

# Deep Generative Models

## Lecture 9

Roman Isachenko

Moscow Institute of Physics and Technology

2022

# Recap of previous lecture

## ELBO with flow-based VAE posterior

$$\log q(\mathbf{z}^*|\mathbf{x}, \phi, \lambda) = \log q(\mathbf{z}|\mathbf{x}, \phi) + \log \left| \det \left( \frac{d\mathbf{z}}{d\mathbf{z}^*} \right) \right|$$

$$\mathcal{L}(\phi, \theta, \lambda) = \mathbb{E}_{q(\mathbf{z}^*|\mathbf{x}, \phi, \lambda)} \log p(\mathbf{x}|\mathbf{z}^*, \theta) - \textcolor{violet}{KL}(q(\mathbf{z}^*|\mathbf{x}, \phi, \lambda) || p(\mathbf{z}^*)).$$

The **second term** in ELBO is **reverse** KL divergence with respect to  $q(\mathbf{z}^*|\mathbf{x}, \phi, \lambda)$ .

## Theorem

VAE with the flow-based prior  $p(\mathbf{z}|\lambda)$  for latent code  $\mathbf{z}^*$  is equivalent to VAE with flow-based posterior  $q(\mathbf{z}^*|\mathbf{x}, \phi, \lambda)$  for latent code  $\mathbf{z}$ .

$$\begin{aligned} \mathcal{L}(\phi, \theta, \lambda) &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \log p(\mathbf{x}|\mathbf{z}, \theta) - \underbrace{KL(q(\mathbf{z}|\mathbf{x}, \phi) || p(\mathbf{z}|\textcolor{violet}{\lambda}))}_{\text{flow-based prior}} \\ &= \mathbb{E}_{q(\mathbf{z}^*|\mathbf{x}, \phi, \lambda)} \log p(\mathbf{x}|f(\mathbf{z}^*, \lambda), \theta) - \underbrace{KL(q(\mathbf{z}^*|\mathbf{x}, \phi, \textcolor{violet}{\lambda}) || p(\mathbf{z}^*))}_{\text{flow-based posterior}} \end{aligned}$$

# Recap of previous lecture

## Likelihood-free learning

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

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.

## Two sample test

$$H_0 : \pi(\mathbf{x}) = p(\mathbf{x}|\boldsymbol{\theta}), \quad H_1 : \pi(\mathbf{x}) \neq p(\mathbf{x}|\boldsymbol{\theta})$$

If test statistic  $T(\mathcal{S}_1, \mathcal{S}_2) < \alpha$ , then accept  $H_0$ , else reject it.

- ▶  $p(\mathbf{x}|\boldsymbol{\theta})$  minimizes the value of test statistic  $T(\mathcal{S}_1, \mathcal{S}_2)$ .
- ▶ It is hard to find an appropriate test statistic in high dimensions.  $T(\mathcal{S}_1, \mathcal{S}_2)$  could be learnable.

## Recap of previous lecture

- ▶ **Generator:** generative model  $\mathbf{x} = G(\mathbf{z})$ , which makes generated sample more realistic.
- ▶ **Discriminator:** a classifier  $D(\mathbf{x}) \in [0, 1]$ , which distinguishes real samples from generated samples.

### GAN optimality theorem

The minimax game

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

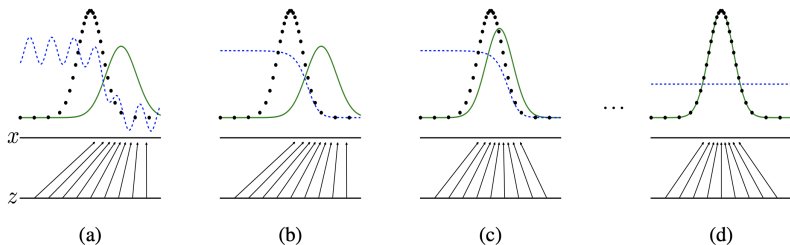
has the global optimum  $\pi(\mathbf{x}) = p(\mathbf{x}|\theta)$ , in this case  $D^*(\mathbf{x}) = 0.5$ .

$$\min_G V(G, D^*) = \min_G [2JSD(\pi||p) - \log 4] = -\log 4, \quad \pi(\mathbf{x}) = p(\mathbf{x}|\theta).$$

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.

