Deep Generative Models

Lecture 5

Roman Isachenko

AI Masters

2025, Spring

Recap of previous lecture

EM-algorithm

- E-step

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

- M-step

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

Amortized variational inference

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

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

- M-step

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

Recap of previous lecture

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

M-step: $\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}}(\mathbf{x})$, Monte Carlo estimation

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

E-step: $\nabla_{\boldsymbol{\phi}}\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}}(\mathbf{x})$, reparametrization trick

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

Variational assumption

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

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

Recap of previous lecture

Training (EM-algorithm)

- pick random sample $\mathbf{x}_i$, $i \sim \text{Uniform}\{1, n\}$ (or batch).
- compute the objective (using reparametrization trick):

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

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

- make gradient step using stochastic gradients w.r.t. $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ via autograd

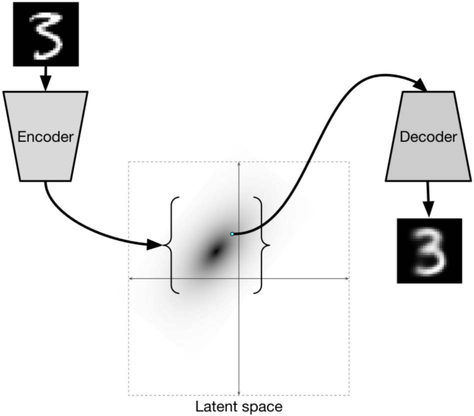
Inference

- sample $\mathbf{z}^*$ from the prior distribution $p(\mathbf{z})$ ($\mathcal{N}(0, \mathbf{I})$);
- sample from the decoder $p(\mathbf{x}|\mathbf{z}^*, \boldsymbol{\theta})$.

Note: you do not need the encoder $q(\mathbf{z}|\mathbf{x}, \boldsymbol{\phi})$ during the generation.

Recap of previous lecture

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



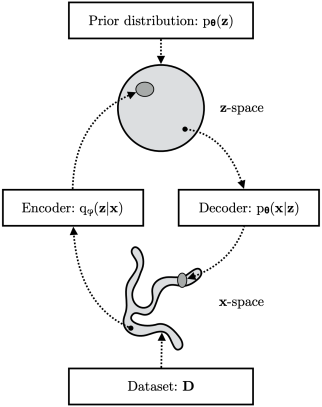


image credit: http://ijdykeman.github.io/ml/2016/12/21/cvae.html
Kingma D. P., Welling M. An introduction to variational autoencoders, 2019

Recap of previous lecture

	VAE	**NF**
Objective	ELBO $\mathcal{L}$	Forward KL/MLE
Encoder	stochastic $\mathbf{z} \sim q(\mathbf{z}\|\mathbf{x}, \phi)$	deterministic $\mathbf{z} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})$ $q(\mathbf{z}\|\mathbf{x}, \boldsymbol{\theta}) = \delta(\mathbf{z} - \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}))$
Decoder	stochastic $\mathbf{x} \sim p(\mathbf{x}\|\mathbf{z}, \boldsymbol{\theta})$	deterministic $\mathbf{x} = \mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z})$ $p(\mathbf{x}\|\mathbf{z}, \boldsymbol{\theta}) = \delta(\mathbf{x} - \mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z}))$
Parameters	$\phi, \boldsymbol{\theta}$	$\boldsymbol{\theta} \equiv \phi$

Theorem

MLE for normalizing flow is equivalent to maximization of ELBO for VAE model with deterministic encoder and decoder:

$$p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \delta(\mathbf{x} - \mathbf{f}_{\boldsymbol{\theta}}^{-1}(\mathbf{z})) = \delta(\mathbf{x} - \mathbf{g}_{\boldsymbol{\theta}}(\mathbf{z}));$$

$$q(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}) = p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}) = \delta(\mathbf{z} - \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})).$$

Nielsen D., et al. SurVAE Flows: Surjections to Bridge the Gap between VAEs and Flows, 2020

Recap of previous lecture

Assumptions

- Let $c \sim \text{Categorical}(\boldsymbol{\pi})$, where
$$\boldsymbol{\pi} = (\pi_1, \dots, \pi_K), \quad \pi_k = P(c = k), \quad \sum_{k=1}^{K} \pi_k = 1.$$
- Let VAE model has discrete latent representation c with prior $p(c) = \text{Uniform}\{1, \dots, K\}$.

