbol{\epsilon})\big)\big] = \mathbb{E}_{\boldsymbol{\epsilon}}\left[\nabla_{\mathbf{z}} \log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})\ \frac{\partial \mathbf{z}(\mathbf{x}, \boldsymbol{\epsilon})}{\partial \boldsymbol{\phi}}\right],$$

which is exactly what backpropagation computes through the path $\mathbf{x} \mapsto (\boldsymbol{\mu}_{\boldsymbol{\phi}}, \boldsymbol{\sigma}_{\boldsymbol{\phi}}) \mapsto \mathbf{z} \mapsto \log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$. Interchanging ∇ and $\mathbb{E}$ is justified by dominated convergence (bounded second moments suffice). ■

1.8 Relation to MLE and Tightness

Summing Eqn. (3) over i:

$$\frac{1}{n}\sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(\mathbf{x}_i) = \widehat{\mathcal{L}}(\boldsymbol{\theta}, \boldsymbol{\phi}) + \frac{1}{n}\sum_{i=1}^{n} \mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}_i) \,\|\, p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}_i)) \,.$$

Maximizing $\widehat{\mathcal{L}}$ w.r.t. $(\boldsymbol{\theta}, \boldsymbol{\phi})$ jointly increases the likelihood while decreasing the amortized posterior gap; the bound is tight iff $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}_i) = p_{\boldsymbol{\theta}}(\mathbf{z} \mid \mathbf{x}_i)$ for all i.

1.9 Algorithm (VAE Training via SGVB)

1. **Input:** data $\{\mathbf{x}_i\}$, latent dimension r, prior $p(\mathbf{z})$, decoder family $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$, encoder family $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})$.
2. **Initialize** parameters $(\boldsymbol{\theta}, \boldsymbol{\phi})$ (e.g., Xavier/He).
3. **Repeat** over mini-batches $\mathcal{B}$:
   (a) For each $\mathbf{x} \in \mathcal{B}$, compute encoder outputs $(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}), \boldsymbol{\ell}_{\boldsymbol{\phi}}(\mathbf{x}))$ and set $\boldsymbol{\sigma}_{\boldsymbol{\phi}} = \exp(\frac{1}{2}\boldsymbol{\ell}_{\boldsymbol{\phi}})$.
   (b) Sample $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_r)$ and set $\mathbf{z} = \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}) + \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\mathbf{x}) \odot \boldsymbol{\epsilon}$.
   (c) Compute reconstruction log-likelihood $\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$ and KL via Eqn. (6); average over the batch to obtain $\widehat{\mathcal{L}}_{\mathcal{B}}$.
   (d) Backpropagate $\nabla_{\boldsymbol{\theta},\boldsymbol{\phi}}(-\widehat{\mathcal{L}}_{\mathcal{B}})$ and update with SGD/Adam.
4. **Output:** learned generative model $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$; inference model $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})$.

1.10 Extensions (Brief)

β-VAE. Replace the KL term by $\beta\,\mathrm{KL}$:

$$\mathcal{L}_{\beta} = \mathbb{E}_q[\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})] - \beta\,\mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) \| p(\mathbf{z}))\,,$$

trading off reconstruction and regularization (information bottleneck interpretation).

Conditional VAE (cVAE). For side information $\mathbf{y}$, model $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z}, \mathbf{y})$ and $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}, \mathbf{y})$; the ELBO conditions on $\mathbf{y}$.

Discrete latents. Use a continuous relaxation (GumbelSoftmax/Concrete) for categorical $\mathbf{z}$ to enable reparameterized gradients.

Importance-weighted ELBO (IWAE). Tighter bound with K samples: $\log p(\mathbf{x}) \geq \mathbb{E}\left[\log \frac{1}{K}\sum_{k=1}^{K} \frac{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z}^{(k)})}{q_{\boldsymbol{\phi}}(\mathbf{z}^{(k)} \mid \mathbf{x})}\right]$.

1.11 Summary of Variables and Their Dimensions

- $\mathbf{x}_i \in \mathbb{R}^d$: observed data vector.
- $\mathbf{z} \in \mathbb{R}^r$: latent code; prior $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I}_r)$.
- Encoder $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x}) = \mathcal{N}\big(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}), \mathrm{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{x}))\big)$ with outputs $\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}) \in \mathbb{R}^r$, $\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{x}) \in \mathbb{R}^r$.
- Decoder $p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})$: Gaussian (mean $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{z}) \in \mathbb{R}^d$, variance σ_x^2) or Bernoulli (probs $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{z}) \in (0,1)^d$).
- ELBO $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) = \mathbb{E}_q[\log p_{\boldsymbol{\theta}}(\mathbf{x} \mid \mathbf{z})] - \mathrm{KL}(q \| p)$ (scalar).
- Reparameterization: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_r)$, $\mathbf{z} = \boldsymbol{\mu}_{\boldsymbol{\phi}} + \boldsymbol{\sigma}_{\boldsymbol{\phi}} \odot \boldsymbol{\epsilon}$.

1.12 Summary

From first principles, VAEs perform maximum likelihood learning for latent-variable models by optimizing the ELBO Eqn. (4), obtained either by Jensen's inequality or by a KL decomposition Eqn. (3). With a Gaussian encoder and standard normal prior, the KL admits the closed form Eqn. (6). The reparameterization trick Eqn. (7) converts expectations over $q_{\boldsymbol{\phi}}(\mathbf{z} \mid \mathbf{x})$ into expectations over a fixed noise source, enabling unbiased low-variance gradient estimates (SGVB) and end-to-end training with backpropagation. Likelihood choices determine the reconstruction term (Gaussian $\leftrightarrow$ MSE; Bernoulli $\leftrightarrow$ cross-entropy). Extensions like β-VAE, cVAE, discrete latents, and IWAE follow from the same variational principles.