

# Deep Generative Models

## Lecture 7

Roman Isachenko



AI Masters

2024, Summer

## Recap of previous lecture

- ▶ 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.

## ELBO revisiting

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

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

## ELBO with flow-based VAE prior

$$\begin{aligned} \mathcal{L}(\phi, \theta) &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} [\log p(\mathbf{x}|\mathbf{z}, \theta) + \log p(\mathbf{z}|\lambda) - \log q(\mathbf{z}|\mathbf{x}, \phi)] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \left[ \log p(\mathbf{x}|\mathbf{z}, \theta) + \underbrace{\left( \log p(\mathbf{f}_{\lambda}(\mathbf{z})) + \log |\det(\mathbf{J}_{\mathbf{f}})| \right)}_{\text{flow-based prior}} - \log q(\mathbf{z}|\mathbf{x}, \phi) \right] \\ \mathbf{z} &= \mathbf{f}_{\lambda}^{-1}(\mathbf{z}^*) = \mathbf{g}_{\lambda}(\mathbf{z}^*), \quad \mathbf{z}^* \sim p(\mathbf{z}^*) = \mathcal{N}(0, \mathbf{I}) \end{aligned}$$

# 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

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

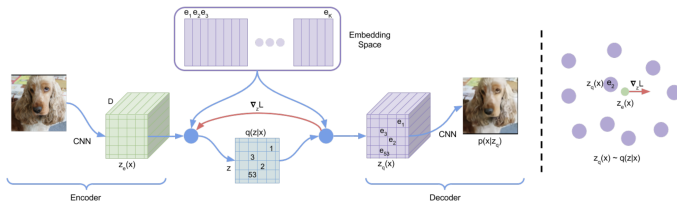
$$\text{KL}(q(c|\mathbf{x}, \boldsymbol{\phi})||p(c)) = -H(q(c|\mathbf{x}, \boldsymbol{\phi})) + \log K.$$

## 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

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

## Straight-through gradient estimation

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

# Outline

1. Likelihood-free learning
2. Generative adversarial networks (GAN)
3. Wasserstein distance

# Outline

1. Likelihood-free learning
2. Generative adversarial networks (GAN)
3. Wasserstein distance

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

Imagine we have two sets of samples

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

Let define discriminative model (classifier):

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

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

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

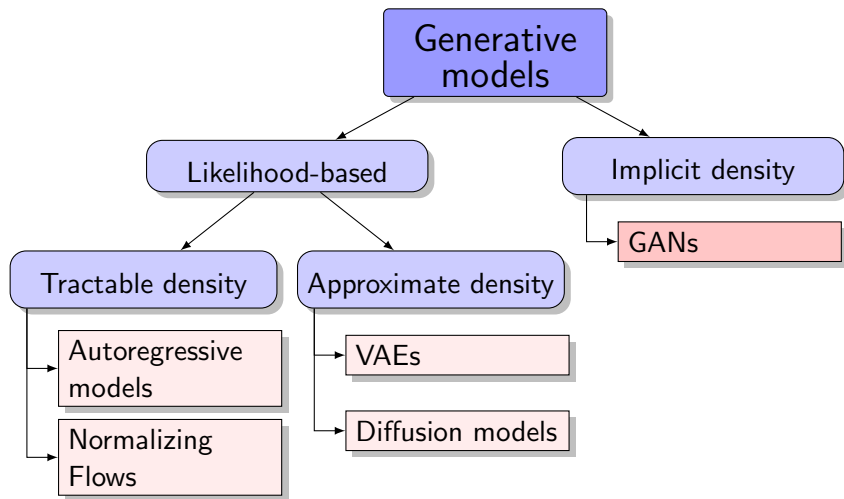
$$\min_G \max_D [\mathbb{E}_{\pi(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{p(\mathbf{x}|\theta)} \log(1 - D(\mathbf{x}))]$$

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

# Outline

1. Likelihood-free learning
2. Generative adversarial networks (GAN)
3. Wasserstein distance

# 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

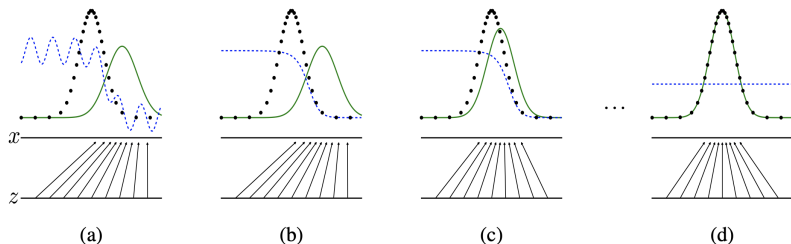
- ▶ Generator updates are made in parameter space, discriminator is not optimal at every step.
- ▶ Generator and discriminator loss keeps oscillating during GAN training.