ELBO

$$\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}}(\mathbf{x}) = \mathbb{E}_{q(c|\mathbf{x},\boldsymbol{\phi})} \log p(\mathbf{x}|c, \boldsymbol{\theta}) - KL(q(c|\mathbf{x}, \boldsymbol{\phi})||p(c)) \to \max_{\boldsymbol{\phi},\boldsymbol{\theta}}.$$
$$KL(q(c|\mathbf{x}, \boldsymbol{\phi})||p(c)) = -H(q(c|\mathbf{x}, \boldsymbol{\phi})) + \log K.$$

- Our encoder should output discrete distribution $q(c|\mathbf{x}, \boldsymbol{\phi})$.
- We need the analogue of the reparametrization trick for the discrete distribution $q(c|\mathbf{x}, \boldsymbol{\phi})$.
- Our decoder $p(\mathbf{x}|c, \boldsymbol{\theta})$ should input discrete random variable c.

Oord A., Vinyals O., Kavukcuoglu K. Neural Discrete Representation Learning, 2017

Outline

1. Vector quantization: (discrete VAE latent representations)
2. ELBO surgery
3. Learnable VAE prior
4. Likelihood-free learning

Outline

1. Vector quantization: (discrete VAE latent representations)
2. ELBO surgery
3. Learnable VAE prior
4. Likelihood-free learning

Vector quantization

Define the dictionary space $\{\mathbf{e}_k\}_{k=1}^{K}$, where $\mathbf{e}_k \in \mathbb{R}^L$, K is the size of the dictionary.

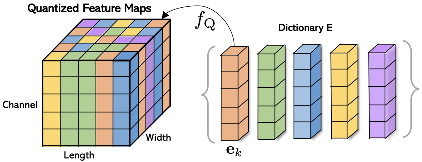
Quantized representation

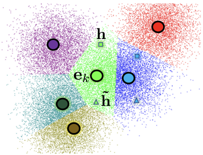
$\mathbf{z}_q \in \mathbb{R}^L$ for $\mathbf{z} \in \mathbb{R}^L$ is defined by a nearest neighbor look-up using the dictionary space

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

Quantization procedure

If we have tensor with the spatial dimensions we apply the quantization for each of $W \times H$ locations.





Zhao Y. et al. Feature Quantization Improves GAN Training, 2020

Vector Quantized VAE (VQ-VAE)

- Let our encoder outputs continuous representation $\mathbf{z}_e = \mathrm{NN}_{e,\phi}(\mathbf{x}) \in \mathbb{R}^L$.
- Quantization will give us the deterministic mapping from the encoder output $\mathbf{z}_e$ to its quantized representation $\mathbf{z}_q$.
- Let use the dictionary elements $\mathbf{e}_c$ in the decoder distribution $p(\mathbf{x}|\mathbf{e}_c, \boldsymbol{\theta})$ (instead of $p(\mathbf{x}|c, \boldsymbol{\theta})$).

Deterministic variational posterior

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

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

Note: KL term (regularizer) does not affect the ELBO objective.

Oord A., Vinyals O., Kavukcuoglu K. Neural Discrete Representation Learning, 2017

Vector Quantized VAE (VQ-VAE): forward

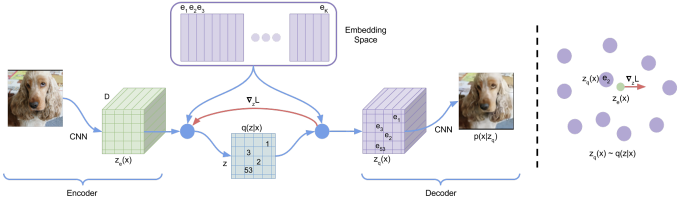
Deterministic variational posterior

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

ELBO

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

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



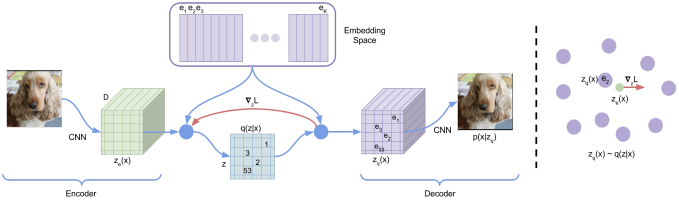
Problem: $\arg\min$ is not differentiable.