## Recap of previous lecture

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



# Outline

1. Vanishing gradients and mode collapse
2. Adversarial variational Bayes
3. Wasserstein distance
4. Wasserstein GAN

# Outline

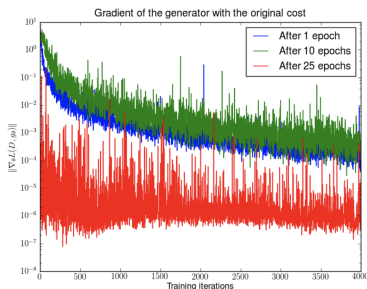
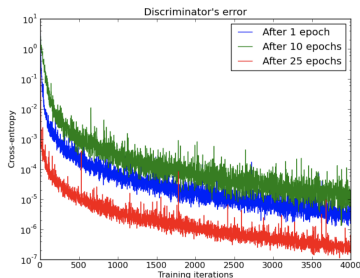
1. Vanishing gradients and mode collapse
2. Adversarial variational Bayes
3. Wasserstein distance
4. Wasserstein GAN

# Vanishing gradients

## Objective

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

Early in learning,  $G$  is poor,  $D$  can reject samples with high confidence. In this case,  $\log(1 - D(G(\mathbf{z}, \theta), \phi))$  saturates.



---

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



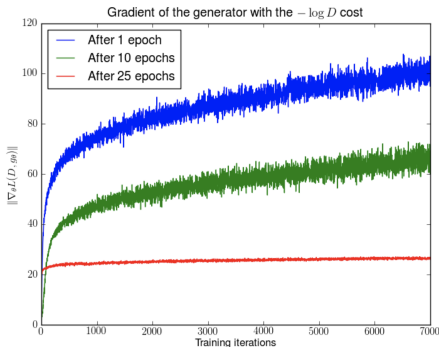
# Vanishing gradients

## Objective

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

## Non-saturating GAN

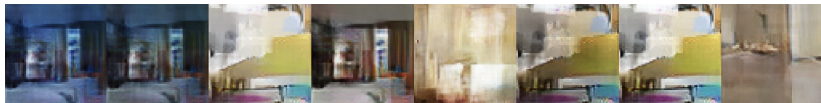
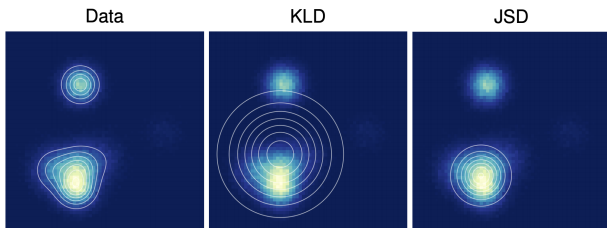
- ▶ Maximize  $\log D(G(\mathbf{z}))$  instead of minimizing  $\log(1 - D(G(\mathbf{z})))$ .
- ▶ Gradients are getting much stronger, but the training is unstable (with increasing mean and variance).



