WHAT IS CRITICAL IN GAN TRAINING?

Hao-Wen Dong

Music and Computing (MAC) Lab, CITI, Academia Sinica

Outlines

- **Generative Adversarial Networks** (GAN[1])
- **Wasserstein GANs** (WGAN[2])
- **Lipschitz Regularization**
  - **Spectral Normalization** (SN-GAN[3])
  - **Gradient Penalties** (WGAN-GP[4], DRAGAN[5], GAN-GP[6])
- **What is critical in GAN training?**

Generative Adversarial Networks

Generative Adversarial Networks (GANs)

- **Two-player game** between the **discriminator *D*** and the **generator *G***

(to assign real data a 1) (to assign fake data a 0)

$$J^{(D)}(D,G) = -\underset{x\sim p_{data}}{\mathbb{E}}[\log D(x)] - \underset{z\sim p_z}{\mathbb{E}}\left[\log\left(1-D\left(G(z)\right)\right)\right]$$

data distribution — prior distribution — fake data

$$J^{(G)}(G) = \underset{z\sim p_z}{\mathbb{E}}\left[\log\left(1-D\left(G(z)\right)\right)\right]$$

(to make *D* assign generated data a 1)

Original Algorithm

(Goodfellow *et. al* [1])

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right]$$

end for

- Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
- Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

Original Convergence Proof

(one of)

Proposition 1. *For G fixed, the optimal discriminator D is*

$$D^*_G(\boldsymbol{x}) = \frac{p_{data}(\boldsymbol{x})}{p_{data}(\boldsymbol{x}) + p_g(\boldsymbol{x})}$$

For a finite data set X, we only have

$$p_{data} = \begin{cases} 1, & x \in X \\ 0, & otherwise \end{cases}$$

hard to optimize

(may need density estimation)

usually not the case

Proposition 2. *If G and D have enough capacity, and at each step of Algorithm 1, the discriminator is allowed to reach its optimum given G, and p_g is updated so as to improve the criterion*

$$\mathbb{E}_{\boldsymbol{x}\sim p_{data}}[\log D^*_G(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x}\sim p_g}[\log(1 - D^*_G(\boldsymbol{x}))]$$

usually not the case

then p_g converges to p_{data}

if the criterion can be easily improved

Minimax and Non-saturating GANs

$$J^{(D)}(D, G) = -\underset{x \sim p_{data}}{\mathbb{E}}[\log D(x)] - \underset{z \sim p_z}{\mathbb{E}}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$$

Minimax: $J^{(G)}(G) = \underset{z \sim p_z}{\mathbb{E}}\left[\log\left(1 - D\left(G(z)\right)\right)\right]$

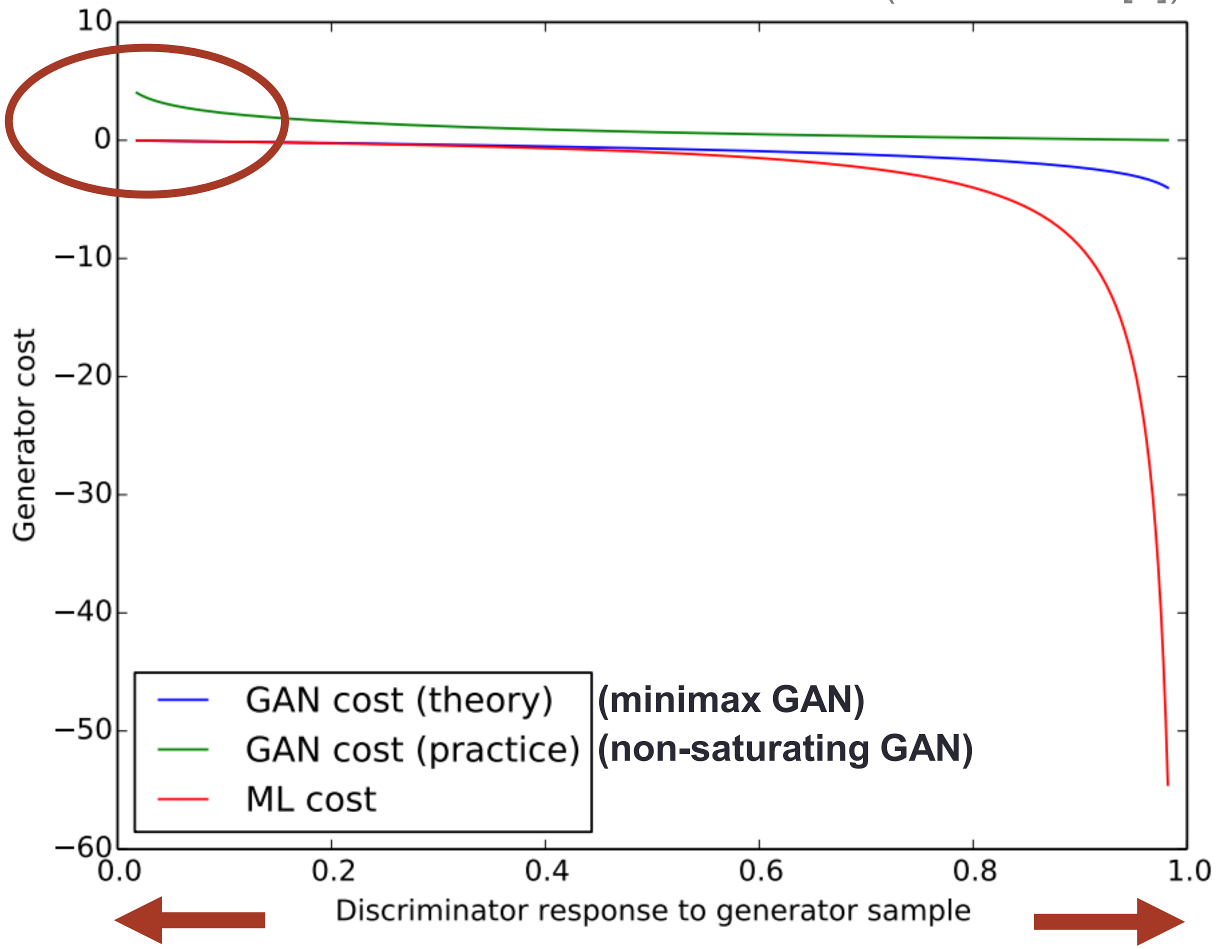
Non-saturating: $J^{(G)}(G) = -\underset{z \sim p_z}{\mathbb{E}}\left[\log\left(D\left(G(z)\right)\right)\right]$ **(used in practice)**

