Deep Generative Models

Lecture 10

Roman Isachenko

Ozon Masters

Spring, 2022

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

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

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

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

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.

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

Recap of previous lecture

ELBO objective

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

What is the problem to make the variational posterior model an **implicit** model?
We have to estimate density ratio

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

Adversarial Variational Bayes

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

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

Recap of previous lecture

Standard GAN

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

Main problems

- Vanishing gradients (non-saturating GAN does not suffer of it);
- Mode collapse (caused by behaviour of Jensen-Shannon divergence).

Informal theoretical results

Distribution of real images $\pi(\mathbf{x})$ and distribution of generated images $p(\mathbf{x}|\boldsymbol{\theta})$ are low-dimensional and have disjoint supports. In this case

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

Goodfellow I. J. et al. Generative Adversarial Networks, 2014
Arjovsky M., Bottou L. Towards Principled Methods for Training Generative Adversarial Networks, 2017

Recap of previous lecture

Wasserstein distance

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

- $\Gamma(\pi, p)$ – the set of all joint distributions $\Gamma(\mathbf{x}, \mathbf{y})$ with marginals π and p ($\int \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{x} = p(\mathbf{y})$, $\int \gamma(\mathbf{x}, \mathbf{y}) d\mathbf{y} = \pi(\mathbf{x})$)
- $\gamma(\mathbf{x}, \mathbf{y})$ – transportation plan (the amount of "dirt" that should be transported from point $\mathbf{x}$ to point $\mathbf{y}$).
- $\gamma(\mathbf{x}, \mathbf{y})$ – the amount, $\|\mathbf{x} - \mathbf{y}\|$– the distance.

Theorem (Kantorovich-Rubinstein duality)

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

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

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

Outline

1. WGAN with Gradient Penalty
2. Spectral Normalization GAN
3. f-divergence minimization

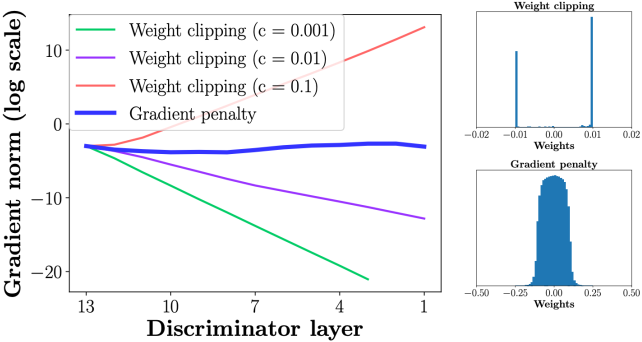
Outline

1. WGAN with Gradient Penalty
2. Spectral Normalization GAN
3. f-divergence minimization

Wasserstein GAN with Gradient Penalty

Weight clipping analysis

- The critic ignores higher moments of the data distribution.
- The gradients either grow or decay exponentially.



Gradient penalty makes the gradients more stable.

Gulrajani I. et al. Improved Training of Wasserstein GANs, 2017

Wasserstein GAN with Gradient Penalty

Theorem

Let $\pi(\mathbf{x})$ and $p(\mathbf{x})$ be two distribution in $\mathcal{X}$, a compact metric space. Then, there is 1-Lipschitz function f^* which is the optimal solution of

$$\max_{\|f\|_L \leq 1} \left[\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}) - \mathbb{E}_{p(\mathbf{x})} f(\mathbf{x})\right].$$

Let γ be the optimal transportation plan between $\pi(\mathbf{x})$ and $p(\mathbf{x})$. Then, if f^* is differentiable, $\gamma(\mathbf{x} = \mathbf{y}) = 0$ and $\hat{\mathbf{x}}_t = t\mathbf{x} + (1-t)\mathbf{y}$ with $\mathbf{x} \sim \pi(\mathbf{x})$, $\mathbf{y} \sim p(\mathbf{x}|\boldsymbol{\theta})$, $t \in [0, 1]$ it holds that

$$\mathbb{P}_{(\mathbf{x},\mathbf{y})\sim\gamma} \left[\nabla f^*(\hat{\mathbf{x}}_t) = \frac{\mathbf{y} - \hat{\mathbf{x}}_t}{\|\mathbf{y} - \hat{\mathbf{x}}_t\|}\right] = 1.$$

Corollary

f^* has gradient norm 1 almost everywhere under $\pi(\mathbf{x})$ and $p(\mathbf{x})$.

Gulrajani I. et al. Improved Training of Wasserstein GANs, 2017

Wasserstein GAN with Gradient Penalty

A differentiable function is 1-Lipschtiz if and only if it has gradients with norm at most 1 everywhere.

