

# Deep Generative Models

## Lecture 4

Roman Isachenko



2026, Spring

# Recap of Previous Lecture

## Latent Variable Models (LVM)

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

## MLE Problem for LVM

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

## Naive Monte Carlo Estimation

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

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

## Recap of Previous Lecture

### ELBO Derivation 1 (Inequality)

$$\log p_{\theta}(\mathbf{x}) = \log \int p_{\theta}(\mathbf{x}, \mathbf{z}) d\mathbf{z} \geq \mathbb{E}_q \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{q(\mathbf{z})} = \mathcal{L}_{q,\theta}(\mathbf{x})$$

### ELBO Derivation 2 (Equality)

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

### Variational Decomposition

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

## Recap of Previous Lecture

### Variational Evidence Lower Bound (ELBO)

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

$$\mathcal{L}_{q,\theta}(\mathbf{x}) = \int q(\mathbf{z}) \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{q(\mathbf{z})} d\mathbf{z} = \mathbb{E}_q \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q(\mathbf{z}) \| p(\mathbf{z}))$$

### Log-likelihood Decomposition

$$\log p_{\theta}(\mathbf{x}) = \mathbb{E}_q \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q(\mathbf{z}) \| p(\mathbf{z})) + \text{KL}(q(\mathbf{z}) \| p_{\theta}(\mathbf{z}|\mathbf{x})).$$

- ▶ Rather than maximizing likelihood, maximize the ELBO:

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

- ▶ Maximizing the ELBO with respect to the variational distribution  $q$  is equivalent to minimizing the KL divergence:

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

## Recap of Previous Lecture

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

### Variational Posterior

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

### Amortized Variational Inference

We restrict the family of possible distributions  $q(\mathbf{z})$  to a parametric class  $q_{\phi}(\mathbf{z}|\mathbf{x})$ , conditioned on data  $\mathbf{x}$  and parameterized by  $\phi$ .

### Gradient Update

$$\begin{bmatrix} \phi_k \\ \theta_k \end{bmatrix} = \left[ \begin{array}{l} \phi_{k-1} + \eta \cdot \nabla_{\phi} \mathcal{L}_{\phi,\theta}(\mathbf{x}) \\ \theta_{k-1} + \eta \cdot \nabla_{\theta} \mathcal{L}_{\phi,\theta}(\mathbf{x}) \end{array} \right] \Big|_{(\phi_{k-1}, \theta_{k-1})}$$

## Recap of Previous Lecture

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) \rightarrow \max_{\phi, \theta}.$$

Gradient w.r.t.  $\theta$  — Monte Carlo Estimation

$$\begin{aligned}\nabla_{\theta} \mathcal{L}_{\phi, \theta}(\mathbf{x}) &= \int q_{\phi}(\mathbf{z}|\mathbf{x}) \nabla_{\theta} \log p_{\theta}(\mathbf{x}|\mathbf{z}) d\mathbf{z} \approx \\ &\approx \nabla_{\theta} \log p_{\theta}(\mathbf{x}|\mathbf{z}^*), \quad \mathbf{z}^* \sim q_{\phi}(\mathbf{z}|\mathbf{x}).\end{aligned}$$

Gradient w.r.t.  $\phi$  — Reparameterization Trick

$$\begin{aligned}\nabla_{\phi} \mathcal{L}_{\phi, \theta}(\mathbf{x}) &= \int p(\epsilon) \nabla_{\phi} \log p_{\theta}(\mathbf{x}|\mathbf{g}_{\phi}(\mathbf{x}, \epsilon)) d\epsilon - \nabla_{\phi} \text{KL} \\ &\approx \nabla_{\phi} \log p_{\theta}(\mathbf{x}|\mathbf{g}_{\phi}(\mathbf{x}, \epsilon^*)) - \nabla_{\phi} \text{KL}\end{aligned}$$

Variational Assumption

$$p(\epsilon) = \mathcal{N}(0, \mathbf{I}); \quad q_{\phi}(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_{\phi}(\mathbf{x}), \boldsymbol{\sigma}_{\phi}^2(\mathbf{x})).$$

$$\mathbf{z} = \mathbf{g}_{\phi}(\mathbf{x}, \epsilon) = \boldsymbol{\sigma}_{\phi}(\mathbf{x}) \odot \epsilon + \boldsymbol{\mu}_{\phi}(\mathbf{x}).$$

# Recap of Previous Lecture

## Training

- ▶ Pick a batch of samples  $\{\mathbf{x}_i\}_{i=1}^B$  (here we use Monte Carlo technique).
- ▶ Compute the objective for each sample (apply the reparametrization trick):

$$\epsilon^* \sim p(\epsilon); \quad \mathbf{z}^* = \mathbf{g}_\phi(\mathbf{x}, \epsilon^*);$$

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) \approx \log p_\theta(\mathbf{x}|\mathbf{z}^*) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})).$$

- ▶ Update parameters via stochastic gradient steps with respect to  $\phi$  and  $\theta$ .

## Inference

- ▶ Sample  $\mathbf{z}^*$  from the prior  $p(\mathbf{z}) (\mathcal{N}(0, \mathbf{I}))$ ;
- ▶ Generate data from the decoder  $p_\theta(\mathbf{x}|\mathbf{z}^*)$ .

**Note:** The encoder  $q_\phi(\mathbf{z}|\mathbf{x})$  isn't needed during generation.

# Recap of Previous Lecture

$$\mathcal{L}_{q,\theta}(\mathbf{x}) = \mathbb{E}_q \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))$$