(won't stop training when *D* is stronger)

Comparisons of GANs

(Goodfellow [7])

Less training difficulties at the initial stage when *G* can hardly fool *D*



***D* is not fooled**

***D* is fooled**

Wasserstein GANs

Wasserstein Distance (Earth-Mover Distance)

$$W(\mathbb{P}_r, \mathbb{P}_\theta) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_\theta)} \mathbb{E}_{(x,y)\sim\gamma}[\|x - y\|]$$

Theorem 1. *Let $\mathbb{P}_r$ be a fixed distribution over $\mathcal{X}$. Let Z be a random variable (e.g Gaussian) over another space $\mathcal{Z}$. Let $g : \mathcal{Z} \times \mathbb{R}^d \to \mathcal{X}$ be a function, that will be denoted $g_\theta(z)$ with z the first coordinate and θ the second. Let $\mathbb{P}_\theta$ denote the distribution of $g_\theta(Z)$. Then,*

can be optimized easier

1. *If g is continuous in θ, so is $W(\mathbb{P}_r, \mathbb{P}_\theta)$.*
2. *If g is locally Lipschitz and satisfies regularity assumption 1, then $W(\mathbb{P}_r, \mathbb{P}_\theta)$ is continuous everywhere, and differentiable almost everywhere.*
3. *Statements 1-2 are false for the Jensen-Shannon divergence $JS(\mathbb{P}_r, \mathbb{P}_\theta)$ and all the KLs.*

Kantorovich-Rubinstein duality

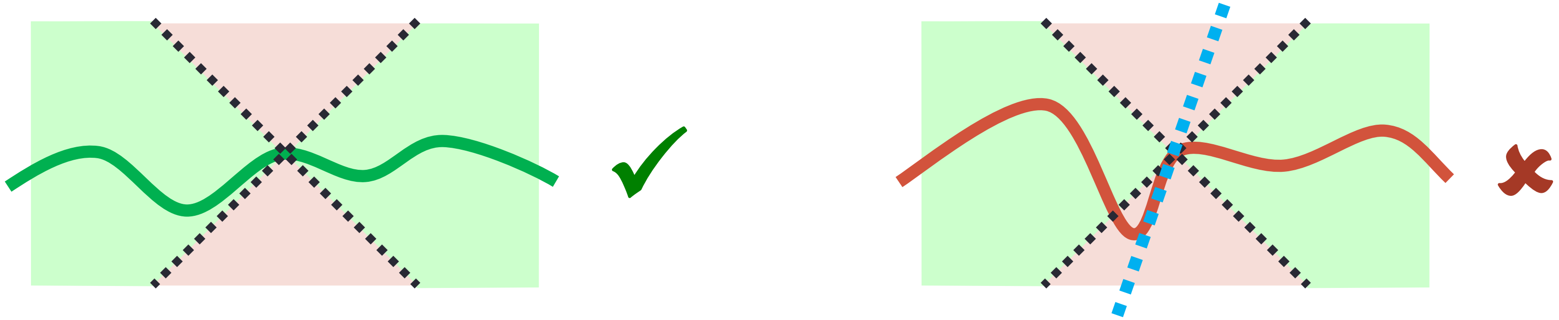
- **Kantorovich-Rubinstein duality**

$$W(\mathbb{P}_r, \mathbb{P}_\theta) = \sup_{\|f\|_L \le 1} \mathbb{E}_{x\sim\mathbb{P}_r}[f(x)] - \mathbb{E}_{x\sim\mathbb{P}_\theta}[f(x)]$$

all the 1-Lipschitz functions $f: X \to \Re$

- **Definition** A function $f: \Re \to \Re$ is called **Lipschitz continuous** if

$$\exists K \in \Re \quad s.t. \quad \forall x_1, x_2 \in \Re \quad |f(x_1) - f(x_2)| \le K|x_1 - x_2|$$



