

# Deep Generative Models

## Lecture 7

Roman Isachenko

Moscow Institute of Physics and Technology

2024, Autumn

# 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

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

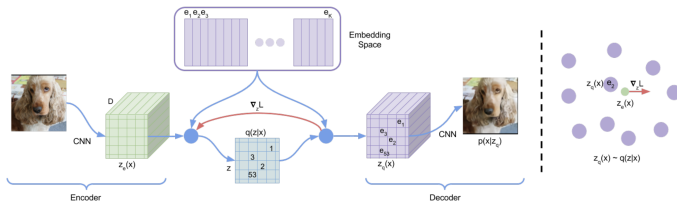
$$\text{KL}(q(c|\mathbf{x}, \phi) || p(c)) = -H(q(c|\mathbf{x}, \phi)) + \log K.$$

## Vector quantization

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

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

# Recap of previous lecture



## Deterministic variational posterior

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

## ELBO

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

## Straight-through gradient estimation

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

# Recap of previous lecture

## Gumbel-max trick

Let  $g_k \sim \text{Gumbel}(0, 1)$  for  $k = 1, \dots, K$ . Then

$$c = \arg \max_k [\log \pi_k + g_k]$$

has a categorical distribution  $c \sim \text{Categorical}(\pi)$ .

## Gumbel-softmax relaxation

Concrete distribution = continuous + discrete

$$\hat{\mathbf{c}} = \text{Softmax} \left( \frac{\log q(\mathbf{c}|\mathbf{x}, \phi) + \mathbf{g}}{\tau} \right)$$

## Reparametrization trick

$$\nabla_{\phi} \mathbb{E}_{q(\mathbf{c}|\mathbf{x}, \phi)} \log p(\mathbf{x}|\mathbf{e}_c, \theta) = \mathbb{E}_{\text{Gumbel}(0,1)} \nabla_{\phi} \log p(\mathbf{x}|\mathbf{z}, \theta),$$

where  $\mathbf{z} = \sum_{k=1}^K \hat{c}_k \mathbf{e}_k$  (all operations are differentiable now).

---

Maddison C. J., Mnih A., Teh Y. W. *The Concrete distribution: A continuous relaxation of discrete random variables*, 2016

Jang E., Gu S., Poole B. *Categorical reparameterization with Gumbel-Softmax*, 2016

# Recap of previous lecture

## Theorem

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

## ELBO surgery

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

## Optimal prior

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

The optimal prior distribution  $p(\mathbf{z})$  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

# Outline

1. Learnable VAE prior
2. Likelihood-free learning
3. Generative adversarial networks (GAN)

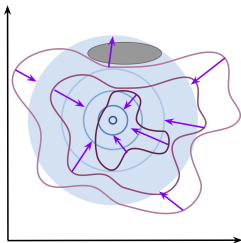
# Outline

1. Learnable VAE prior
2. Likelihood-free learning
3. Generative adversarial networks (GAN)

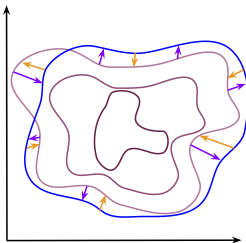
## Optimal VAE prior

- ▶ 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  $p(\mathbf{z}|\lambda)$



ELBO revisiting

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

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



# NF-based VAE prior

## NF model in latent space

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

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

- ▶ 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

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

# Outline

1. Learnable VAE prior
2. Likelihood-free learning
3. Generative adversarial networks (GAN)

# 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  $\epsilon$  this model will generate samples with great quality, but likelihood of test sample will be very poor.

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

Great likelihood  
Poor samples

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

$$\begin{aligned} \log [0.01p(\mathbf{x}) + 0.99p_{\text{noise}}(\mathbf{x})] &\geq \\ &\geq \log [0.01p(\mathbf{x})] = \log p(\mathbf{x}) - \log 100 \end{aligned}$$

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

# 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  $p(\mathbf{x}|\theta) \approx \pi(\mathbf{x})$ .

Imagine we have two sets of samples

- ▶  $\mathcal{S}_1 = \{\mathbf{x}_i\}_{i=1}^{n_1} \sim \pi(\mathbf{x})$  – real samples;
- ▶  $\mathcal{S}_2 = \{\mathbf{x}_i\}_{i=1}^{n_2} \sim p(\mathbf{x}|\theta)$  – generated (or fake) samples.

Let define discriminative model (classifier):

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

## Assumption

Generative distribution  $p(\mathbf{x}|\theta)$  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  $p(\mathbf{x}|\theta)$ .

The most common way to learn a classifier is to minimize cross entropy loss.

- ▶ **Generator:** generative model  $\mathbf{x} = \mathbf{G}(\mathbf{z})$ , which makes generated sample more realistic. Here  $\mathbf{z}$  comes from the base (known) distribution  $p(\mathbf{z})$  and  $\mathbf{x} \sim p(\mathbf{x}|\theta)$ . Generator tries to **maximize** cross entropy.
- ▶ **Discriminator:** a classifier  $p(y = 1|\mathbf{x}) = D(\mathbf{x}) \in [0, 1]$ , which distinguishes real samples from generated samples. Discriminator tries to **minimize** cross entropy (tries to enhance discriminative model).