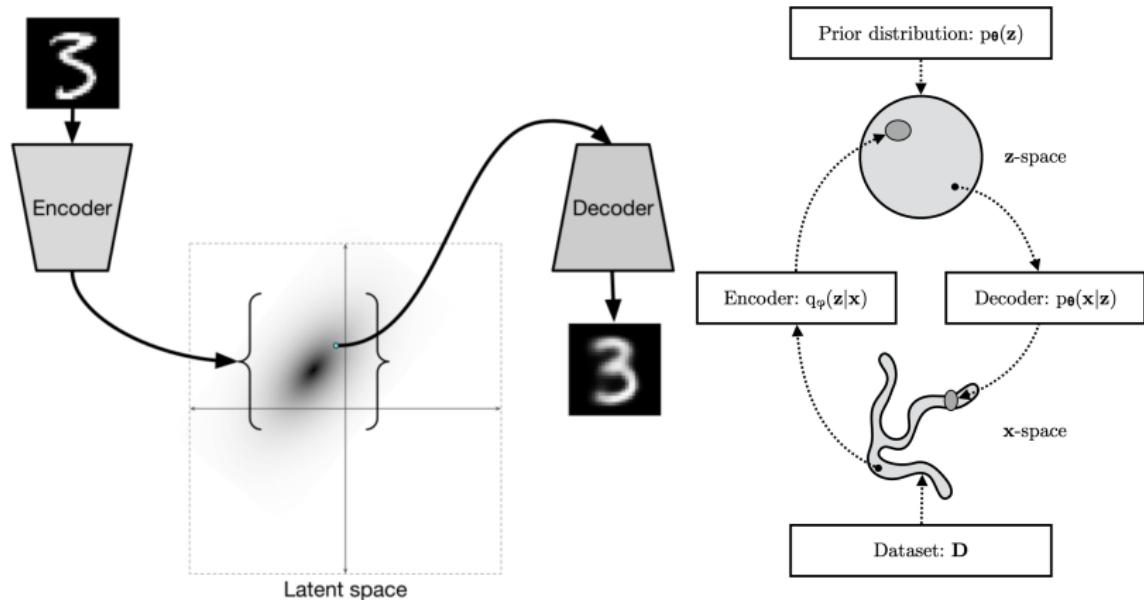


image credit: <http://ijdykeman.github.io/ml/2016/12/21/cvae.html>

Kingma D. P., Welling M. An Introduction to Variational Autoencoders, 2019

# Outline

1. Discrete VAE Latent Representations
2. Vector Quantized VAE
3. ELBO Surgery
4. Learnable VAE Prior

# Outline

1. Discrete VAE Latent Representations
2. Vector Quantized VAE
3. ELBO Surgery
4. Learnable VAE Prior

# Discrete VAE Latents

## Motivation

- ▶ Previous VAE models have used **continuous** latent variables  $\mathbf{z}$ .
- ▶ For some modalities, **discrete** representations  $\mathbf{z}$  may be a more natural choice.
- ▶ Advanced autoregressive models (e.g., PixelCNN) are highly effective for distributions over discrete variables.
- ▶ Current transformer-like models process discrete tokens.

# Discrete VAE Latents

## Motivation

- ▶ Previous VAE models have used **continuous** latent variables  $\mathbf{z}$ .
- ▶ For some modalities, **discrete** representations  $\mathbf{z}$  may be a more natural choice.
- ▶ Advanced autoregressive models (e.g., PixelCNN) are highly effective for distributions over discrete variables.
- ▶ Current transformer-like models process discrete tokens.

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) \rightarrow \max_{\phi, \theta} .$$

# Discrete VAE Latents

## Motivation

- ▶ Previous VAE models have used **continuous** latent variables  $\mathbf{z}$ .
- ▶ For some modalities, **discrete** representations  $\mathbf{z}$  may be a more natural choice.
- ▶ Advanced autoregressive models (e.g., PixelCNN) are highly effective for distributions over discrete variables.
- ▶ Current transformer-like models process discrete tokens.

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) \rightarrow \max_{\phi, \theta} .$$

- ▶ Apply the reparametrization trick to obtain unbiased gradients.
- ▶ Use Gaussian distributions for  $q_{\phi}(\mathbf{z}|\mathbf{x})$  and  $p(\mathbf{z})$  to compute the KL analytically.

# Discrete VAE Latents

## Assumptions

- ▶ Let  $c \sim \text{Cat}(\boldsymbol{\pi})$ , where

$$\boldsymbol{\pi} = (\pi_1, \dots, \pi_K), \quad \pi_k = P(c = k), \quad \sum_{k=1}^K \pi_k = 1.$$

- ▶ Suppose the VAE adopts a discrete latent variable  $c$  with prior  $p(c) = \text{Uniform}\{1, \dots, K\}$ .

# Discrete VAE Latents

## Assumptions

- ▶ Let  $c \sim \text{Cat}(\boldsymbol{\pi})$ , where

$$\boldsymbol{\pi} = (\pi_1, \dots, \pi_K), \quad \pi_k = P(c = k), \quad \sum_{k=1}^K \pi_k = 1.$$

- ▶ Suppose the VAE adopts a discrete latent variable  $c$  with prior  $p(c) = \text{Uniform}\{1, \dots, K\}$ .

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_\phi(c|\mathbf{x})} \log p_\theta(\mathbf{x}|c) - \text{KL}(q_\phi(c|\mathbf{x}) \| p(c)) \rightarrow \max_{\phi, \theta} .$$

$$\text{KL}(q_\phi(c|\mathbf{x}) \| p(c)) = \sum_{k=1}^K q_\phi(k|\mathbf{x}) \log \frac{q_\phi(k|\mathbf{x})}{p(k)}$$

# Discrete VAE Latents

## Assumptions

- ▶ Let  $c \sim \text{Cat}(\boldsymbol{\pi})$ , where

$$\boldsymbol{\pi} = (\pi_1, \dots, \pi_K), \quad \pi_k = P(c = k), \quad \sum_{k=1}^K \pi_k = 1.$$

- ▶ Suppose the VAE adopts a discrete latent variable  $c$  with prior  $p(c) = \text{Uniform}\{1, \dots, K\}$ .

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_\phi(c|\mathbf{x})} \log p_\theta(\mathbf{x}|c) - \text{KL}(q_\phi(c|\mathbf{x}) \| p(c)) \rightarrow \max_{\phi, \theta} .$$

$$\begin{aligned} \text{KL}(q_\phi(c|\mathbf{x}) \| p(c)) &= \sum_{k=1}^K q_\phi(k|\mathbf{x}) \log \frac{q_\phi(k|\mathbf{x})}{p(k)} = \\ &= \sum_{k=1}^K q_\phi(k|\mathbf{x}) \log q_\phi(k|\mathbf{x}) - \sum_{k=1}^K q_\phi(k|\mathbf{x}) \log p(k) \end{aligned}$$

# Discrete VAE Latents

## Assumptions

- ▶ Let  $c \sim \text{Cat}(\boldsymbol{\pi})$ , where

$$\boldsymbol{\pi} = (\pi_1, \dots, \pi_K), \quad \pi_k = P(c = k), \quad \sum_{k=1}^K \pi_k = 1.$$

- ▶ Suppose the VAE adopts a discrete latent variable  $c$  with prior  $p(c) = \text{Uniform}\{1, \dots, K\}$ .