Wasserstein GAN

- **Key**: use a NN to estimate Wasserstein distance (and use it as critics for *G*)

$$J^{(\boldsymbol{D})}(D,G) = -\mathop{\mathbb{E}}_{x\sim p_{data}}[D(x)] - \mathop{\mathbb{E}}_{z\sim p_z}\left[D\big(G(z)\big)\right]$$

$$J^{(\boldsymbol{G})}(G) = \mathop{\mathbb{E}}_{z\sim p_z}\left[D\big(G(z)\big)\right]$$

- Original GAN (non-saturating)

$$J^{(\boldsymbol{D})}(D,G) = -\mathop{\mathbb{E}}_{x\sim p_{data}}[\log D(x)] - \mathop{\mathbb{E}}_{z\sim p_z}\left[\log\Big(1 - D\big(G(z)\big)\Big)\right]$$

$$J^{(\boldsymbol{G})}(G) = -\mathop{\mathbb{E}}_{z\sim p_z}\left[\log\Big(D\big(G(z)\big)\Big)\right]$$

Wasserstein GAN

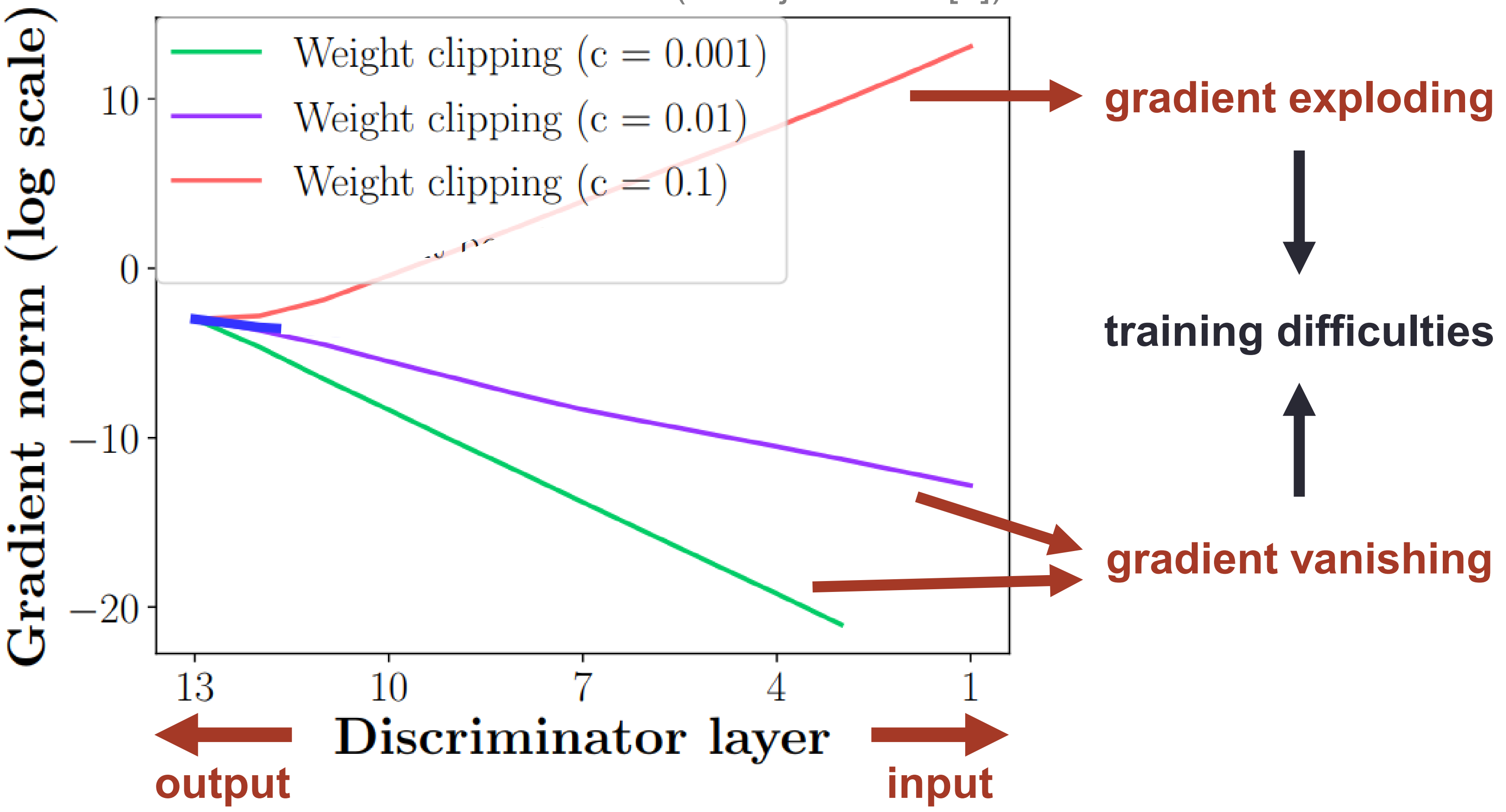
- **Problem**: such NN needs to satisfy a **Lipschitz constraint**
- Global regularization
  - **weight clipping** → original WGAN
  - **spectral normalization** → SNGAN
- Local regularization
  - **gradient penalties** → WGAN-GP

