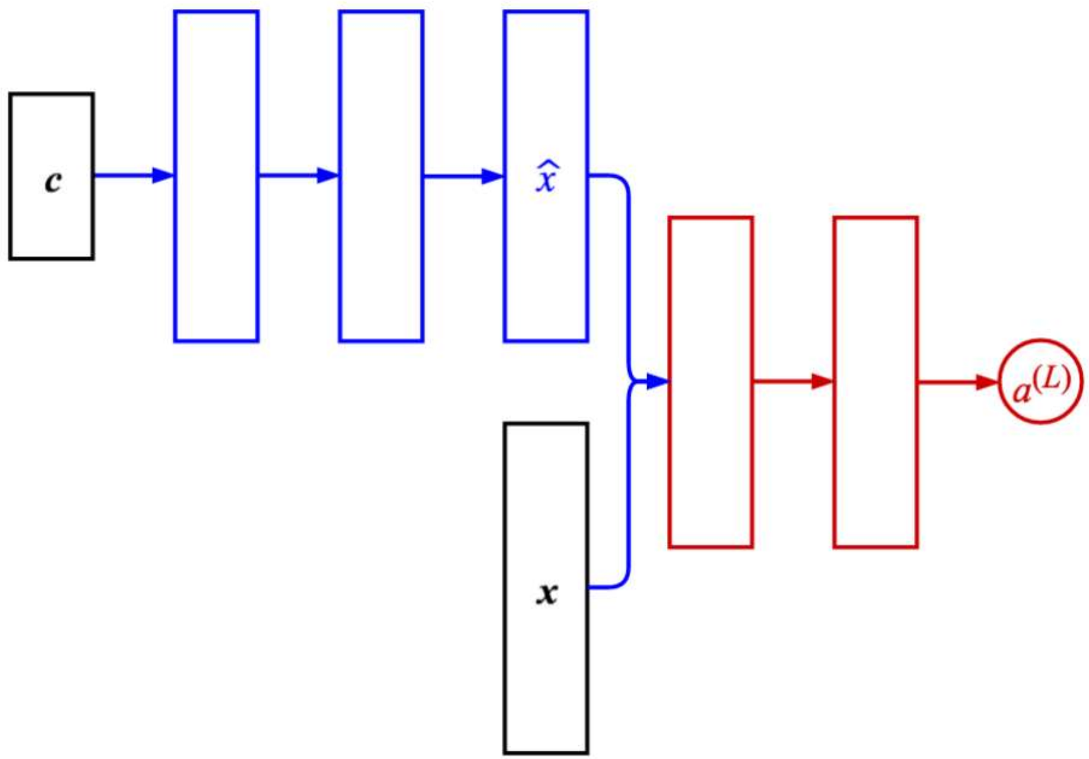
Deep Learning
Lab 14-2

GAN

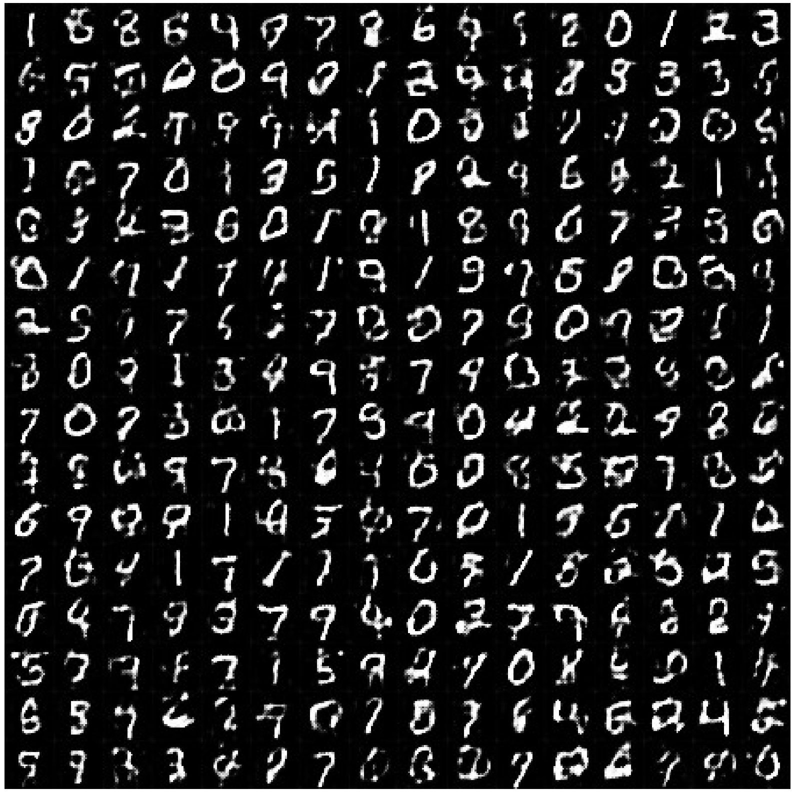
GAN



DCGAN

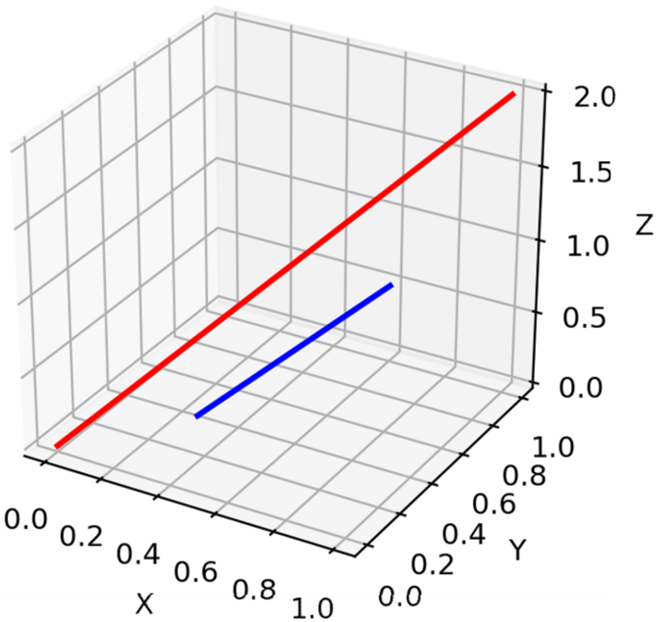
- **Architecture**
  - Replace pooling layers with strided convolutions for discriminator and fractional-strided convolutions for generator.
  - Use batch normalization in both the generator and the discriminator.
  - Use *ReLU* activation in generator for all layers except for the output, which uses *tanh* or *sigmoid*.
  - Use *LeakyReLU* activation in the discriminator for all layers.