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_\phi(c|\mathbf{x})} \log p_\theta(\mathbf{x}|c) - \text{KL}(q_\phi(c|\mathbf{x}) \| p(c)) \rightarrow \max_{\phi, \theta} .$$

$$\text{KL}(q_\phi(c|\mathbf{x}) \| p(c)) = \sum_{k=1}^K q_\phi(k|\mathbf{x}) \log \frac{q_\phi(k|\mathbf{x})}{p(k)} =$$

$$= \sum_{k=1}^K q_\phi(k|\mathbf{x}) \log q_\phi(k|\mathbf{x}) - \sum_{k=1}^K q_\phi(k|\mathbf{x}) \log p(k) =$$

$$= -H(q_\phi(c|\mathbf{x})) + \log K.$$

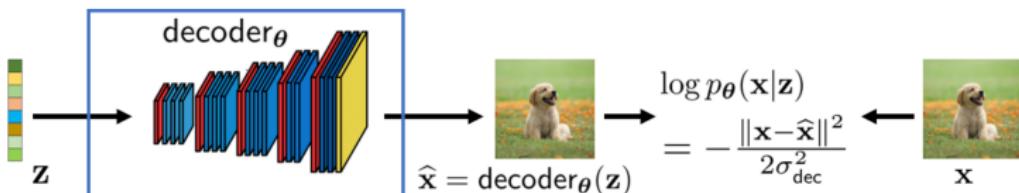
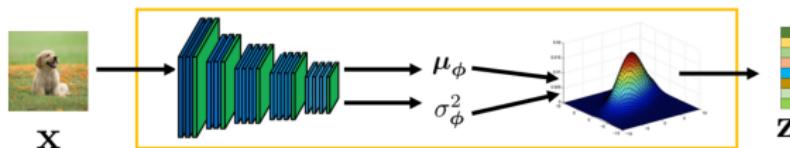
## Discrete VAE Latents

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(c|\mathbf{x})} \log p_{\theta}(\mathbf{x}|c) + H(q_{\phi}(c|\mathbf{x})) - \log K \rightarrow \max_{\phi, \theta} .$$

# Discrete VAE Latents

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(c|\mathbf{x})} \log p_{\theta}(\mathbf{x}|c) + H(q_{\phi}(c|\mathbf{x})) - \log K \rightarrow \max_{\phi, \theta} .$$

- ▶ The encoder should output a discrete distribution  $q_{\phi}(c|\mathbf{x})$ .
- ▶ We need an analogue of the reparametrization trick for discrete  $q_{\phi}(c|\mathbf{x})$ .
- ▶ The decoder  $p_{\theta}(\mathbf{x}|c)$  must take a discrete random variable  $c$  as input.



# Outline

1. Discrete VAE Latent Representations
2. Vector Quantized VAE
3. ELBO Surgery
4. Learnable VAE Prior

## Vector Quantization

Define the codebook (dictionary) space  $\{\mathbf{e}_k\}_{k=1}^K$  with  $\mathbf{e}_k \in \mathbb{R}^L$  and  $K$  the number of codebook entries.

## Vector Quantization

Define the codebook (dictionary) space  $\{\mathbf{e}_k\}_{k=1}^K$  with  $\mathbf{e}_k \in \mathbb{R}^L$  and  $K$  the number of codebook entries.

### Quantized Representation

A quantized vector  $\mathbf{z}_q \in \mathbb{R}^L$ , for any  $\mathbf{z} \in \mathbb{R}^L$ , is defined via nearest-neighbor lookup in the codebook:

$$\mathbf{z}_q = \mathbf{q}(\mathbf{z}) = \mathbf{e}_{k^*}, \quad \text{where } k^* = \arg \min_k \|\mathbf{z} - \mathbf{e}_k\|.$$

# Vector Quantization

Define the codebook (dictionary) space  $\{\mathbf{e}_k\}_{k=1}^K$  with  $\mathbf{e}_k \in \mathbb{R}^L$  and  $K$  the number of codebook entries.

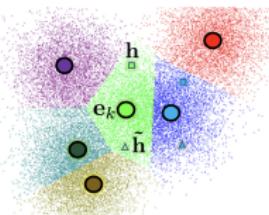
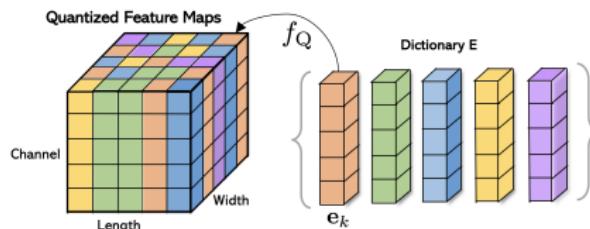
## Quantized Representation

A quantized vector  $\mathbf{z}_q \in \mathbb{R}^L$ , for any  $\mathbf{z} \in \mathbb{R}^L$ , is defined via nearest-neighbor lookup in the codebook:

$$\mathbf{z}_q = \mathbf{q}(\mathbf{z}) = \mathbf{e}_{k^*}, \quad \text{where } k^* = \arg \min_k \|\mathbf{z} - \mathbf{e}_k\|.$$

## Quantization Procedure

If the encoded tensor has spatial dimensions, quantization is independently applied to each of the  $W \times H$  locations.



## Vector Quantized VAE (VQ-VAE)

- ▶ The encoder outputs a continuous vector  $\mathbf{z}_e = \text{NN}_{\mathbf{e}, \phi}(\mathbf{x}) \in \mathbb{R}^L$ .
- ▶ Quantization deterministically maps  $\mathbf{z}_e$  to its quantized codebook vector  $\mathbf{z}_q$ .
- ▶ The decoder is conditioned on codebook entries  $\mathbf{e}_c$ , i.e., via  $p_{\theta}(\mathbf{x}|\mathbf{e}_c)$  (instead of  $p_{\theta}(\mathbf{x}|c)$ ).

## Vector Quantized VAE (VQ-VAE)

- ▶ The encoder outputs a continuous vector  $\mathbf{z}_e = \text{NN}_{\mathbf{e}, \phi}(\mathbf{x}) \in \mathbb{R}^L$ .
- ▶ Quantization deterministically maps  $\mathbf{z}_e$  to its quantized codebook vector  $\mathbf{z}_q$ .
- ▶ The decoder is conditioned on codebook entries  $\mathbf{e}_c$ , i.e., via  $p_{\theta}(\mathbf{x}|\mathbf{e}_c)$  (instead of  $p_{\theta}(\mathbf{x}|c)$ ).

### Deterministic Variational Posterior

$$q_{\phi}(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{for } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

## Vector Quantized VAE (VQ-VAE)

- ▶ The encoder outputs a continuous vector  $\mathbf{z}_e = \text{NN}_{\mathbf{e}, \phi}(\mathbf{x}) \in \mathbb{R}^L$ .
- ▶ Quantization deterministically maps  $\mathbf{z}_e$  to its quantized codebook vector  $\mathbf{z}_q$ .
- ▶ The decoder is conditioned on codebook entries  $\mathbf{e}_c$ , i.e., via  $p_{\theta}(\mathbf{x}|\mathbf{e}_c)$  (instead of  $p_{\theta}(\mathbf{x}|c)$ ).

### Deterministic Variational Posterior

$$q_{\phi}(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{for } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{KL}(q_{\phi}(c|\mathbf{x}) \| p(c)) = - \underbrace{\mathbb{H}(q_{\phi}(c|\mathbf{x}))}_{=0} + \log K = \log K.$$

## Vector Quantized VAE (VQ-VAE)

- ▶ The encoder outputs a continuous vector  $\mathbf{z}_e = \text{NN}_{\mathbf{e}, \phi}(\mathbf{x}) \in \mathbb{R}^L$ .
- ▶ Quantization deterministically maps  $\mathbf{z}_e$  to its quantized codebook vector  $\mathbf{z}_q$ .
- ▶ The decoder is conditioned on codebook entries  $\mathbf{e}_c$ , i.e., via  $p_\theta(\mathbf{x}|\mathbf{e}_c)$  (instead of  $p_\theta(\mathbf{x}|c)$ ).