Lipschitz Regularization

Weight Clipping

- **Key**: clip the weights of the critic into $[-c, c]$

(Gulrajani *et. al* [4])



Spectral Normalization

- **Key**: constraining the spectral norm of each layer

- For each layer $g: \boldsymbol{h}_{in} \rightarrow \boldsymbol{h}_{out}$, by definition we have

$$\|g\|_{Lip} = \sup_{\boldsymbol{h}} \sigma\big(\nabla g(\boldsymbol{h})\big),$$

where

$$\underbrace{\sigma(A)}_{\text{spectral norm}} := \max_{\boldsymbol{h} \neq 0} \frac{\|A\boldsymbol{h}\|_2}{\|\boldsymbol{h}\|_2} = \underbrace{\max_{\|\boldsymbol{h}\|_2 \leq 1} \|A\boldsymbol{h}\|_2}_{\text{the largest singular value of A}}$$

Spectral Normalization

- For a **linear layer** $g(\boldsymbol{h}) = W\boldsymbol{h}$, $\|g\|_{Lip} = \sup_{\boldsymbol{h}} \sigma\big(\nabla g(\boldsymbol{h})\big) = \sup_{\boldsymbol{h}} \sigma(W) = \boxed{\sigma(W)}$
- For typical **activation layers** $a(h)$,

$\boxed{\|a\|_{Lip} = 1}$ for ReLU, LeakyReLU

$\|a\|_{Lip} = K$ for other common activation layers (e.g. sigmoid, tanh)

- With the inequality

$$\|f_1 \circ f_2\|_{Lip} \leq \|f_1\|_{Lip} \cdot \|f_2\|_{Lip},$$

we now have

$$\|f\|_{Lip} \leq \overbrace{\underbrace{\|W_1\|_{Lip}}_{\text{linear}} \cdot \underbrace{\|a_1\|_{Lip}}_{\text{activation}}}^{\text{layer 1}} \cdots \overbrace{\underbrace{\|W_L\|_{Lip}}_{\text{linear}} \cdot \underbrace{\|a_L\|_{Lip}}_{\text{activation}}}^{\text{layer L}} = \boxed{\prod_{l=1}^{L} \sigma(W_l)}$$