Oord A., Vinyals O., Kavukcuoglu K. Neural Discrete Representation Learning, 2017

Vector Quantized VAE (VQ-VAE): backward

ELBO

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



Straight-through gradient estimation

$$\frac{\partial \log p(\mathbf{x}|\mathbf{z}_q, \boldsymbol{\theta})}{\partial \phi} = \frac{\partial \log p(\mathbf{x}|\mathbf{z}_q, \boldsymbol{\theta})}{\partial \mathbf{z}_q} \cdot \frac{\partial \mathbf{z}_q}{\partial \phi} =$$

$$= \frac{\partial \log p(\mathbf{x}|\mathbf{z}_q, \boldsymbol{\theta})}{\partial \mathbf{z}_q} \cdot \frac{\partial \mathbf{z}_q}{\partial \mathbf{z}_e} \cdot \frac{\partial \mathbf{z}_e}{\partial \phi} \approx \frac{\partial \log p(\mathbf{x}|\mathbf{z}_q, \boldsymbol{\theta})}{\partial \mathbf{z}_q} \cdot \frac{\partial \mathbf{z}_e}{\partial \phi}$$

Oord A., Vinyals O., Kavukcuoglu K. Neural Discrete Representation Learning, 2017

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

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

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

Samples diversity

VQ-VAE (Proposed) **BigGAN deep**

Razavi A., Oord A., Vinyals O. Generating Diverse High-Fidelity Images with VQ-VAE-2, 2019

Outline

1. Vector quantization: (discrete VAE latent representations)
2. ELBO surgery
3. Learnable VAE prior
4. Likelihood-free learning

ELBO surgery

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

Theorem

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

- $q_{\text{agg}}(\mathbf{z}|\phi) = \frac{1}{n}\sum_{i=1}^{n} q(\mathbf{z}|\mathbf{x}_i,\phi)$ is the **aggregated** variational posterior distribution.
- $\mathbb{I}_q[\mathbf{x},\mathbf{z}]$ is the mutual information between $\mathbf{x}$ and $\mathbf{z}$ under the data distribution $\pi(\mathbf{x})$ and the distribution $q(\mathbf{z}|\mathbf{x},\phi)$.
- First term pushes $q_{\text{agg}}(\mathbf{z}|\phi)$ towards the prior $p(\mathbf{z})$.
- Second term reduces the amount of information about $\mathbf{x}$ stored in $\mathbf{z}$.

Hoffman M. D., Johnson M. J. ELBO surgery: yet another way to carve up the variational evidence lower bound, 2016

ELBO surgery

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

Proof

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

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

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

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

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

Hoffman M. D., Johnson M. J. ELBO surgery: yet another way to carve up the variational evidence lower bound, 2016

ELBO surgery

ELBO revisiting

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

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{q(\mathbf{z}|\mathbf{x}_i,\phi)}\log p(\mathbf{x}_i|\mathbf{z},\boldsymbol{\theta})}_{\text{Reconstruction loss}} - \underbrace{\mathbb{I}_q[\mathbf{x},\mathbf{z}]}_{\text{MI}} - \underbrace{KL(q_{\text{agg}}(\mathbf{z}|\phi)||p(\mathbf{z}))}_{\text{Marginal KL}}$$

Prior distribution $p(\mathbf{z})$ is only in the last term.

Optimal VAE prior

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

The optimal prior $p(\mathbf{z})$ is the aggregated variational posterior distribution $q_{\text{agg}}(\mathbf{z}|\phi)$!