### Deterministic Variational Posterior

$$q_\phi(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{for } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{KL}(q_\phi(c|\mathbf{x}) \| p(c)) = - \underbrace{\mathbb{H}(q_\phi(c|\mathbf{x}))}_{=0} + \log K = \log K.$$

**Note:** The KL regularizer becomes constant and has no direct effect on the ELBO objective in this case.

## Vector Quantized VAE (VQ-VAE): Forward

### Deterministic Variational Posterior

$$q_{\phi}(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{if } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

# Vector Quantized VAE (VQ-VAE): Forward

## Deterministic Variational Posterior

$$q_{\phi}(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{if } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(c | \mathbf{x})} \log p_{\theta}(\mathbf{x} | \mathbf{e}_c) - \log K$$

## Vector Quantized VAE (VQ-VAE): Forward

### Deterministic Variational Posterior

$$q_{\phi}(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{if } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

### ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(c | \mathbf{x})} \log p_{\theta}(\mathbf{x} | \mathbf{e}_c) - \log K = \log p_{\theta}(\mathbf{x} | \mathbf{z}_q) - \log K,$$

where  $\mathbf{z}_q = \mathbf{e}_{k^*}$ ,  $k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|$ .

# Vector Quantized VAE (VQ-VAE): Forward

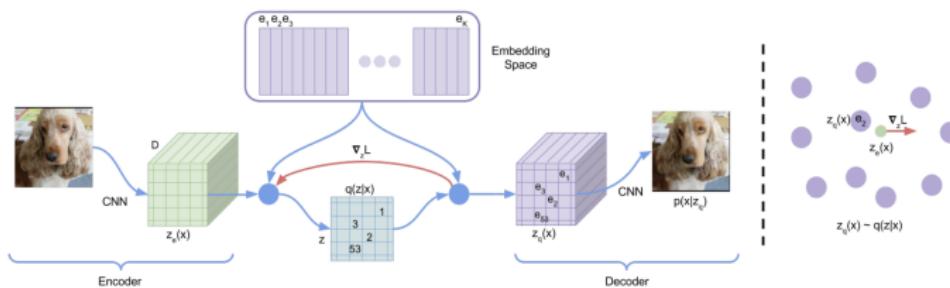
## Deterministic Variational Posterior

$$q_{\phi}(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{if } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(c|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{e}_c) - \log K = \log p_{\theta}(\mathbf{x}|\mathbf{z}_q) - \log K,$$

where  $\mathbf{z}_q = \mathbf{e}_{k^*}$ ,  $k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|$ .



# Vector Quantized VAE (VQ-VAE): Forward

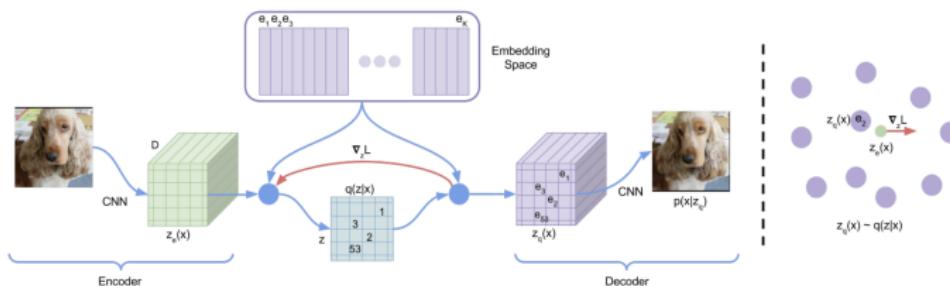
## Deterministic Variational Posterior

$$q_{\phi}(c = k^* | \mathbf{x}) = \begin{cases} 1, & \text{if } k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|; \\ 0, & \text{otherwise.} \end{cases}$$

## ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(c|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{e}_c) - \log K = \log p_{\theta}(\mathbf{x}|\mathbf{z}_q) - \log K,$$

where  $\mathbf{z}_q = \mathbf{e}_{k^*}$ ,  $k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|$ .



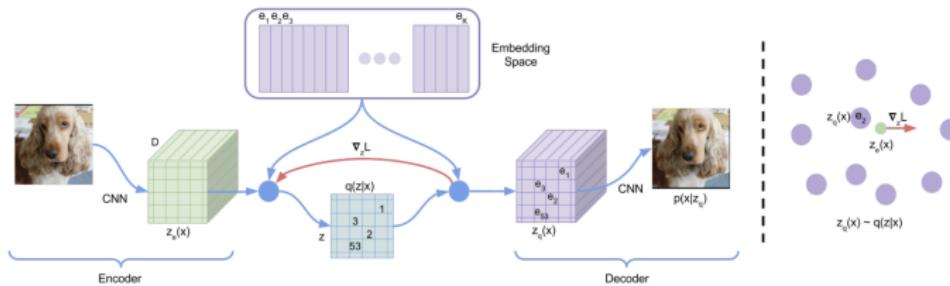
**Challenge:** The  $\arg \min$  operation is non-differentiable.

## Vector Quantized VAE (VQ-VAE): Backward ELBO

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \log p_{\theta}(\mathbf{x}|\mathbf{z}_q) - \log K, \quad \mathbf{z}_q = \mathbf{e}_{k^*}, \quad k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|.$$

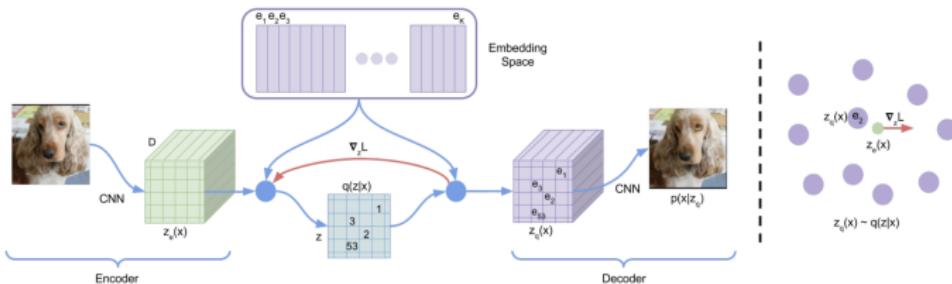
# Vector Quantized VAE (VQ-VAE): Backward ELBO

$$\mathcal{L}_{\phi, \theta}(x) = \log p_{\theta}(x|z_q) - \log K, \quad z_q = \mathbf{e}_{k^*}, \quad k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|.$$



# Vector Quantized VAE (VQ-VAE): Backward ELBO

$$\mathcal{L}_{\phi, \theta}(x) = \log p_{\theta}(x|z_q) - \log K, \quad z_q = e_{k^*}, \quad k^* = \arg \min_k \|z_e - e_k\|.$$