## Objective

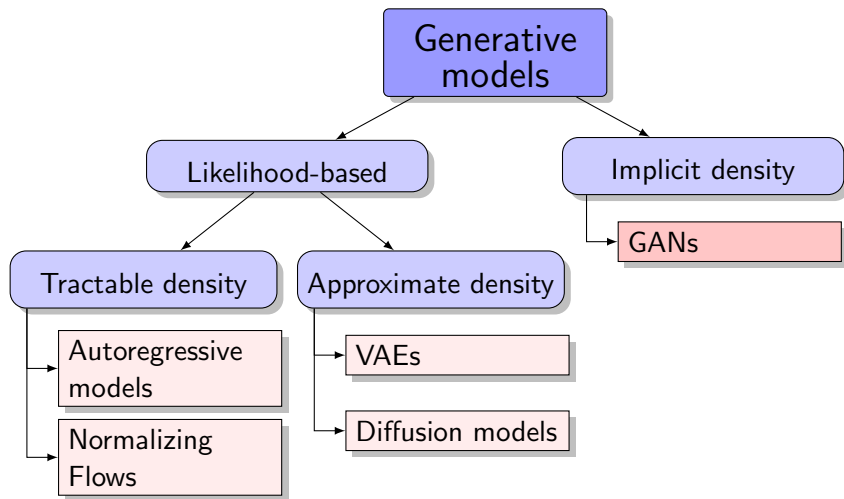
$$\min_G \max_D [\mathbb{E}_{\pi(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{p(\mathbf{x}|\theta)} \log(1 - D(\mathbf{x}))]$$

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

# Outline

1. Learnable VAE prior
2. Likelihood-free learning
3. Generative adversarial networks (GAN)

# Generative models zoo



# GAN optimality

## Theorem

The minimax game

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

has the global optimum  $\pi(\mathbf{x}) = p(\mathbf{x}|\boldsymbol{\theta})$ , in this case  $D^*(\mathbf{x}) = 0.5$ .

## Proof (fixed $G$ )

$$\begin{aligned} V(G, D) &= \mathbb{E}_{\pi(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{p(\mathbf{x}|\boldsymbol{\theta})} \log(1 - D(\mathbf{x})) \\ &= \int \underbrace{[\pi(\mathbf{x}) \log D(\mathbf{x}) + p(\mathbf{x}|\boldsymbol{\theta}) \log(1 - D(\mathbf{x}))]}_{y(D)} d\mathbf{x} \end{aligned}$$

$$\frac{dy(D)}{dD} = \frac{\pi(\mathbf{x})}{D(\mathbf{x})} - \frac{p(\mathbf{x}|\boldsymbol{\theta})}{1 - D(\mathbf{x})} = 0 \quad \Rightarrow \quad D^*(\mathbf{x}) = \frac{\pi(\mathbf{x})}{\pi(\mathbf{x}) + p(\mathbf{x}|\boldsymbol{\theta})}$$



# GAN optimality

Proof continued (fixed  $D = D^*$ )

$$\begin{aligned} V(G, D^*) &= \mathbb{E}_{\pi(\mathbf{x})} \log \left( \frac{\pi(\mathbf{x})}{\pi(\mathbf{x}) + p(\mathbf{x}|\theta)} \right) + \mathbb{E}_{p(\mathbf{x}|\theta)} \log \left( \frac{p(\mathbf{x}|\theta)}{\pi(\mathbf{x}) + p(\mathbf{x}|\theta)} \right) \\ &= KL \left( \pi(\mathbf{x}) \parallel \frac{\pi(\mathbf{x}) + p(\mathbf{x}|\theta)}{2} \right) + KL \left( p(\mathbf{x}|\theta) \parallel \frac{\pi(\mathbf{x}) + p(\mathbf{x}|\theta)}{2} \right) - 2 \log 2 \\ &= 2JSD(\pi(\mathbf{x}) \parallel p(\mathbf{x}|\theta)) - 2 \log 2. \end{aligned}$$

Jensen-Shannon divergence (symmetric KL divergence)

$$JSD(\pi(\mathbf{x}) \parallel p(\mathbf{x}|\theta)) = \frac{1}{2} \left[ KL \left( \pi(\mathbf{x}) \parallel \frac{\pi(\mathbf{x}) + p(\mathbf{x}|\theta)}{2} \right) + KL \left( p(\mathbf{x}|\theta) \parallel \frac{\pi(\mathbf{x}) + p(\mathbf{x}|\theta)}{2} \right) \right]$$

Could be used as a distance measure!

$$V(G^*, D^*) = -2 \log 2, \quad \pi(\mathbf{x}) = p(\mathbf{x}|\theta), \quad D^*(\mathbf{x}) = 0.5.$$

# GAN optimality

## Theorem

The minimax game

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

has the global optimum  $\pi(\mathbf{x}) = p(\mathbf{x}|\theta)$ , in this case  $D^*(\mathbf{x}) = 0.5$ .