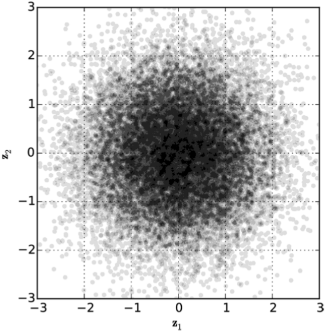
Hoffman M. D., Johnson M. J. ELBO surgery: yet another way to carve up the variational evidence lower bound, 2016

Variational posterior

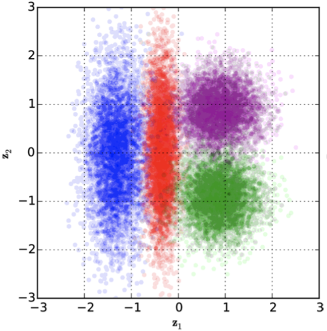
ELBO decomposition

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

- $q(\mathbf{z}|\mathbf{x},\phi) = \mathcal{N}(\boldsymbol{\mu}_\phi(\mathbf{x}), \boldsymbol{\sigma}^2_\phi(\mathbf{x}))$ is a unimodal distribution.
- It is widely believed that **mismatch between $p(\mathbf{z})$ and $q_{\text{agg}}(\mathbf{z}|\phi)$ is the main reason of blurry images of VAE**.



(a) Prior distribution



(b) Posteriors in standard VAE

Rezende D. J., Mohamed S. Variational Inference with Normalizing Flows, 2015

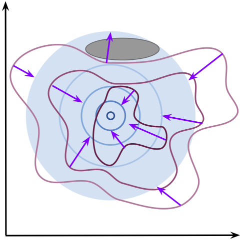
Outline

1. Vector quantization: (discrete VAE latent representations)
2. ELBO surgery
3. Learnable VAE prior
4. Likelihood-free learning

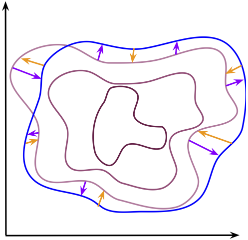
Optimal VAE prior

- Standard Gaussian $p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I}) \Rightarrow$ over-regularization;
- $p(\mathbf{z}) = q_{\text{agg}}(\mathbf{z}|\phi) = \frac{1}{n}\sum_{i=1}^{n} q(\mathbf{z}|\mathbf{x}_i, \phi) \Rightarrow$ overfitting and highly expensive.

Non learnable prior $p(\mathbf{z})$



Learnable prior $p(\mathbf{z}|\boldsymbol{\lambda})$



ELBO revisiting

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

It is Forward KL with respect to $p(\mathbf{z}|\boldsymbol{\lambda})$.

image credit: https://jmtomczak.github.io/blog/7/7_priors.html

NF-based VAE prior

NF model in latent space

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

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

- RealNVP with coupling layers.
- Autoregressive NF (fast $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{z})$, slow $\mathbf{g}_{\boldsymbol{\lambda}}(\mathbf{z}^*)$).

ELBO with NF-based VAE prior

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

Chen X. et al. Variational Lossy Autoencoder, 2016

Outline

1. Vector quantization: (discrete VAE latent representations)

2. ELBO surgery

3. Learnable VAE prior

4. Likelihood-free learning

Likelihood based models

Poor likelihood
Great samples

$$p_1(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{N}(\mathbf{x}|\mathbf{x}_i, \epsilon \mathbf{I})$$

For small ϵ this model will generate samples with great quality, but likelihood of test sample will be very poor.

Great likelihood
Poor samples

$$p_2(\mathbf{x}) = 0.01 p(\mathbf{x}) + 0.99 p_{\text{noise}}(\mathbf{x})$$

$$\log \left[0.01 p(\mathbf{x}) + 0.99 p_{\text{noise}}(\mathbf{x})\right] \geq$$
$$\geq \log \left[0.01 p(\mathbf{x})\right] = \log p(\mathbf{x}) - \log 100$$

Noisy irrelevant samples, but for high dimensions $\log p(\mathbf{x})$ becomes proportional to m.

- Likelihood is not a perfect quality measure for generative model.
- Likelihood could be intractable.

Theis L., Oord A., Bethge M. A note on the evaluation of generative models, 2015

Likelihood-free learning

Where did we start

We would like to approximate true data distribution $\pi(\mathbf{x})$. Instead of searching true $\pi(\mathbf{x})$ over all probability distributions, learn function approximation $p(\mathbf{x}|\boldsymbol{\theta}) \approx \pi(\mathbf{x})$.