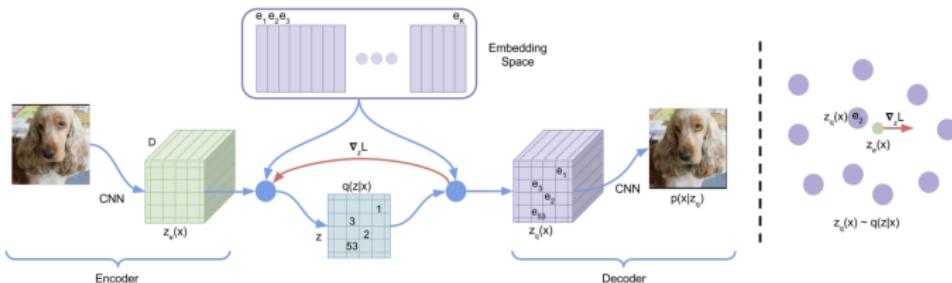


## Straight-Through Gradient Estimator

$$\frac{\partial \log p(x|z_q, \theta)}{\partial \phi} = \frac{\partial \log p_{\theta}(x|z_q)}{\partial z_q} \cdot \frac{\partial z_q}{\partial \phi} =$$

# Vector Quantized VAE (VQ-VAE): Backward ELBO

$$\mathcal{L}_{\phi, \theta}(x) = \log p_{\theta}(x|z_q) - \log K, \quad z_q = e_{k^*}, \quad k^* = \arg \min_k \|z_e - e_k\|.$$

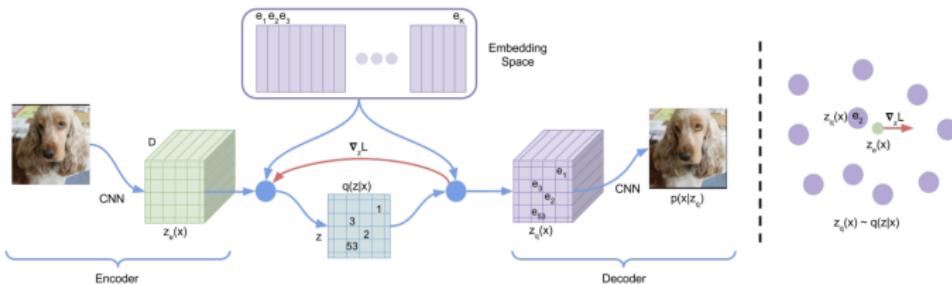


## Straight-Through Gradient Estimator

$$\begin{aligned} \frac{\partial \log p(x|z_q, \theta)}{\partial \phi} &= \frac{\partial \log p_{\theta}(x|z_q)}{\partial z_q} \cdot \frac{\partial z_q}{\partial \phi} = \\ &= \frac{\partial \log p_{\theta}(x|z_q)}{\partial z_q} \cdot \frac{\partial z_q}{\partial z_e} \cdot \frac{\partial z_e}{\partial \phi} \end{aligned}$$

# Vector Quantized VAE (VQ-VAE): Backward ELBO

$$\mathcal{L}_{\phi, \theta}(x) = \log p_{\theta}(x|z_q) - \log K, \quad z_q = \mathbf{e}_{k^*}, \quad k^* = \arg \min_k \|\mathbf{z}_e - \mathbf{e}_k\|.$$



## Straight-Through Gradient Estimator

$$\begin{aligned} \frac{\partial \log p(\mathbf{x}|z_q, \theta)}{\partial \phi} &= \frac{\partial \log p_{\theta}(\mathbf{x}|z_q)}{\partial z_q} \cdot \frac{\partial z_q}{\partial \phi} = \\ &= \frac{\partial \log p_{\theta}(\mathbf{x}|z_q)}{\partial z_q} \cdot \frac{\partial z_q}{\partial z_e} \cdot \frac{\partial z_e}{\partial \phi} \approx \frac{\partial \log p_{\theta}(\mathbf{x}|z_q)}{\partial z_q} \cdot \frac{\partial z_e}{\partial \phi} \end{aligned}$$

# Vector Quantized VAE-2 (VQ-VAE-2)

Extension to the spatial domain:  $\mathbf{c} \in \{1, \dots, K\}^{W \times H}$

$$q_{\phi}(\mathbf{c}|\mathbf{x}) = \prod_{i=1}^W \prod_{j=1}^H q(c_{ij}|\mathbf{x}, \phi); \quad p(\mathbf{c}) = \prod_{i=1}^W \prod_{j=1}^H \text{Uniform}\{1, \dots, K\}.$$

## Sample Diversity



**VQ-VAE (Proposed)**

**BigGAN deep**

# Vector Quantized VAE (VQ-VAE): Final algorithm

## Training

- ▶ Vector-quantize (per spatial location if applicable):

$$k^* = \arg \min_k \|z_e - e_k\|_2, \quad z_q = e_{k^*}, \quad z_e = \text{NN}_{e,\phi}(x).$$

- ▶ Compute ELBO objective:

$$\mathcal{L}_{\phi,\theta}(x) = \log p_\theta(x|z_q) - \log K.$$

- ▶ Compute total loss with codebook and commitment losses (with stop-gradient  $\text{sg}[\cdot]$ ):

$$\mathcal{L} = -\mathcal{L}_{\phi,\theta}(x) + \| \text{sg}[z_e] - e_{k^*} \|_2^2 + \beta \| z_e - \text{sg}[e_{k^*}] \|_2^2$$

- ▶ Use straight-through gradient estimation for encoder.

# Vector Quantized VAE (VQ-VAE): Final algorithm

## Training

- ▶ Vector-quantize (per spatial location if applicable):

$$k^* = \arg \min_k \|z_e - e_k\|_2, \quad z_q = e_{k^*}, \quad z_e = \text{NN}_{e,\phi}(x).$$

- ▶ Compute ELBO objective:

$$\mathcal{L}_{\phi,\theta}(x) = \log p_\theta(x|z_q) - \log K.$$

- ▶ Compute total loss with codebook and commitment losses (with stop-gradient  $\text{sg}[\cdot]$ ):

$$\mathcal{L} = -\mathcal{L}_{\phi,\theta}(x) + \|\text{sg}[z_e] - e_{k^*}\|_2^2 + \beta \|z_e - \text{sg}[e_{k^*}]\|_2^2$$

- ▶ Use straight-through gradient estimation for encoder.

## Sampling

- ▶ Sample  $c \sim p(c) = \text{Uniform}\{1, \dots, K\}$ .
- ▶ Generate  $x \sim p_\theta(x|e_c)$ .