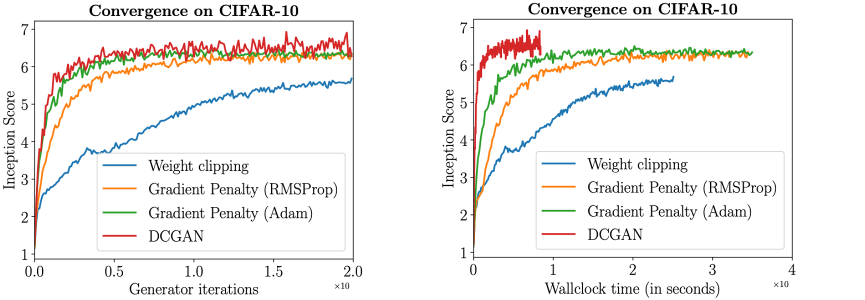
Gradient penalty

$$W(\pi||p) = \underbrace{\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}) - \mathbb{E}_{p(\mathbf{x})} f(\mathbf{x})}_{\text{original critic loss}} + \lambda \underbrace{\mathbb{E}_{U[0,1]} \left[(\|\nabla_{\hat{\mathbf{x}}} f(\hat{\mathbf{x}})\|_2 - 1)^2 \right]}_{\text{gradient penalty}},$$

- Samples $\hat{\mathbf{x}}_t = t\mathbf{x} + (1-t)\mathbf{y}$ with $t \in [0, 1]$ are uniformly sampled along straight lines between pairs of points: $\mathbf{x}$ from the data distribution $\pi(\mathbf{x})$ and $\mathbf{y}$ from the generator distribution $p(\mathbf{x}|\boldsymbol{\theta})$.
- Enforcing the unit gradient norm constraint everywhere is intractable, it turns out to be sifficient to enforce it only along these straight lines.

Gulrajani I. et al. Improved Training of Wasserstein GANs, 2017

Wasserstein GAN with Gradient Penalty



WGAN-GP convergence

Min. score	Only GAN	Only WGAN-GP	Both succeeded	Both failed
1.0	0	8	192	0
3.0	1	88	110	1
5.0	0	147	42	11
7.0	1	104	5	90
9.0	0	0	0	200

Gulrajani I. et al. Improved Training of Wasserstein GANs, 2017

Spectral Normalization GAN

Definition

$\|\mathbf{A}\|_2$ is a *spectral norm* of matrix $\mathbf{A}$:

$$\|\mathbf{A}\|_2 = \max_{\mathbf{h}\neq 0} \frac{\|\mathbf{A}\mathbf{h}\|_2}{\|\mathbf{h}\|_2} = \max_{\|\mathbf{h}\|_2 \leq 1} \|\mathbf{A}\mathbf{h}\|_2 = \lambda_{\max}(\mathbf{A}^T\mathbf{A}),$$

where $\lambda_{\max}(\mathbf{A}^T\mathbf{A})$ is the largest eigenvalue value of $\mathbf{A}^T\mathbf{A}$.

Statement 1

if g is a K-Lipschitz function then

$$\|\mathbf{g}\|_L \leq K = \sup_{\mathbf{x}} \|\nabla \mathbf{g}(\mathbf{x}))\|_2.$$

Statement 2

Lipschitz norm of superposition is bounded above by product of Lipschitz norms

$$\|\mathbf{g}_1 \circ \mathbf{g}_2\|_L \leq \|\mathbf{g}_1\|_L \cdot \|\mathbf{g}_2\|_L$$

Miyato T. et al. Spectral Normalization for Generative Adversarial Networks, 2018

Outline

1. WGAN with Gradient Penalty

2. Spectral Normalization GAN

3. f-divergence minimization

Spectral Normalization GAN

Let consider the critic $f(\mathbf{x}, \phi)$ of the following form:

$$f(\mathbf{x}, \phi) = \mathbf{W}_{K+1}\sigma_K(\mathbf{W}_K\sigma_{K-1}(\dots\sigma_1(\mathbf{W}_1\mathbf{x})\dots)).$$

This feedforward network is a superposition of simple functions.

- σ_k is a pointwise nonlinearities. We assume that $\|\sigma_k\|_L = 1$ (it holds for ReLU).
- $\mathbf{g}(\mathbf{x}) = \mathbf{W}\mathbf{x}$ is a linear transformation ($\nabla \mathbf{g}(\mathbf{x}) = \mathbf{W}$).

$$\|\mathbf{g}\|_L \leq \sup_{\mathbf{x}} \|\nabla \mathbf{g}(\mathbf{x})\|_2 = \|\mathbf{W}\|_2.$$

Critic spectral norm

$$\|f\|_L \leq \|\mathbf{W}_{K+1}\| \cdot \prod_{k=1}^{K} \|\sigma_k\|_L \cdot \|\mathbf{W}_k\|_2 = \prod_{k=1}^{K+1} \|\mathbf{W}_k\|_2.$$

If we replace the weights in the critic $f(\mathbf{x}, \phi)$ by $\mathbf{W}_k^{SN} = \mathbf{W}_k / \|\mathbf{W}_k\|_2$, we will get $\|f\|_L \leq 1$.