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

# 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*

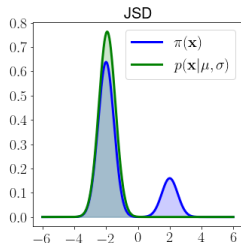
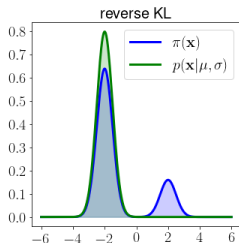
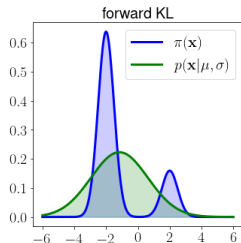
# Jensen-Shannon vs Kullback-Leibler

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

## Mode covering vs mode seeking

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

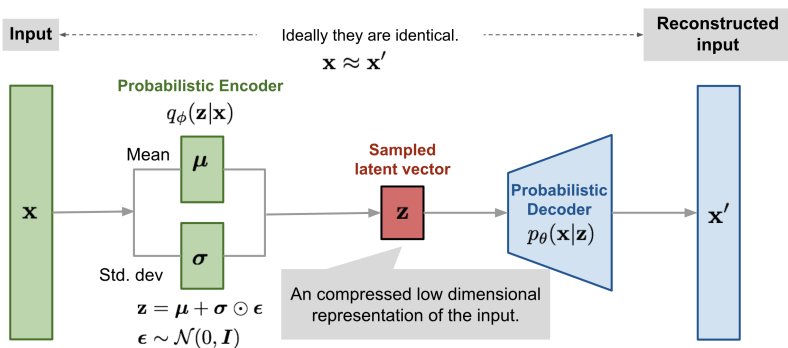
$$JSD(\pi||p) = \frac{1}{2} \left[ KL \left( \pi(\mathbf{x}) || \frac{\pi(\mathbf{x}) + p(\mathbf{x})}{2} \right) + KL \left( p(\mathbf{x}) || \frac{\pi(\mathbf{x}) + p(\mathbf{x})}{2} \right) \right]$$



# Outline

1. Vanishing gradients and mode collapse
2. Adversarial variational Bayes
3. Wasserstein distance
4. Wasserstein GAN

# VAE recap



- ▶ Encoder  $q(\mathbf{z}|\mathbf{x}, \phi) = \mathcal{N}(\mathbf{z}|\mu_{\phi}(\mathbf{x}), \sigma_{\phi}(\mathbf{x}))$ .
- ▶ Variational posterior  $q(\mathbf{z}|\mathbf{x}, \phi)$  originally approximates the true posterior  $p(\mathbf{z}|\mathbf{x}, \theta)$ .
- ▶ Which methods are you already familiar with to make the posterior is more flexible?

image credit:

<https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html>

# Adversarial Variational Bayes

## ELBO objective

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

What is the problem to make the variational posterior model an implicit model?

- ▶ The first term is the reconstruction loss that needs only samples from  $q(\mathbf{z}|\mathbf{x}, \phi)$  to evaluate.
- ▶ Reparametrization trick allows to get gradients of the reconstruction loss

$$\begin{aligned}\nabla_{\phi} \int q(\mathbf{z}|\mathbf{x}, \phi) f(\mathbf{z}) d\mathbf{z} &= \nabla_{\phi} \int r(\epsilon) f(\mathbf{z}) d\epsilon \\ &= \int r(\epsilon) \nabla_{\phi} f(g(\mathbf{x}, \epsilon, \phi)) d\epsilon \approx \nabla_{\phi} f(g(\mathbf{x}, \epsilon^*, \phi)),\end{aligned}$$

where  $\epsilon^* \sim r(\epsilon)$ ,  $\mathbf{z} = g(\mathbf{x}, \epsilon, \phi)$ ,  $\mathbf{z} \sim q(\mathbf{z}|\mathbf{x}, \phi)$ .

# Adversarial Variational Bayes

## ELBO objective

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

What is the problem to make the variational posterior model an implicit model?