# Outline

1. Discrete VAE Latent Representations
2. Vector Quantized VAE
3. ELBO Surgery
4. Learnable VAE Prior

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) = \frac{1}{n} \sum_{i=1}^n \left[ \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) \right].$$

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) = \frac{1}{n} \sum_{i=1}^n \left[ \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) \right].$$

## Theorem

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \text{KL}(q_{\text{agg}, \phi}(\mathbf{z}) \| p(\mathbf{z})) + \mathbb{I}_q[\mathbf{x}, \mathbf{z}];$$

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) = \frac{1}{n} \sum_{i=1}^n \left[ \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) \right].$$

## Theorem

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \text{KL}(q_{\text{agg}, \phi}(\mathbf{z}) \| p(\mathbf{z})) + \mathbb{I}_q[\mathbf{x}, \mathbf{z}];$$

- ▶  $q_{\text{agg}, \phi}(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z}|\mathbf{x}_i)$  denotes the **aggregated** variational posterior.
- ▶  $\mathbb{I}_q[\mathbf{x}, \mathbf{z}]$  is the mutual information between  $\mathbf{x}$  and  $\mathbf{z}$  under the data distribution  $p_{\text{data}}(\mathbf{x})$  and  $q_\phi(\mathbf{z}|\mathbf{x})$ .
- ▶ The first term encourages  $q_{\text{agg}, \phi}(\mathbf{z})$  to match the prior  $p(\mathbf{z})$ .
- ▶ The second term reduces the information about  $\mathbf{x}$  encoded in  $\mathbf{z}$ .

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \text{KL}(q_{\text{agg},\phi}(\mathbf{z}) \| p(\mathbf{z})) + \mathbb{I}_q[\mathbf{x}, \mathbf{z}].$$

## Proof

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z})} d\mathbf{z}$$

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \text{KL}(q_{\text{agg},\phi}(\mathbf{z}) \| p(\mathbf{z})) + \mathbb{I}_q[\mathbf{x}, \mathbf{z}].$$

## Proof

$$\begin{aligned} \frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z})} d\mathbf{z} = \\ &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_{\text{agg},\phi}(\mathbf{z}) q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z}) q_{\text{agg},\phi}(\mathbf{z})} d\mathbf{z} \end{aligned}$$

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \text{KL}(q_{\text{agg},\phi}(\mathbf{z}) \| p(\mathbf{z})) + \mathbb{I}_q[\mathbf{x}, \mathbf{z}].$$

## Proof

$$\begin{aligned} \frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z})} d\mathbf{z} = \\ &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_{\text{agg},\phi}(\mathbf{z}) q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z}) q_{\text{agg},\phi}(\mathbf{z})} d\mathbf{z} = \\ &= \int \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_{\text{agg},\phi}(\mathbf{z})}{p(\mathbf{z})} d\mathbf{z} + \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{q_{\text{agg},\phi}(\mathbf{z})} d\mathbf{z} \end{aligned}$$

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \text{KL}(q_{\text{agg},\phi}(\mathbf{z}) \| p(\mathbf{z})) + \mathbb{I}_q[\mathbf{x}, \mathbf{z}].$$

## Proof

$$\begin{aligned} \frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z})} d\mathbf{z} = \\ &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_{\text{agg},\phi}(\mathbf{z}) q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z}) q_{\text{agg},\phi}(\mathbf{z})} d\mathbf{z} = \\ &= \int \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_{\text{agg},\phi}(\mathbf{z})}{p(\mathbf{z})} d\mathbf{z} + \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{q_{\text{agg},\phi}(\mathbf{z})} d\mathbf{z} = \\ &= \text{KL}(q_{\text{agg},\phi}(\mathbf{z}) \| p(\mathbf{z})) + \frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| q_{\text{agg},\phi}(\mathbf{z})) \end{aligned}$$

# ELBO Surgery

$$\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) = \text{KL}(q_{\text{agg},\phi}(\mathbf{z}) \| p(\mathbf{z})) + \mathbb{I}_q[\mathbf{x}, \mathbf{z}].$$

## Proof

$$\begin{aligned}\frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z})) &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z})} d\mathbf{z} = \\ &= \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_{\text{agg},\phi}(\mathbf{z}) q_\phi(\mathbf{z}|\mathbf{x}_i)}{p(\mathbf{z}) q_{\text{agg},\phi}(\mathbf{z})} d\mathbf{z} = \\ &= \int \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_{\text{agg},\phi}(\mathbf{z})}{p(\mathbf{z})} d\mathbf{z} + \frac{1}{n} \sum_{i=1}^n \int q_\phi(\mathbf{z}|\mathbf{x}_i) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}_i)}{q_{\text{agg},\phi}(\mathbf{z})} d\mathbf{z} = \\ &= \text{KL}(q_{\text{agg},\phi}(\mathbf{z}) \| p(\mathbf{z})) + \frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| q_{\text{agg},\phi}(\mathbf{z})) \\ \mathbb{I}_q[\mathbf{x}, \mathbf{z}] &= \frac{1}{n} \sum_{i=1}^n \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| q_{\text{agg},\phi}(\mathbf{z})).\end{aligned}$$

# ELBO Surgery

## Revisiting the ELBO

$$\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) = \frac{1}{n} \sum_{i=1}^n [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i) \| p(\mathbf{z}))]$$

# ELBO Surgery

## Revisiting the ELBO

$$\begin{aligned} \frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) &= \frac{1}{n} \sum_{i=1}^n [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i)\|p(\mathbf{z}))] = \\ &= \underbrace{\frac{1}{n} \sum_{i=1}^n \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z})}_{\text{Reconstruction Loss}} - \underbrace{\mathbb{I}_q[\mathbf{x}, \mathbf{z}]}_{\text{Mutual Information}} - \underbrace{\text{KL}(q_{\text{agg}, \phi}(\mathbf{z})\|p(\mathbf{z}))}_{\text{Marginal KL}} \end{aligned}$$

# ELBO Surgery

## Revisiting the ELBO

$$\begin{aligned} \frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) &= \frac{1}{n} \sum_{i=1}^n [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i)\|p(\mathbf{z}))] = \\ &= \underbrace{\frac{1}{n} \sum_{i=1}^n \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z})}_{\text{Reconstruction Loss}} - \underbrace{\mathbb{I}_q[\mathbf{x}, \mathbf{z}]}_{\text{Mutual Information}} - \underbrace{\text{KL}(q_{\text{agg}, \phi}(\mathbf{z})\|p(\mathbf{z}))}_{\text{Marginal KL}} \end{aligned}$$

The prior distribution  $p(\mathbf{z})$  only appears in the last term.

# ELBO Surgery

## Revisiting the ELBO

$$\begin{aligned}\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) &= \frac{1}{n} \sum_{i=1}^n [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i)\|p(\mathbf{z}))] = \\ &= \underbrace{\frac{1}{n} \sum_{i=1}^n \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z})}_{\text{Reconstruction Loss}} - \underbrace{\mathbb{I}_q[\mathbf{x}, \mathbf{z}]}_{\text{Mutual Information}} - \underbrace{\text{KL}(q_{\text{agg}, \phi}(\mathbf{z})\|p(\mathbf{z}))}_{\text{Marginal KL}}\end{aligned}$$

The prior distribution  $p(\mathbf{z})$  only appears in the last term.