Imagine we have two sets of samples

- $\{\mathbf{x}_i\}_{i=1}^{n_1} \sim \pi(\mathbf{x})$ – real samples;
- $\{\mathbf{x}_i\}_{i=1}^{n_2} \sim p(\mathbf{x}|\boldsymbol{\theta})$ – generated (or fake) samples.

Let define discriminative model (classifier):

$$p(y = 1|\mathbf{x}) = P\big(\{\mathbf{x} \sim \pi(\mathbf{x})\}\big); \quad p(y = 0|\mathbf{x}) = P\big(\{\mathbf{x} \sim p(\mathbf{x}|\boldsymbol{\theta})\}\big)$$

Assumption

Generative distribution $p(\mathbf{x}|\boldsymbol{\theta})$ equals to the true distribution $\pi(\mathbf{x})$ if we can not distinguish them using discriminative model $p(y|\mathbf{x})$. It means that $p(y = 1|\mathbf{x}) = 0.5$ for each sample $\mathbf{x}$.

Generative adversarial networks (GAN)

- The more powerful discriminative model we will have, the more likely we will get the "best" generative distribution $p(\mathbf{x}|\boldsymbol{\theta})$.
- The most common way to learn a classifier is to minimize cross entropy loss.

Cross entropy for discriminative model

$$\min_{p(y|\mathbf{x})} \left[-\mathbb{E}_{\pi(\mathbf{x})} \log p(y=1|\mathbf{x}) - \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} \log p(y=0|\mathbf{x})\right]$$

$$\max_{p(y|\mathbf{x})} \left[\mathbb{E}_{\pi(\mathbf{x})} \log p(y=1|\mathbf{x}) + \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} \log p(y=0|\mathbf{x})\right]$$

Generative model

Assume generative model $p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z})$ with the base distribution $p(\mathbf{z})$ and deterministic map $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \delta(\mathbf{x} - \mathbf{G}_{\boldsymbol{\theta}}(\mathbf{z}))$.

Goodfellow I. J. et al. Generative Adversarial Networks, 2014

Generative adversarial networks (GAN)

Cross entropy for discriminative model

$$\max_{p(y|\mathbf{x})} \left[\mathbb{E}_{\pi(\mathbf{x})} \log p(y = 1|\mathbf{x}) + \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} \log p(y = 0|\mathbf{x}) \right]$$

- **Discriminator:** a classifier $p(y = 1|\mathbf{x}, \phi) = D_\phi(\mathbf{x}) \in [0, 1]$, which distinguishes real samples from $\pi(\mathbf{x})$ and generated samples from $p(\mathbf{x}|\boldsymbol{\theta})$. Discriminator tries to **minimize** cross entropy.
- **Generator:** generative model $\mathbf{x} = \mathbf{G}_\theta(\mathbf{z})$ with $\mathbf{z} \sim p(\mathbf{z})$, which makes the generated sample more realistic. Generator tries to **maximize** cross entropy.

GAN Objective

$$\min_G \max_D \left[\mathbb{E}_{\pi(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} \log(1 - D(\mathbf{x})) \right]$$

$$\min_G \max_D \left[\mathbb{E}_{\pi(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{p(\mathbf{z})} \log(1 - D(\mathbf{G}(\mathbf{z}))) \right]$$

Goodfellow I. J. et al. Generative Adversarial Networks, 2014

Summary

- Vector Quantization is the way to create VAE with discrete latent space and deterministic variational posterior.
- Straight-through gradient ignores quantize operation in backprop.
- The ELBO surgery reveals insights about a prior distribution in VAE. The optimal prior is the aggregated variational posterior distribution.
- It is widely believed that mismatch between $p(\mathbf{z})$ and $q_{\text{agg}}(\mathbf{z}|\phi)$ is the main reason of blurry images of VAE.
- We could use NF-based prior in VAE (even autoregressive).
- Likelihood is not a perfect criteria to measure quality of generative model.
- Adversarial learning suggests to solve minimax problem to match the distributions.