Spectral Normalization

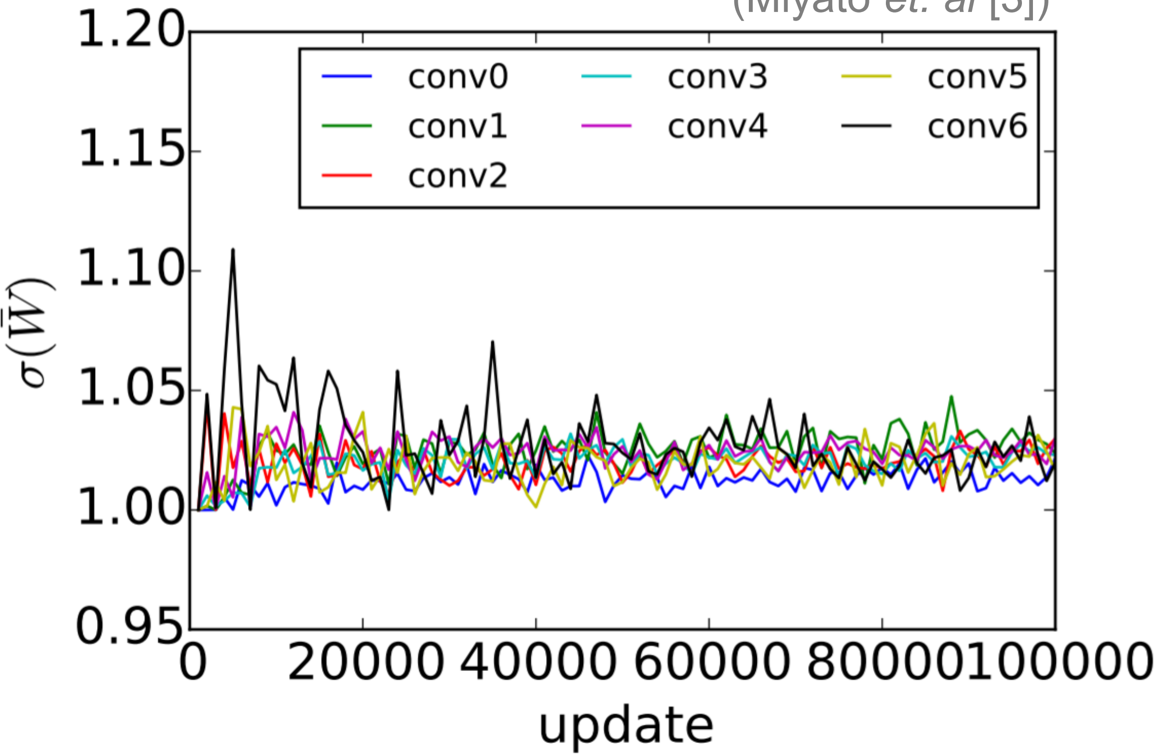
- **Spectral normalization**

$$\bar{W}_{SN}(W) := \frac{W}{\sigma(W)}$$

(W: weight matrix)

- Now we have $\|f\|_{Lip} \leq 1$ anywhere
- Fast approximation of $\sigma(W)$ using power iteration method (see the paper)

(Miyato *et. al* [3])



Gradient Penalties

- **Key**: punish the critic discriminator when it violate the Lipschitz constraint
- But it's impossible to enforce punishment anywhere

$$\mathbb{E}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}[(\|\nabla_{\hat{x}}D(\hat{x})\|_2 - 1)^2]$$

punish it when the gradient norm get away from 1 → **make the gradient norm stay close to 1**

- Two common sampling approaches for $\hat{x}$

WGAN-GP $\mathbb{P}_{\hat{x}} = \alpha\mathbb{P}_x + (1-\alpha)\mathbb{P}_g$ → **between data and model distribution**

DRAGAN $\mathbb{P}_{\hat{x}} = \alpha\mathbb{P}_x + (1-\alpha)\mathbb{P}_{noise}$ → **around data distribution**