Miyato T. et al. Spectral Normalization for Generative Adversarial Networks, 2018

Spectral Normalization GAN

How to compute $\|\mathbf{W}\|_2 = \lambda_{\max}(\mathbf{W}^T\mathbf{W})$?
If we apply SVD to compute the $\|\mathbf{W}\|_2$ at each iteration, the algorithm becomes intractable.

Power iteration method

- $\mathbf{u}_0$ – random vector.
- for $k = 0, \dots, n-1$: (n is a large enough number of steps)

$$\mathbf{v}_{k+1} = \frac{\mathbf{W}^T\mathbf{u}_k}{\|\mathbf{W}^T\mathbf{u}_k\|}, \quad \mathbf{u}_{k+1} = \frac{\mathbf{W}\mathbf{v}_{k+1}}{\|\mathbf{W}\mathbf{v}_{k+1}\|}.$$

- approximate the spectral norm

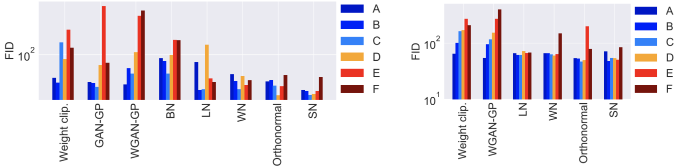
$$\|\mathbf{W}\|_2 = \lambda_{\max}(\mathbf{W}^T\mathbf{W}) \approx \mathbf{u}_n^T\mathbf{W}\mathbf{v}_n.$$

Miyato T. et al. Spectral Normalization for Generative Adversarial Networks, 2018

Spectral Normalization GAN

Algorithm 1 SGD with spectral normalization

- Initialize $\tilde{\boldsymbol{u}}_l \in \mathcal{R}^{d_l}$ for $l = 1, \ldots, L$ with a random vector (sampled from isotropic distribution).
- For each update and each layer l:
  1. Apply power iteration method to a unnormalized weight W^l:
  $$\tilde{\boldsymbol{v}}_l \leftarrow (W^l)^{\mathrm{T}} \tilde{\boldsymbol{u}}_l / \|(W^l)^{\mathrm{T}} \tilde{\boldsymbol{u}}_l\|_2 \tag{20}$$
  $$\tilde{\boldsymbol{u}}_l \leftarrow W^l \tilde{\boldsymbol{v}}_l / \|W^l \tilde{\boldsymbol{v}}_l\|_2 \tag{21}$$
  2. Calculate $\bar{W}_{\mathrm{SN}}$ with the spectral norm:
  $$\bar{W}^l_{\mathrm{SN}}(W^l) = W^l / \sigma(W^l), \text{ where } \sigma(W^l) = \tilde{\boldsymbol{u}}_l^{\mathrm{T}} W^l \tilde{\boldsymbol{v}}_l \tag{22}$$
  3. Update W^l with SGD on mini-batch dataset $\mathcal{D}_M$ with a learning rate α:
  $$W^l \leftarrow W^l - \alpha \nabla_{W^l} \ell(\bar{W}^l_{\mathrm{SN}}(W^l), \mathcal{D}_M) \tag{23}$$



(a) CIFAR-10 (b) STL-10

Miyato T. et al. Spectral Normalization for Generative Adversarial Networks, 2018

Outline

1. WGAN with Gradient Penalty

2. Spectral Normalization GAN

3. f-divergence minimization

Divergences

- Forward KL divergence in maximum likelihood estimation.
- Reverse KL in variational inference.
- JS divergence in standard GAN.
- Wasserstein distance in WGAN.

What is a divergence?

Let $\mathcal{S}$ be the set of all possible probability distributions. Then $D : \mathcal{S} \times \mathcal{S} \to \mathbb{R}$ is a divergence if

- $D(\pi || p) \geq 0$ for all $\pi, p \in \mathcal{S}$;
- $D(\pi || p) = 0$ if and only if $\pi \equiv p$.

General divergence minimization task

$$\min_{p} D(\pi || p)$$

Chalenge

We do not know the real distribution $\pi(\mathbf{x})$!

f-divergence family

f-divergence

$$D_f(\pi||p) = \mathbb{E}_{p(\mathbf{x})} f\left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})}\right) = \int p(\mathbf{x}) f\left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})}\right) d\mathbf{x}.$$

Here $f : \mathbb{R}_+ \to \mathbb{R}$ is a convex, lower semicontinuous function satisfying $f(1) = 0$.