## Optimal VAE Prior

$$\text{KL}(q_{\text{agg}, \phi}(\mathbf{z})\|p(\mathbf{z})) = 0 \Leftrightarrow p(\mathbf{z}) = q_{\text{agg}, \phi}(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z}|\mathbf{x}_i).$$

# ELBO Surgery

## Revisiting the ELBO

$$\begin{aligned}\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) &= \frac{1}{n} \sum_{i=1}^n [\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z}) - \text{KL}(q_\phi(\mathbf{z}|\mathbf{x}_i)\|p(\mathbf{z}))] = \\ &= \underbrace{\frac{1}{n} \sum_{i=1}^n \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)} \log p_\theta(\mathbf{x}_i|\mathbf{z})}_{\text{Reconstruction Loss}} - \underbrace{\mathbb{I}_q[\mathbf{x}, \mathbf{z}]}_{\text{Mutual Information}} - \underbrace{\text{KL}(q_{\text{agg}, \phi}(\mathbf{z})\|p(\mathbf{z}))}_{\text{Marginal KL}}\end{aligned}$$

The prior distribution  $p(\mathbf{z})$  only appears in the last term.

## Optimal VAE Prior

$$\text{KL}(q_{\text{agg}, \phi}(\mathbf{z})\|p(\mathbf{z})) = 0 \Leftrightarrow p(\mathbf{z}) = q_{\text{agg}, \phi}(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z}|\mathbf{x}_i).$$

Hence, the optimal prior  $p(\mathbf{z})$  is the aggregated variational posterior  $q_{\text{agg}, \phi}(\mathbf{z})$ .

## Marginal KL

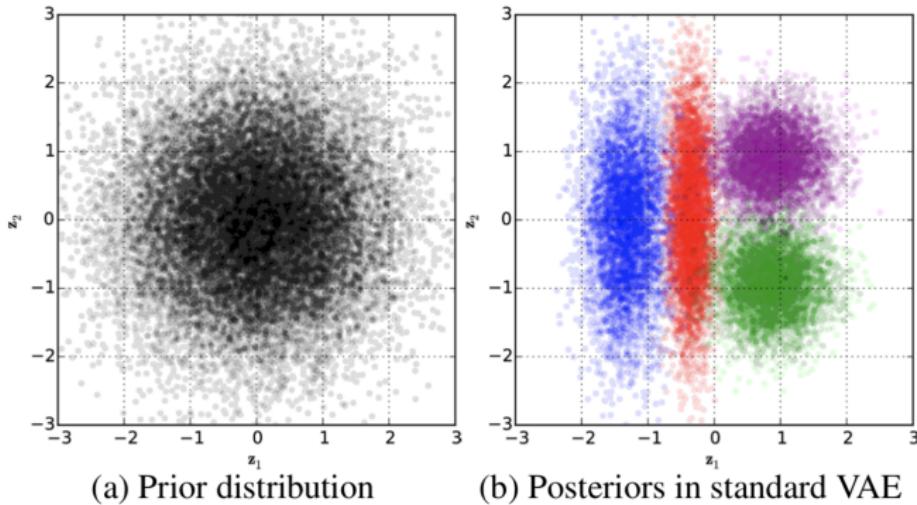
$$\text{KL}(q_{\text{agg}, \phi}(\mathbf{z}) \| p(\mathbf{z}))$$

- ▶  $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\phi(\mathbf{x}), \boldsymbol{\sigma}_\phi^2(\mathbf{x}))$  is unimodal.
- ▶ It is generally believed that the **mismatch between  $p(\mathbf{z})$  and  $q_{\text{agg}, \phi}(\mathbf{z})$**  is the primary explanation for blurry VAE-generated images.

## Marginal KL

$$\text{KL}(q_{\text{agg}, \phi}(\mathbf{z}) \| p(\mathbf{z}))$$

- ▶  $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\phi(\mathbf{x}), \boldsymbol{\sigma}_\phi^2(\mathbf{x}))$  is unimodal.
- ▶ It is generally believed that the **mismatch between  $p(\mathbf{z})$  and  $q_{\text{agg}, \phi}(\mathbf{z})$**  is the primary explanation for blurry VAE-generated images.



# Outline

1. Discrete VAE Latent Representations
2. Vector Quantized VAE
3. ELBO Surgery
4. Learnable VAE Prior

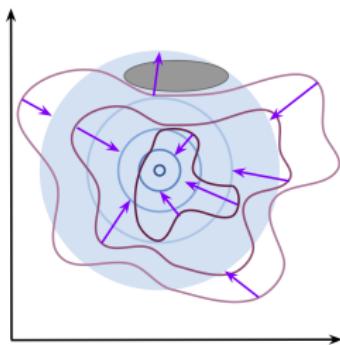
## Optimal VAE Prior

- ▶ Standard Gaussian  $p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$  often leads to over-regularization.
- ▶  $p(\mathbf{z}) = q_{\text{agg}, \phi}(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z} | \mathbf{x}_i)$  risks overfitting and incurs high computational cost.

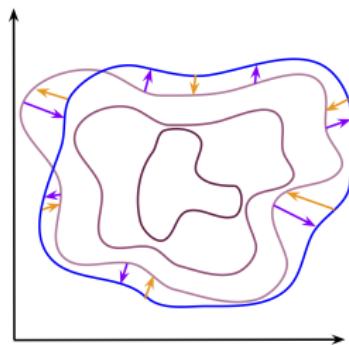
# Optimal VAE Prior

- ▶ Standard Gaussian  $p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$  often leads to over-regularization.
- ▶  $p(\mathbf{z}) = q_{\text{agg}, \phi}(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z} | \mathbf{x}_i)$  risks overfitting and incurs high computational cost.

Non-Learnable Prior  $p(\mathbf{z})$



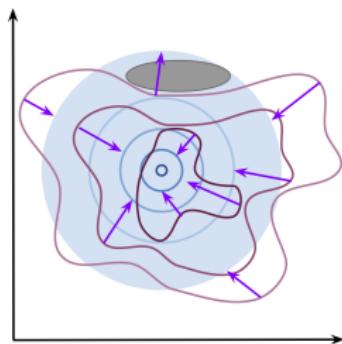
Learnable Prior  $p_\lambda(\mathbf{z})$



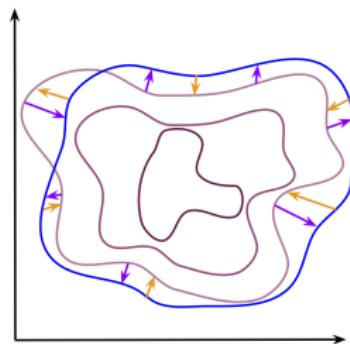
# Optimal VAE Prior

- ▶ Standard Gaussian  $p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$  often leads to over-regularization.
- ▶  $p(\mathbf{z}) = q_{\text{agg}, \phi}(\mathbf{z}) = \frac{1}{n} \sum_{i=1}^n q_\phi(\mathbf{z} | \mathbf{x}_i)$  risks overfitting and incurs high computational cost.