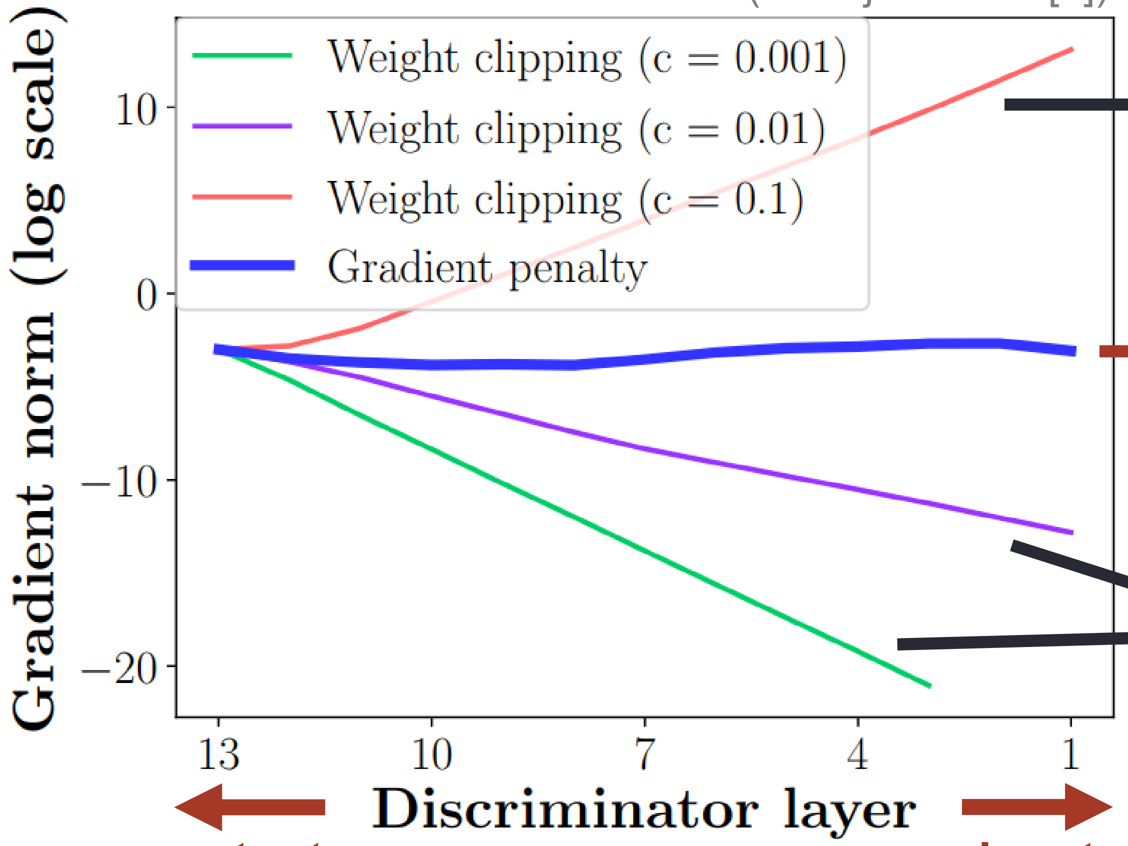
$$\alpha\sim U[0,1]$$

WGAN-GP

$$\mathbb{P}_{\hat{x}} = \alpha \mathbb{P}_x + (1 - \alpha)\mathbb{P}_g$$

between data and model distribution

(Gulrajani *et. al* [4])



gradient exploding

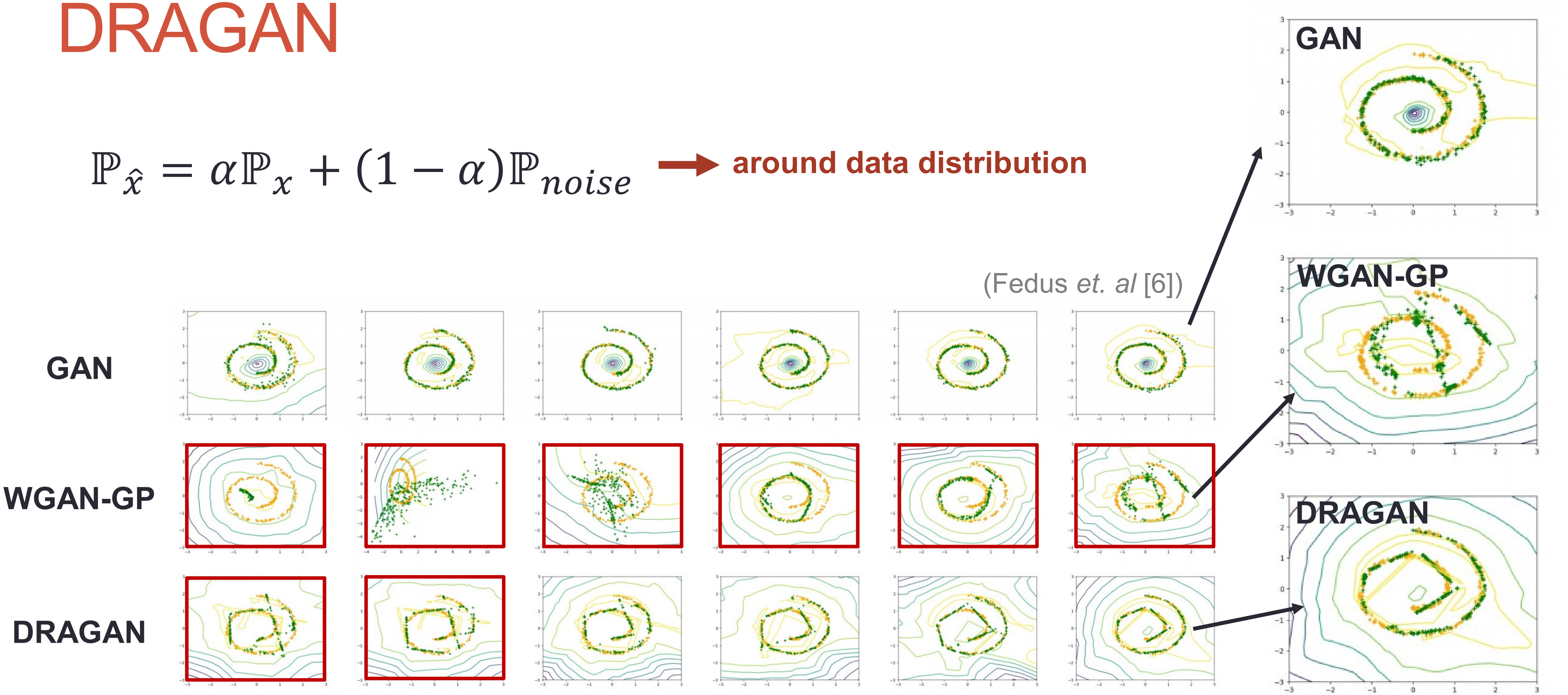
with gradient penalty

gradient vanishing

output **input**

DRAGAN

$$\mathbb{P}_{\hat{x}} = \alpha \mathbb{P}_x + (1-\alpha)\mathbb{P}_{noise}$$ → **around data distribution**

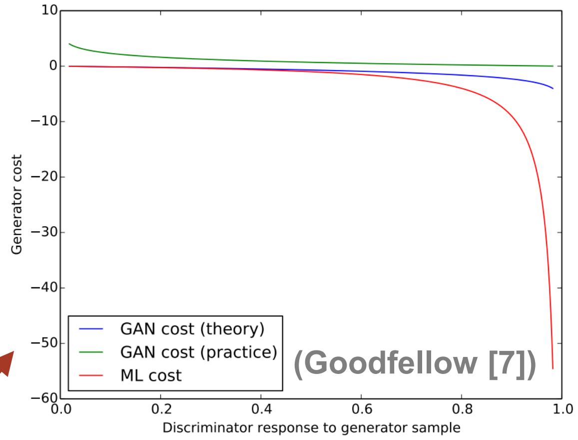


(Fedus *et. al* [6])

What is critical in GAN training?