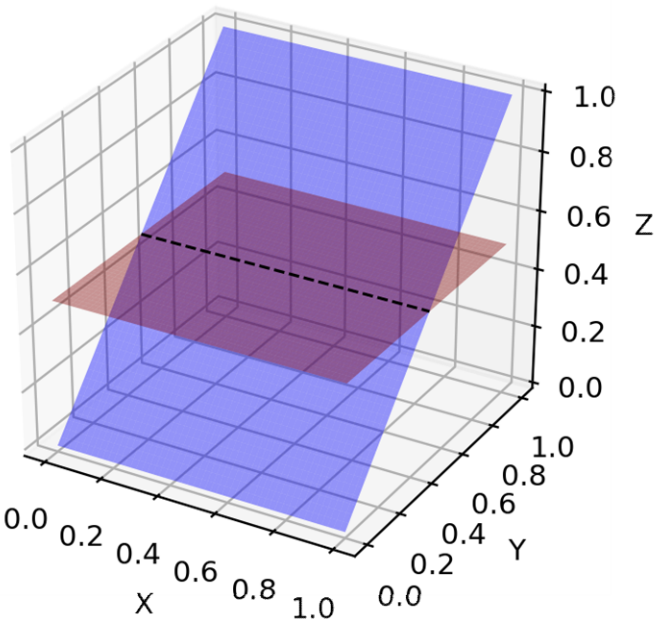
DCGAN



GAN

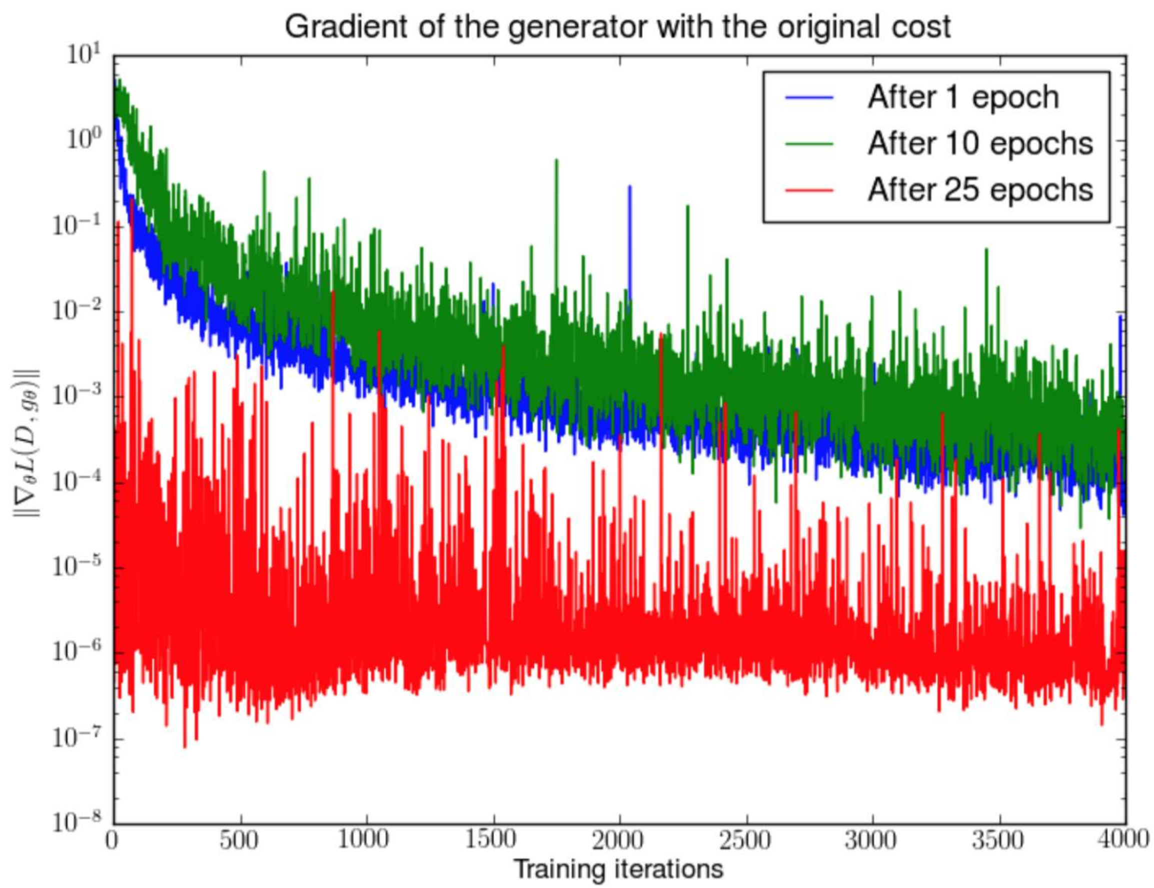
- **Loss**
  - Jensen–Shannon divergence
    - Constant when two distribution is not overlap.