# GAN training

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

## Objective

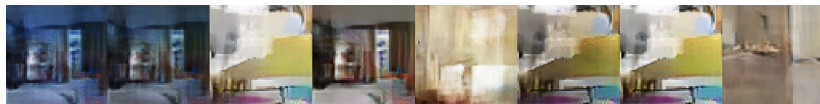
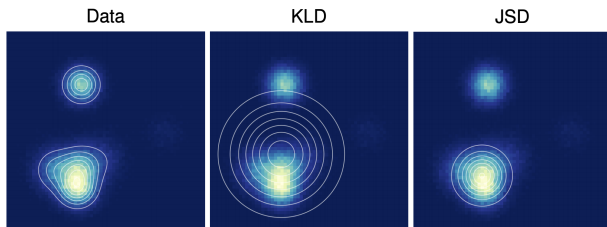
$$\min_{\theta} \max_{\phi} [\mathbb{E}_{\pi(\mathbf{x})} \log D_\phi(\mathbf{x}) + \mathbb{E}_{p(\mathbf{z})} \log(1 - D_\phi(\mathbf{G}_\theta(\mathbf{z})))]$$



- ▶  $\mathbf{z} \sim p(\mathbf{z})$  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*



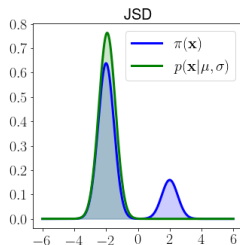
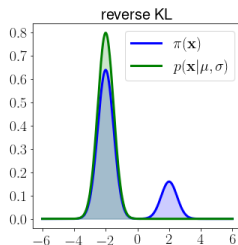
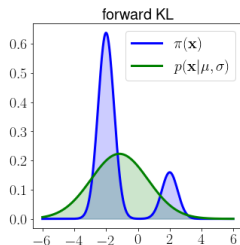
# Jensen-Shannon vs Kullback-Leibler

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

## Mode covering vs mode seeking

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

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

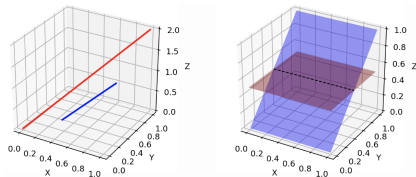


# Outline

1. Likelihood-free learning
2. Generative adversarial networks (GAN)
3. Wasserstein distance

# Informal theoretical results

- ▶ The dimensionality of  $\mathbf{z}$  is lower than the dimensionality of  $\mathbf{x}$ . Hence, support of  $p(\mathbf{x}|\boldsymbol{\theta})$  with  $\mathbf{x} = \mathbf{G}_{\boldsymbol{\theta}}(\mathbf{z})$  lies on low-dimensional manifold.
- ▶ Distribution of real images  $\pi(\mathbf{x})$  is also concentrated on a low dimensional manifold.



- ▶ If  $\pi(\mathbf{x})$  and  $p(\mathbf{x}|\boldsymbol{\theta})$  have disjoint supports, then there is a smooth optimal discriminator.
- ▶ For such low-dimensional disjoint manifolds

$$KL(\pi||p) = KL(p||\pi) = \infty, \quad JSD(\pi||p) = \log 2$$

---

Weng L. From GAN to WGAN, 2019

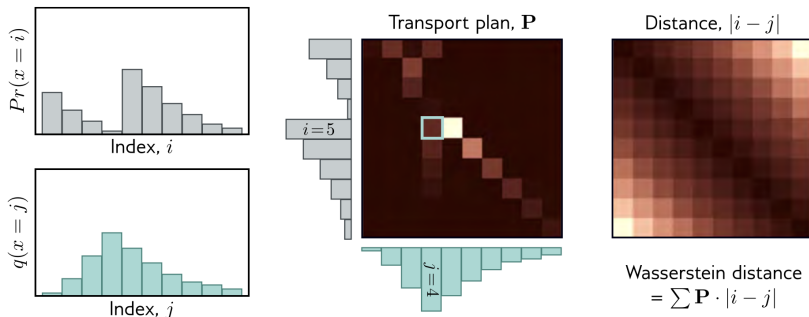
Arjovsky M., Bottou L. Towards Principled Methods for Training Generative Adversarial Networks, 2017

# Wasserstein distance (discrete)

A.k.a. **Earth Mover's distance**.