Why is WGAN more stable?

theoretically, only **minimax GAN** may suffer from **gradient vanishing**



(Goodfellow [7])

the **properties of the underlying divergence** that is being optimized

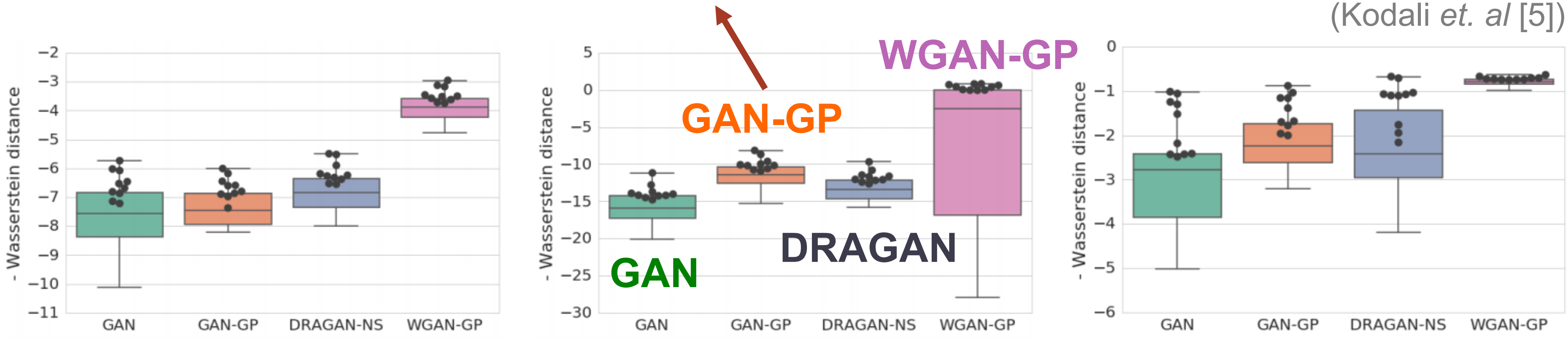
or

the **Lipschitz constraint**

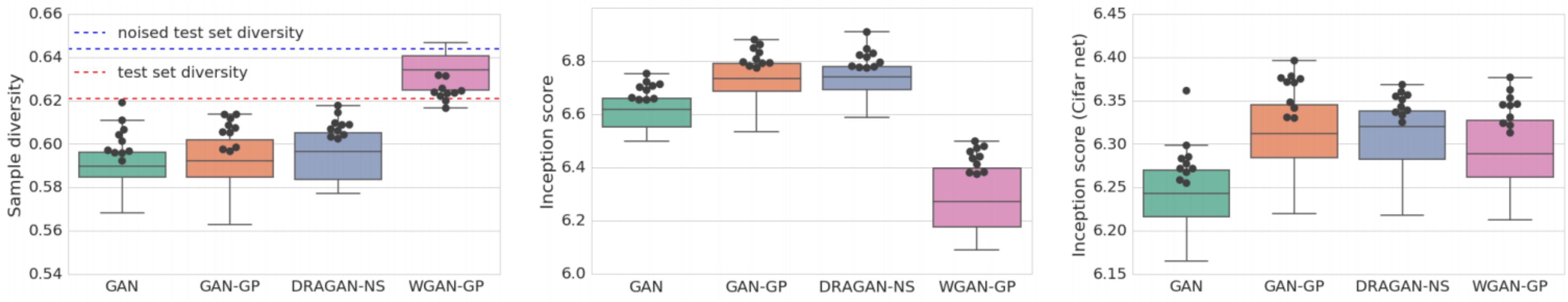
Comparisons

non-saturating GAN + gradient penalties

(Kodali *et. al* [5])



(a) Color MNIST (b) CelebA (c) CIFAR-10



(a) CelebA (b) Inception Score (ImageNet) (c) Inception Score (CIFAR)

Why is WGAN more stable?

properties of the underlying divergence that is being optimized

or

Lipschitz constraint **Why?**

(Recap) Original Convergence Proof

Proposition 1. *For G fixed, the optimal discriminator D is*

$$D_G^*(\boldsymbol{x}) = \frac{p_{data}(\boldsymbol{x})}{p_{data}(\boldsymbol{x}) + p_g(\boldsymbol{x})}$$