GAN

- Loss
- **Problems**
  - Gradient vanishing



GAN

- **Loss**
- **Problems**
  - Gradient vanishing
  - Mode collapse

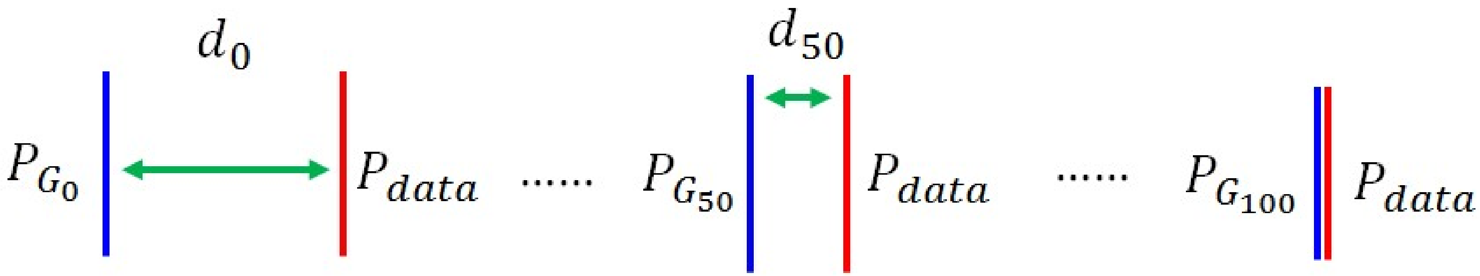
GAN

- **Loss**
- **Problems**
  - Gradient vanishing
  - Mode collapse
  - Highly-sensitive to hyper-parameters

WGAN

- **Loss**
  - Wasserstein Distance:

$$W_p(\mu,\nu) := \left(\inf_{\gamma \in \Gamma(\mu,\nu)} \int_{M \times M} d(x,y)^p \, \mathrm{d}\gamma(x,y) \right)^{1/p}$$



WGAN

- **Loss**
  - Wasserstein Distance (p = 1):

$$W(P_{\mathsf{data}}, P_g) = \inf_{Q \in \Gamma(P_{\mathsf{data}}, P_g)} E_{(\mathbf{x}, \hat{\mathbf{x}}) \sim Q}[||\mathbf{x} - \hat{\mathbf{x}}||]$$

  - In case **D is 1-Lipchitz**, then

$$\begin{aligned} W(P_{\mathsf{data}}, P_g) &= \sup_f E_{\mathbf{x} \sim P_{\mathsf{data}}}[f(\mathbf{x})] - E_{\mathbf{x} \sim P_g}[f(\mathbf{x})] \\ &= \sup_f \int_{\boldsymbol{x}} (P_{\mathsf{data}}(\boldsymbol{x}) - P_g(\boldsymbol{x})) f(\boldsymbol{x}) d\boldsymbol{x} \end{aligned}$$

WGAN

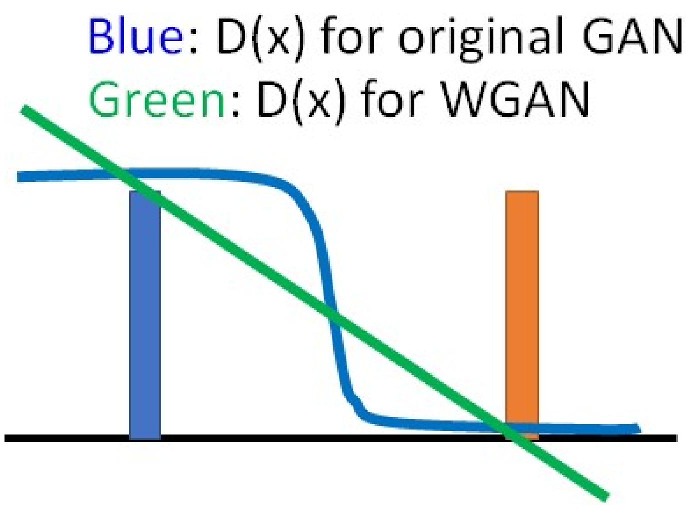
- **Loss**
  - Discriminator Training:

Sample $\{x^{(i)}\}_{i=1}^{m} \sim \mathbb{P}_r$ a batch from the real data.
Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.
$\boxed{g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^{m} f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))\right]}$
$w \leftarrow w + \alpha \cdot \boxed{\mathrm{RMSProp}}(w, g_w)$
$w \leftarrow \mathrm{clip}(w, -c, c)$