Name	$D_f(P\|Q)$	Generator $f(u)$
Kullback-Leibler	$\int p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x$	$u \log u$
Reverse KL	$\int q(x) \log \frac{q(x)}{p(x)} \, \mathrm{d}x$	$-\log u$
Pearson χ^2	$\int \frac{(q(x)-p(x))^2}{p(x)} \, \mathrm{d}x$	$(u-1)^2$
Squared Hellinger	$\int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 \mathrm{d}x$	$(\sqrt{u}-1)^2$
Jensen-Shannon	$\frac{1}{2} \int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \, \mathrm{d}x$	$-(u+1) \log \frac{1+u}{2} + u \log u$
GAN	$\int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \, \mathrm{d}x - \log(4)$	$u \log u - (u+1) \log(u+1)$

Nowozin S., Cseke B., Tomioka R. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization, 2016

f-divergence family

Fenchel conjugate

$$f^*(t) = \sup_{u \in \mathrm{dom}_f} (ut - f(u)), \quad f(u) = \sup_{t \in \mathrm{dom}_{f^*}} (ut - f^*(t))$$

Important property: $f^{**} = f$ for convex f.

f-divergence

$$D_f(\pi || p) = \mathbb{E}_{p(\mathbf{x})} f\left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})}\right) = \int p(\mathbf{x}) f\left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})}\right) d\mathbf{x} =$$

$$= \int p(\mathbf{x}) \sup_{t \in \mathrm{dom}_{f^*}} \left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})} t - f^*(t)\right) d\mathbf{x} =$$

$$= \int \sup_{t \in \mathrm{dom}_{f^*}} (\pi(\mathbf{x}) t - p(\mathbf{x}) f^*(t)) \, d\mathbf{x}.$$

Here we seek value of t, which gives us maximum value of $\pi(\mathbf{x})t - p(\mathbf{x})f^*(t)$, for each data point $\mathbf{x}$.

Nowozin S., Cseke B., Tomioka R. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization, 2016

f-divergence family

f-divergence

$$D_f(\pi||p) = \mathbb{E}_{p(\mathbf{x})} f\left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})}\right) = \int p(\mathbf{x}) f\left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})}\right) d\mathbf{x}.$$

Variational f-divergence estimation

$$D_f(\pi||p) = \int \sup_{t \in \mathrm{dom}_{f^*}} \left(\pi(\mathbf{x})t - p(\mathbf{x})f^*(t)\right) d\mathbf{x} \geq$$

$$\geq \sup_{T \in \mathcal{T}} \int \left(\pi(\mathbf{x})T(\mathbf{x}) - p(\mathbf{x})f^*(T(\mathbf{x}))\right) d\mathbf{x} =$$

$$= \sup_{T \in \mathcal{T}} \left[\mathbb{E}_\pi T(\mathbf{x}) - \mathbb{E}_p f^*(T(\mathbf{x}))\right]$$

This is a lower bound because of Jensen-Shannon inequality and restricted class of functions $\mathcal{T} : \mathcal{X} \to \mathbb{R}$.

Nowozin S., Cseke B., Tomioka R. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization, 2016

f-divergence family

Variational divergence estimation

$$D_f(\pi || p) \geq \sup_{T \in \mathcal{T}} \left[\mathbb{E}_\pi T(\mathbf{x}) - \mathbb{E}_p f^*(T(\mathbf{x}))\right]$$

The lower bound is tight for $T^*(\mathbf{x}) = f'\left(\frac{\pi(\mathbf{x})}{p(\mathbf{x})}\right)$.

Example (JSD)

- Let define function f and its conjugate f^*

$$f(u) = u \log u - (u+1)\log(u+1), \quad f^*(t) = -\log(1 - e^t).$$

- Let reparametrize $T(\mathbf{x}) = \log D(\mathbf{x})$.

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

Nowozin S., Cseke B., Tomioka R. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization, 2016

f-divergence family

Variational divergence estimation

$$D_f(\pi||p) \geq \sup_{T \in \mathcal{T}} \left[\mathbb{E}_\pi T(\mathbf{x}) - \mathbb{E}_p f^*(T(\mathbf{x}))\right]$$

Note: To evaluate lower bound we only need samples from $\pi(\mathbf{x})$ and $p(\mathbf{x})$. Hence, we could fit implicit generative model.

(a) GAN (b) KL (c) Squared Hellinger

Nowozin S., Cseke B., Tomioka R. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization, 2016

Summary

- Weight clipping is a terrible way to enforce Lipschitzness. Gradient Penalty works better.
- Spectral normalization is a weight normalization technique to enforce Lipshitzness, which is helpful for generator and discriminator.
- f-divergence family is a unified framework for divergence minimization, which uses variational approximation. Standard GAN is a special case of it.