## Optimal transport formulation

The minimum cost of moving and transforming a pile of dirt in the shape of one probability distribution to the shape of the other distribution.



## Wasserstein distance (continuous)

$$W(\pi, p) = \inf_{\gamma \in \Gamma(\pi, p)} \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \gamma} \|\mathbf{x} - \mathbf{y}\| = \inf_{\gamma \in \Gamma(\pi, p)} \int \|\mathbf{x} - \mathbf{y}\| \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}$$

- ▶  $\gamma(\mathbf{x}, \mathbf{y})$  – transportation plan (the amount of "dirt" that should be transported from point  $\mathbf{x}$  to point  $\mathbf{y}$ )

$$\int \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} = p(\mathbf{y}); \quad \int \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{y} = \pi(\mathbf{x}).$$

- ▶  $\Gamma(\pi, p)$  – the set of all joint distributions  $\gamma(\mathbf{x}, \mathbf{y})$  with marginals  $\pi$  and  $p$ .
- ▶  $\gamma(\mathbf{x}, \mathbf{y})$  – the amount,  $\|\mathbf{x} - \mathbf{y}\|$  – the distance.

## Wasserstein metric

$$W_s(\pi, p) = \inf_{\gamma \in \Gamma(\pi, p)} \left( \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \gamma} \|\mathbf{x} - \mathbf{y}\|^s \right)^{1/s}$$

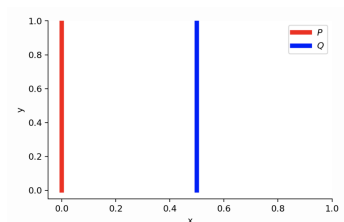
Here we will use  $W(\pi, p) = W_1(\pi, p)$  that corresponds to the optimal transport formulation.

# Wasserstein distance vs KL vs JSD

Consider 2d distributions

$$\pi(x, y) = (0, U[0, 1])$$

$$p(x, y|\theta) = (\theta, U[0, 1])$$



- $\theta = 0$ . Distributions are the same

$$KL(\pi||p) = KL(p||\pi) = JSD(p||\pi) = W(\pi, p) = 0$$

- $\theta \neq 0$

$$KL(\pi||p) = \int_{U[0,1]} 1 \log \frac{1}{0} dy = \infty = KL(p||\pi)$$

$$JSD(\pi||p) = \frac{1}{2} \left( \int_{U[0,1]} 1 \log \frac{1}{1/2} dy + \int_{U[0,1]} 1 \log \frac{1}{1/2} dy \right) = \log 2$$

$$W(\pi, p) = |\theta|$$

---

Weng L. From GAN to WGAN, 2019

Arjovsky M., Chintala S., Bottou L. Wasserstein GAN, 2017

# Wasserstein distance vs KL vs JSD

## Theorem 1

Let  $\mathbf{G}_\theta(\mathbf{z})$  be (almost) any feedforward neural network, and  $p(\mathbf{z})$  a prior over  $\mathbf{z}$  such that  $\mathbb{E}_{p(\mathbf{z})} \|\mathbf{z}\| < \infty$ . Then therefore  $W(\pi, p)$  is continuous everywhere and differentiable almost everywhere.

## Theorem 2

Let  $\pi$  be a distribution on a compact space  $\mathcal{X}$  and  $\{p_t\}_{t=1}^\infty$  be a sequence of distributions on  $\mathcal{X}$ .

$$KL(\pi \| p_t) \rightarrow 0 \text{ (or } KL(p_t \| \pi) \rightarrow 0) \quad (1)$$

$$JSD(\pi \| p_t) \rightarrow 0 \quad (2)$$

$$W(\pi \| p_t) \rightarrow 0 \quad (3)$$

Then, considering limits as  $t \rightarrow \infty$ , (1) implies (2), (2) implies (3).

# Summary

- ▶ 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.
- ▶ KL and JS divergences work poorly as model objective in the case of disjoint supports.
- ▶ Earth-Mover distance is a more appropriate objective function for distribution matching problem.