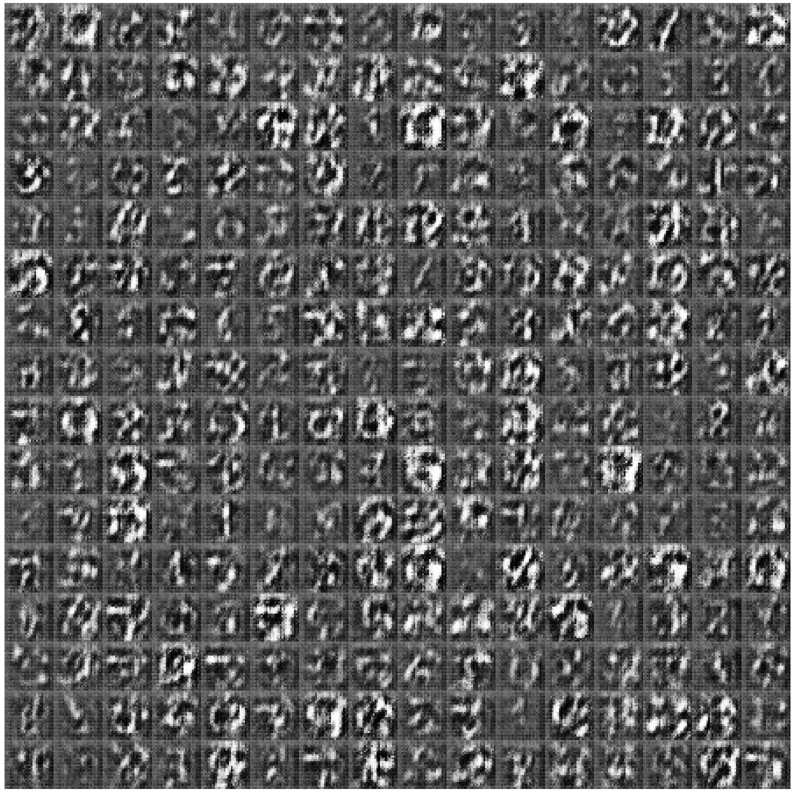
WGAN

- Loss

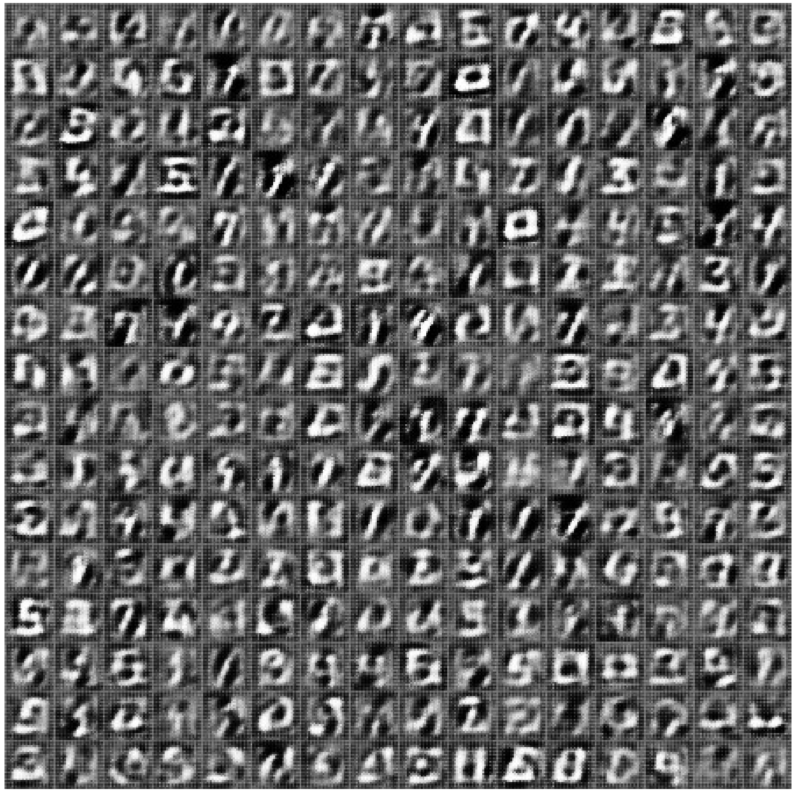
- **Architecture**
  - Do not apply *sigmoid* function to the last layer for the discriminator.