- ▶ The second term requires the explicit value of  $q(\mathbf{z}|\mathbf{x}, \phi)$ .
- ▶ We could join second and third terms:

$$KL = \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \log \frac{q(\mathbf{z}|\mathbf{x}, \phi)}{p(\mathbf{z})} = \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \log \frac{q(\mathbf{z}|\mathbf{x}, \phi)\pi(\mathbf{x})}{p(\mathbf{z})\pi(\mathbf{x})}.$$

- ▶ We have to estimate density ratio

$$r(\mathbf{x}, \mathbf{z}) = \frac{q_1(\mathbf{x}, \mathbf{z})}{q_2(\mathbf{x}, \mathbf{z})} = \frac{q(\mathbf{z}|\mathbf{x}, \phi)\pi(\mathbf{x})}{p(\mathbf{z})\pi(\mathbf{x})}.$$

## Density ratio trick

Consider two distributions  $q_1(\mathbf{x})$ ,  $q_2(\mathbf{x})$  and probabilistic model

$$p(\mathbf{x}|y) = \begin{cases} q_1(\mathbf{x}), & \text{if } y = 1, \\ q_2(\mathbf{x}), & \text{if } y = 0, \end{cases} \quad y \sim \text{Bern}(0.5).$$

## Density ratio

$$\begin{aligned} \frac{q_1(\mathbf{x})}{q_2(\mathbf{x})} &= \frac{p(\mathbf{x}|y=1)}{p(\mathbf{x}|y=0)} = \frac{p(y=1|\mathbf{x})p(\mathbf{x})}{p(y=1)} \bigg/ \frac{p(y=0|\mathbf{x})p(\mathbf{x})}{p(y=0)} = \\ &= \frac{p(y=1|\mathbf{x})}{p(y=0|\mathbf{x})} = \frac{p(y=1|\mathbf{x})}{1 - p(y=1|\mathbf{x})} = \frac{D(\mathbf{x})}{1 - D(\mathbf{x})} \end{aligned}$$

Here  $D(\mathbf{x})$  is a discriminator model the output of which is a probability that  $\mathbf{x}$  is a sample from  $q_1(\mathbf{x})$  rather than from  $q_2(\mathbf{x})$ .

$$\max_D [\mathbb{E}_{q_1(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{q_2(\mathbf{x})} \log(1 - D(\mathbf{x}))]$$



## Density ratio trick

$$r(\mathbf{x}, \mathbf{z}) = \frac{q_1(\mathbf{x}, \mathbf{z})}{q_2(\mathbf{x}, \mathbf{z})} = \frac{q(\mathbf{z}|\mathbf{x}, \phi)\pi(\mathbf{x})}{p(\mathbf{z})\pi(\mathbf{x})}.$$

## Density ratio

$$\frac{q_1(\mathbf{x}, \mathbf{z})}{q_2(\mathbf{x}, \mathbf{z})} = \frac{p(y=1|\mathbf{x}, \mathbf{z})}{1 - p(y=1|\mathbf{x}, \mathbf{z})} = \frac{D(\mathbf{x}, \mathbf{z})}{1 - D(\mathbf{x}, \mathbf{z})}$$

## Adversarial Variational Bayes

$$\max_D [\mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \log D(\mathbf{x}, \mathbf{z}) + \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{p(\mathbf{z})} \log(1 - D(\mathbf{x}, \mathbf{z}))]$$

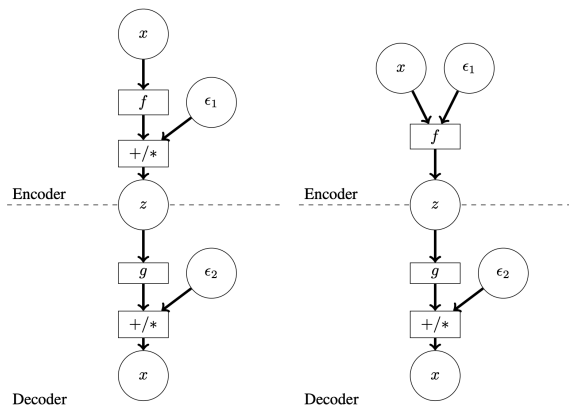
Monte-Carlo estimation for KL divergence:

$$KL(q(\mathbf{z}|\mathbf{x}, \phi) || p(\mathbf{z})) \approx \frac{D(\mathbf{x}, \mathbf{z})}{1 - D(\mathbf{x}, \mathbf{z})}.$$