Non-Learnable Prior  $p(\mathbf{z})$



Learnable Prior  $p_\lambda(\mathbf{z})$



$$\frac{1}{n} \sum_{i=1}^n \mathcal{L}_{\phi, \theta}(\mathbf{x}_i) = \text{RL} - \text{MI} - \text{KL}(q_{\text{agg}, \phi}(\mathbf{z}) \| p_\lambda(\mathbf{z}))$$

This is the forward KL divergence with respect to  $p_\lambda(\mathbf{z})$ .

image credit: [https://jmtomczak.github.io/blog/7/7\\_priors.html](https://jmtomczak.github.io/blog/7/7_priors.html)

# NF-Based VAE Prior

## NF Model in Latent Space

$$\log p_{\lambda}(\mathbf{z}) = \log p(\mathbf{z}^*) + \log \left| \det \left( \frac{d\mathbf{z}^*}{d\mathbf{z}} \right) \right| = \log p(\mathbf{f}_{\lambda}(\mathbf{z})) + \log |\det(\mathbf{J}_f)|$$

$$\mathbf{z} = \mathbf{g}_{\lambda}(\mathbf{z}^*) = \mathbf{f}_{\lambda}^{-1}(\mathbf{z}^*)$$

## NF-Based VAE Prior

### NF Model in Latent Space

$$\log p_{\lambda}(\mathbf{z}) = \log p(\mathbf{z}^*) + \log \left| \det \left( \frac{d\mathbf{z}^*}{d\mathbf{z}} \right) \right| = \log p(\mathbf{f}_{\lambda}(\mathbf{z})) + \log |\det(\mathbf{J}_f)|$$

$$\mathbf{z} = \mathbf{g}_{\lambda}(\mathbf{z}^*) = \mathbf{f}_{\lambda}^{-1}(\mathbf{z}^*)$$

- ▶ RealNVP with coupling layers,
- ▶ Autoregressive normalizing flows (efficient  $\mathbf{f}_{\lambda}(\mathbf{z})$ , but  $\mathbf{g}_{\lambda}(\mathbf{z}^*)$  can be slow).

## NF-Based VAE Prior

### NF Model in Latent Space

$$\log p_{\lambda}(\mathbf{z}) = \log p(\mathbf{z}^*) + \log \left| \det \left( \frac{d\mathbf{z}^*}{d\mathbf{z}} \right) \right| = \log p(\mathbf{f}_{\lambda}(\mathbf{z})) + \log |\det(\mathbf{J}_f)|$$

$$\mathbf{z} = \mathbf{g}_{\lambda}(\mathbf{z}^*) = \mathbf{f}_{\lambda}^{-1}(\mathbf{z}^*)$$

- ▶ RealNVP with coupling layers,
- ▶ Autoregressive normalizing flows (efficient  $\mathbf{f}_{\lambda}(\mathbf{z})$ , but  $\mathbf{g}_{\lambda}(\mathbf{z}^*)$  can be slow).

### ELBO with NF-Based VAE Prior

$$\mathcal{L}_{\phi, \theta}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z}))$$

## NF-Based VAE Prior

### NF Model in Latent Space

$$\log p_{\lambda}(\mathbf{z}) = \log p(\mathbf{z}^*) + \log \left| \det \left( \frac{d\mathbf{z}^*}{d\mathbf{z}} \right) \right| = \log p(\mathbf{f}_{\lambda}(\mathbf{z})) + \log |\det(\mathbf{J}_f)|$$

$$\mathbf{z} = \mathbf{g}_{\lambda}(\mathbf{z}^*) = \mathbf{f}_{\lambda}^{-1}(\mathbf{z}^*)$$

- ▶ RealNVP with coupling layers,
- ▶ Autoregressive normalizing flows (efficient  $\mathbf{f}_{\lambda}(\mathbf{z})$ , but  $\mathbf{g}_{\lambda}(\mathbf{z}^*)$  can be slow).

### ELBO with NF-Based VAE Prior

$$\begin{aligned}\mathcal{L}_{\phi, \theta}(\mathbf{x}) &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) = \\ &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} [\log p_{\theta}(\mathbf{x}|\mathbf{z}) + \log p_{\lambda}(\mathbf{z}) - \log q_{\phi}(\mathbf{z}|\mathbf{x})]\end{aligned}$$

# NF-Based VAE Prior

## NF Model in Latent Space

$$\log p_{\lambda}(\mathbf{z}) = \log p(\mathbf{z}^*) + \log \left| \det \left( \frac{d\mathbf{z}^*}{d\mathbf{z}} \right) \right| = \log p(\mathbf{f}_{\lambda}(\mathbf{z})) + \log |\det(\mathbf{J}_f)|$$

$$\mathbf{z} = \mathbf{g}_{\lambda}(\mathbf{z}^*) = \mathbf{f}_{\lambda}^{-1}(\mathbf{z}^*)$$

- ▶ RealNVP with coupling layers,
- ▶ Autoregressive normalizing flows (efficient  $\mathbf{f}_{\lambda}(\mathbf{z})$ , but  $\mathbf{g}_{\lambda}(\mathbf{z}^*)$  can be slow).

## ELBO with NF-Based VAE Prior

$$\begin{aligned}\mathcal{L}_{\phi, \theta}(\mathbf{x}) &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - \text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) = \\ &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} [\log p_{\theta}(\mathbf{x}|\mathbf{z}) + \log p_{\lambda}(\mathbf{z}) - \log q_{\phi}(\mathbf{z}|\mathbf{x})] = \\ &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \left[ \log p_{\theta}(\mathbf{x}|\mathbf{z}) + \underbrace{\left( \log p(\mathbf{f}_{\lambda}(\mathbf{z})) + \log |\det(\mathbf{J}_f)| \right)}_{\text{NF-based prior}} - \log q_{\phi}(\mathbf{z}|\mathbf{x}) \right]\end{aligned}$$

## Summary

- ▶ Discrete VAE latents offer a natural class of latent variable models.
- ▶ Vector quantization provides a way to construct VAEs with discrete latents and deterministic variational posteriors.
- ▶ The straight-through gradient estimator allows gradients to pass as if quantization were an identity operation during backpropagation.
- ▶ ELBO surgery gives insights into the prior's influence in VAEs; the optimal prior is the aggregated variational posterior.
- ▶ The mismatch between  $p(\mathbf{z})$  and  $q_{\text{agg},\phi}(\mathbf{z})$  is widely regarded as the principal reason for VAE-generated image blurriness.
- ▶ Normalizing flow-based priors, including autoregressive flows, can be incorporated directly into VAEs.