## Expectations

If the generator could be **any** function and the discriminator is **optimal** at every step, then the generator is **guaranteed to converge** to the data distribution.

## Reality

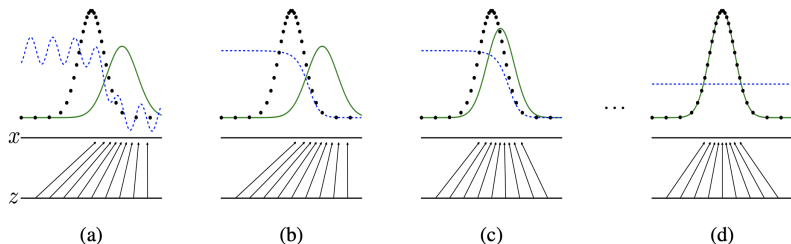
- ▶ Generator updates are made in parameter space, discriminator is not optimal at every step.
- ▶ Generator and discriminator loss keeps oscillating during GAN training.

# GAN training

Let further assume that generator and discriminator are parametric models:  $D_\phi(\mathbf{x})$  and  $\mathbf{G}_\theta(\mathbf{z})$ .

## Objective

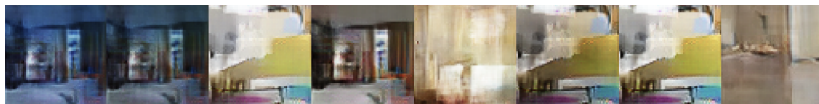
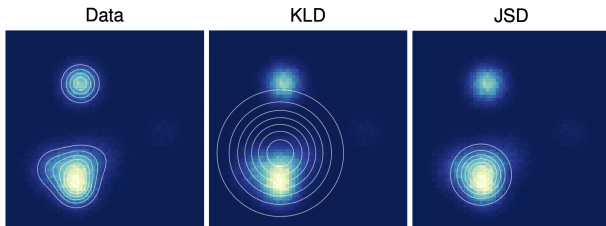
$$\min_{\theta} \max_{\phi} [\mathbb{E}_{\pi(\mathbf{x})} \log D_\phi(\mathbf{x}) + \mathbb{E}_{p(\mathbf{z})} \log(1 - D_\phi(\mathbf{G}_\theta(\mathbf{z})))]$$



- ▶  $\mathbf{z} \sim p(\mathbf{z})$  is a latent variable.
- ▶  $p(\mathbf{x}|\mathbf{z}, \theta) = \delta(\mathbf{x} - \mathbf{G}_\theta(\mathbf{z}))$  is deterministic decoder (like NF).
- ▶ We do not have encoder at all.

# Mode collapse

The phenomena where the generator of a GAN collapses to one or few distribution modes.



Alternate architectures, adding regularization terms, injecting small noise perturbations and other millions bags and tricks are used to avoid the mode collapse.

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

*Metz L. et al. Unrolled Generative Adversarial Networks, 2016*

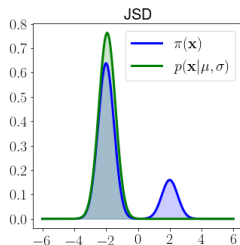
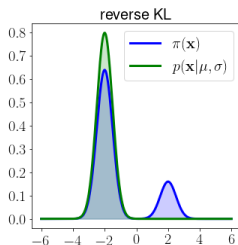
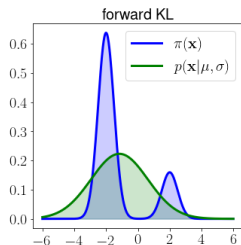
# Jensen-Shannon vs Kullback-Leibler

- ▶  $\pi(\mathbf{x})$  is a fixed mixture of 2 gaussians.
- ▶  $p(\mathbf{x}|\mu, \sigma) = \mathcal{N}(\mu, \sigma^2)$ .

## Mode covering vs mode seeking

$$KL(\pi||p) = \int \pi(\mathbf{x}) \log \frac{\pi(\mathbf{x})}{p(\mathbf{x})} d\mathbf{x}, \quad KL(p||\pi) = \int p(\mathbf{x}) \log \frac{p(\mathbf{x})}{\pi(\mathbf{x})} d\mathbf{x}$$

$$JSD(\pi||p) = \frac{1}{2} \left[ KL \left( \pi(\mathbf{x}) || \frac{\pi(\mathbf{x}) + p(\mathbf{x})}{2} \right) + KL \left( p(\mathbf{x}) || \frac{\pi(\mathbf{x}) + p(\mathbf{x})}{2} \right) \right]$$



# Summary

- ▶ 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.
- ▶ GAN tries to optimize Jensen-Shannon divergence (in theory).
- ▶ Mode collapse is one of the main problems of vanilla GAN. Lots of tips and tricks has to be used to make the GAN training is stable and scalable.