# Adversarial Variational Bayes

## ELBO objective

$$\mathcal{L}(\phi, \theta) = \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \left[ \log p(\mathbf{x}|\mathbf{z}, \theta) - \log \frac{q(\mathbf{z}|\mathbf{x}, \phi)}{p(\mathbf{z})} \right] \rightarrow \max_{\phi, \theta}.$$



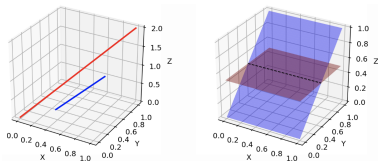
Mescheder L., Nowozin S., Geiger A. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks, 2017

# Outline

1. Vanishing gradients and mode collapse
2. Adversarial variational Bayes
3. Wasserstein distance
4. Wasserstein GAN

# Informal theoretical results

- ▶ Since  $\mathbf{z}$  usually has lower dimensionality compared to  $\mathbf{x}$ , manifold  $G(\mathbf{z}, \boldsymbol{\theta})$  has a measure 0 in  $\mathbf{x}$  space. Hence, support of  $p(\mathbf{x}|\boldsymbol{\theta})$  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. We are not able to learn anything by backproping through it.
- ▶ For such low-dimensional disjoint manifolds
$$KL(\pi||p) = KL(p||\pi) = \infty, \quad JSD(\pi||p) = \log 2$$
- ▶ Adding continuous noise to the inputs of the discriminator smoothes the distributions of the probability mass.

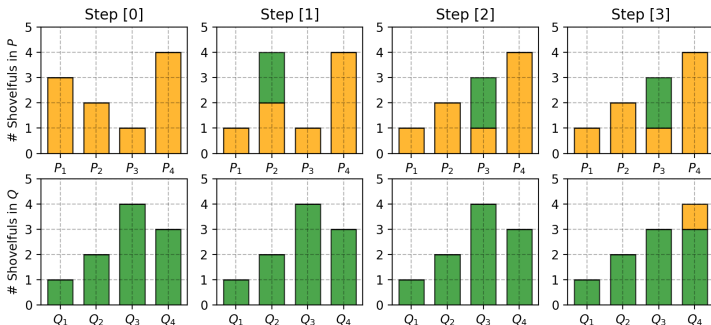
---

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**. 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.



$$W(P, Q) = 2(\text{step 1}) + 2(\text{step 2}) + 1(\text{step 3}) = 5$$

## 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.

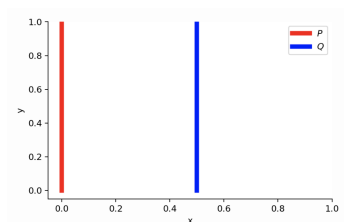
For better understanding of transportation plan function  $\gamma$ , try to write down the plan for previous discrete case.

# 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  $G(\mathbf{z}, \theta)$  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).



# Outline

1. Vanishing gradients and mode collapse
2. Adversarial variational Bayes
3. Wasserstein distance
4. Wasserstein GAN

# Wasserstein GAN

## Wasserstein distance

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

The infimum across all possible joint distributions in  $\Gamma(\pi, p)$  is intractable.

## Theorem (Kantorovich-Rubinstein duality)

$$W(\pi||p) = \frac{1}{K} \max_{\|f\|_L \leq K} [\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}) - \mathbb{E}_{p(\mathbf{x})} f(\mathbf{x})],$$

where  $\|f\|_L \leq K$  are  $K$ -Lipschitz continuous functions  
( $f : \mathcal{X} \rightarrow \mathbb{R}$ )

$$|f(\mathbf{x}_1) - f(\mathbf{x}_2)| \leq K \|\mathbf{x}_1 - \mathbf{x}_2\|, \quad \text{for all } \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}.$$

Now we need only samples to get Monte Carlo estimate for  $W(\pi||p)$ .