WGAN

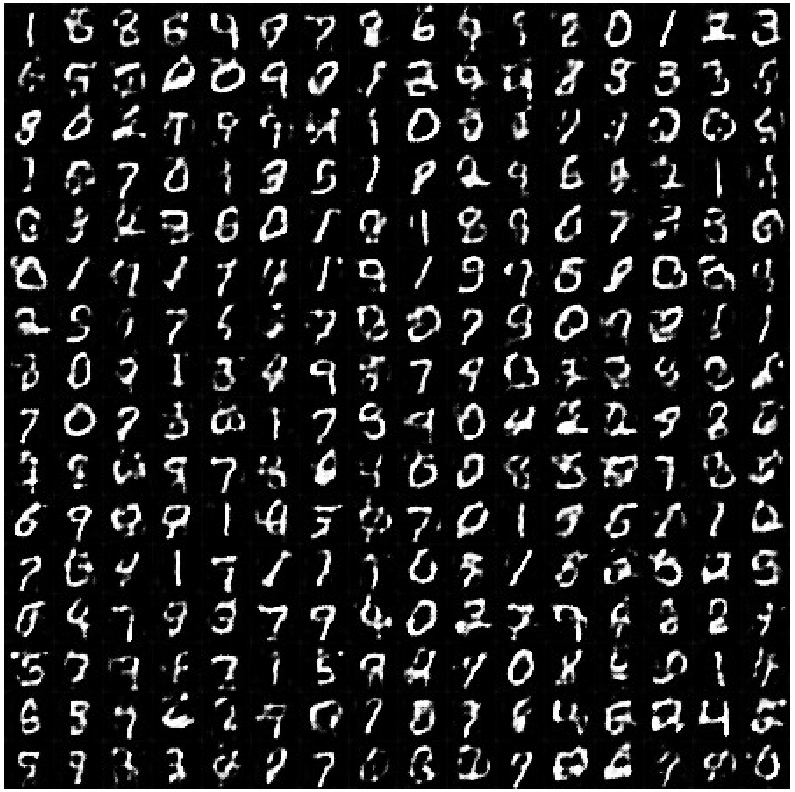


DCGAN

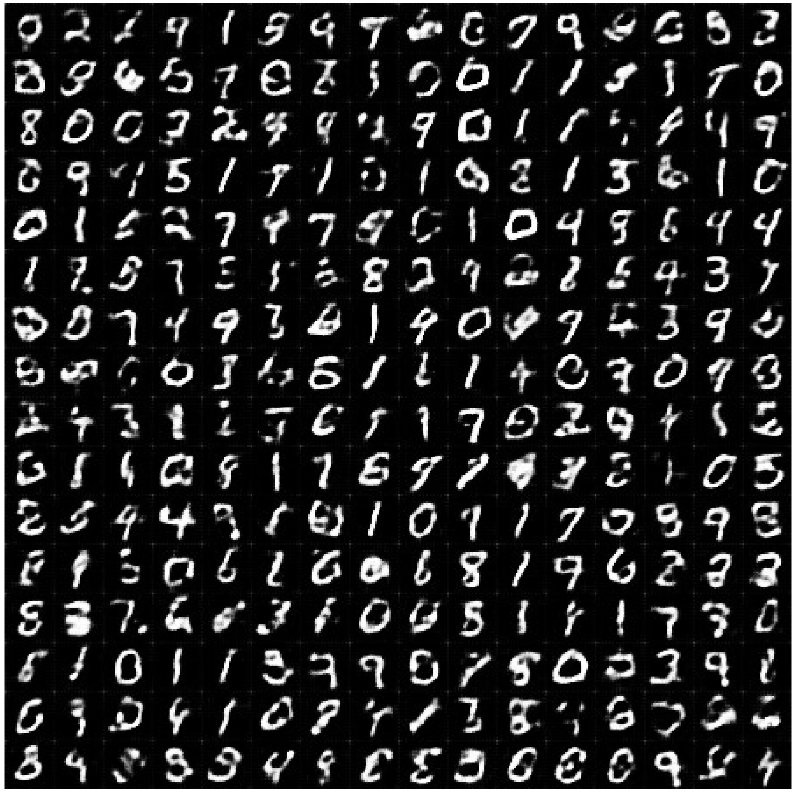


WGAN

WGAN



DCGAN



WGAN

WGAN

- **Weight Clipping**
  - Discriminator Training:

Sample $\{x^{(i)}\}_{i=1}^{m} \sim \mathbb{P}_r$ a batch from the real data.