For a finite data set X, we only have

$$p_{data} = \begin{cases} 1, & x \in X \\ 0, & otherwise \end{cases}$$

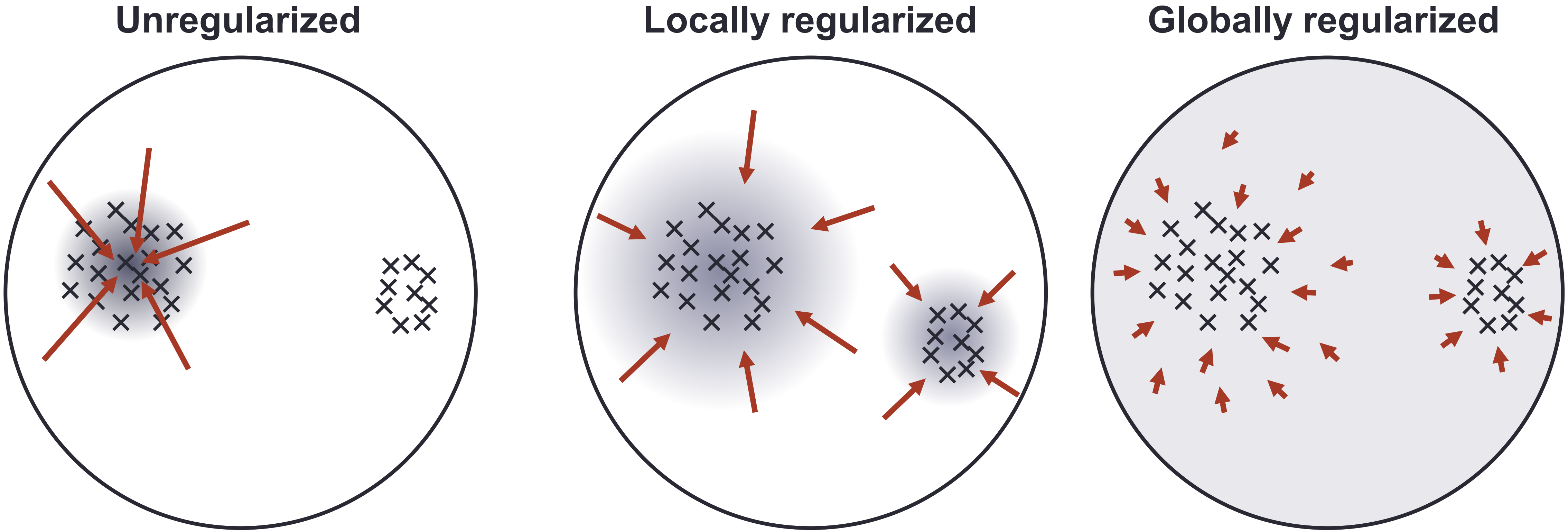
hard to optimize
(may need density estimation)

Proposition 2. *If G and D have enough capacity, and at each step of Algorithm 1, the discriminator is allowed to reach its optimum given G, and p_g is updated so as to improve the criterion*

$$\mathbb{E}_{\boldsymbol{x} \sim p_{data}}[\log D_G^*(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x} \sim p_g}[\log(1 - D_G^*(\boldsymbol{x}))]$$

then p_g converges to p_{data}

From a Distribution Estimation Viewpoint (my thoughts)



smoother critics

- give a more stable guidance to the generator
- alleviate mode collapse issue

Open Questions

- **Gradient penalties**
  - are usually too strong in WGAN-GP
  - may create spurious local optima
  - improved-improved-WGAN [8]
- **Spectral normalization**
  - may impact the optimization procedure?
  - can be used as a general regularization tool for any NN?

References

[1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "**Generative Adversarial Networks**," in *Proc. NIPS*, 2014.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou, "**Wasserstein Generative Adversarial Networks**," in *Proc. ICML*, 2017.

[3] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida, "**Spectral Normalization for Generative Adversarial Networks**," in *Proc. ICLR*, 2018.

[4] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville, "**Improved training of Wasserstein GANs**," In *Proc. NIPS*, 2017.

[5] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira, "**On Convergence and Stability of GANs**," *arXiv preprint arXiv:1705.07215*, 2017.

[6] William Fedus, Mihaela Rosca, Balaji Lakshminarayanan, Andrew M. Dai, Shakir Mohamed, and Ian Goodfellow, "**Many Paths to Equilibrium: GANs Do Not Need to Decrease a Divergence At Every Step**," *in Proc. ICLR*, 2017.

[7] Ian J. Goodfellow, "**On distinguishability criteria for estimating generative models**," in *Proc. ICLR*, *Workshop Track*, 2015.

[8] Xiang Wei, Boqing Gong, Zixia Liu, Wei Lu, and Liqiang Wang, "**Improving the Improved Training of Wasserstein GANs: A Consistency Term and Its Dual Effect**," in *Proc. ICLR*, 2018.

Thank you for your attention!