# Wasserstein GAN

## Theorem (Kantorovich-Rubinstein duality)

$$W(\pi||p) = \frac{1}{K} \max_{\|f\|_L \leq K} [\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}) - \mathbb{E}_{p(\mathbf{x})} f(\mathbf{x})] ,$$

- ▶ Now we have to ensure that  $f$  is  $K$ -Lipschitz continuous.
- ▶ Let  $f(\mathbf{x}, \phi)$  be a feedforward neural network parametrized by  $\phi$ .
- ▶ If parameters  $\phi$  lie in a compact set  $\Phi$  then  $f(\mathbf{x}, \phi)$  will be  $K$ -Lipschitz continuous function.
- ▶ Let the parameters be clamped to a fixed box  $\Phi \in [-c, c]^d$  (e.x.  $c = 0.01$ ) after each gradient update.

$$\begin{aligned} K \cdot W(\pi||p) &= \max_{\|f\|_L \leq K} [\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}) - \mathbb{E}_{p(\mathbf{x})} f(\mathbf{x})] \geq \\ &\geq \max_{\phi \in \Phi} [\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}, \phi) - \mathbb{E}_{p(\mathbf{x})} f(\mathbf{x}, \phi)] \end{aligned}$$

# Wasserstein GAN

## Standard GAN objective

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

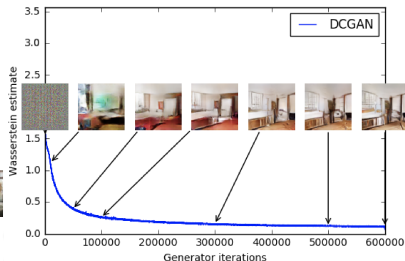
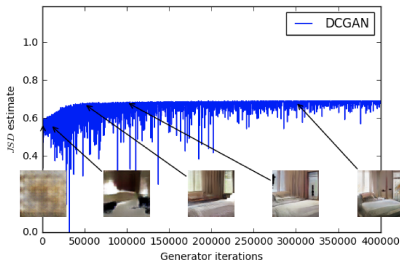
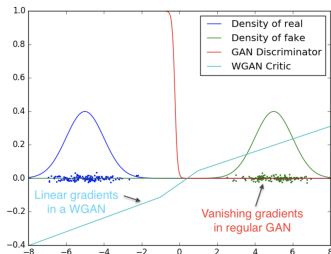
## WGAN objective

$$\min_{\theta} W(\pi||p) \approx \min_{\theta} \max_{\phi \in \Phi} [\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}, \phi) - \mathbb{E}_{p(\mathbf{z})} f(G(\mathbf{z}, \theta), \phi)] .$$

- ▶ Discriminator  $D$  is similar to the function  $f$ , but not the same (it is not a classifier anymore). In the WGAN model, function  $f$  is usually called **critic**.
- ▶ "*Weight clipping is a clearly terrible way to enforce a Lipschitz constraint*". If the clipping parameter  $c$  is too large, it is hard to train the critic till optimality. If the clipping parameter  $c$  is too small, it could lead to vanishing gradients.

# Wasserstein GAN

- ▶ WGAN has non-zero gradients for disjoint supports.
- ▶  $JSD(\pi||p)$  correlates poorly with the sample quality. Stays constant nearly maximum value  $\log 2 \approx 0.69$ .
- ▶  $W(\pi||p)$  is highly correlated with the sample quality.



# Summary

- ▶ Mode collapse and vanishing gradients are the two main problems of vanilla GAN. Lots of tips and tricks has to be used to make the GAN training is stable and scalable.
- ▶ Adversarial Variational Bayes uses density ratio trick to get more powerful variational posterior.
- ▶ 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.
- ▶ Kantorovich-Rubinstein duality gives the way to calculate the EM distance using only samples.
- ▶ Wasserstein GAN uses Kantorovich-Rubinstein duality for getting Earth Mover distance as model objective.