Sample $\{z^{(i)}\}_{i=1}^{m} \sim p(z)$ a batch of prior samples.

$g_w \leftarrow \nabla_w \left[\frac{1}{m}\sum_{i=1}^{m} f_w(x^{(i)}) - \frac{1}{m}\sum_{i=1}^{m} f_w(g_\theta(z^{(i)}))\right]$

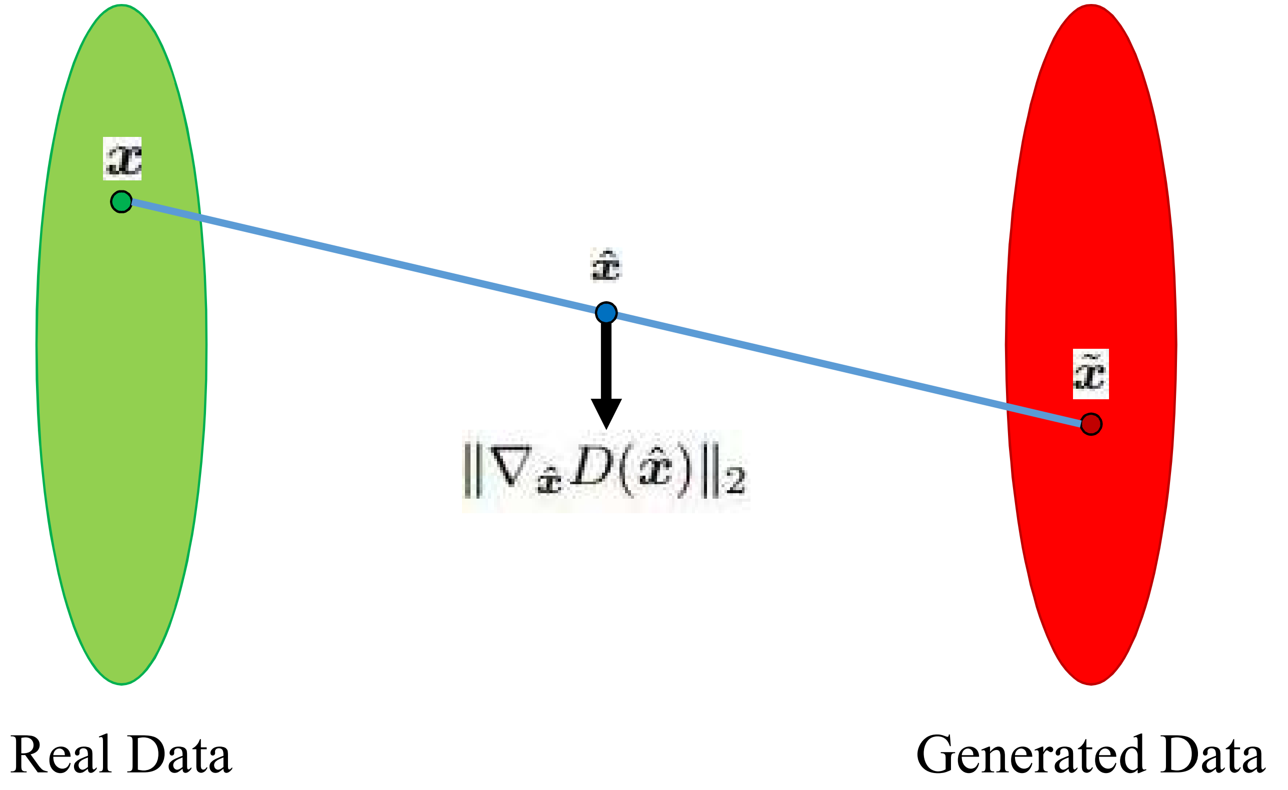
$w \leftarrow w + \alpha \cdot \mathrm{RMSProp}(w, g_w)$

$\boxed{w \leftarrow \mathrm{clip}(w, -c, c)}$

Make sure that **D is 1-Lipchitz**

Improved WGAN

- **Gradient penalty**



Improved WGAN

- **Gradient penalty**

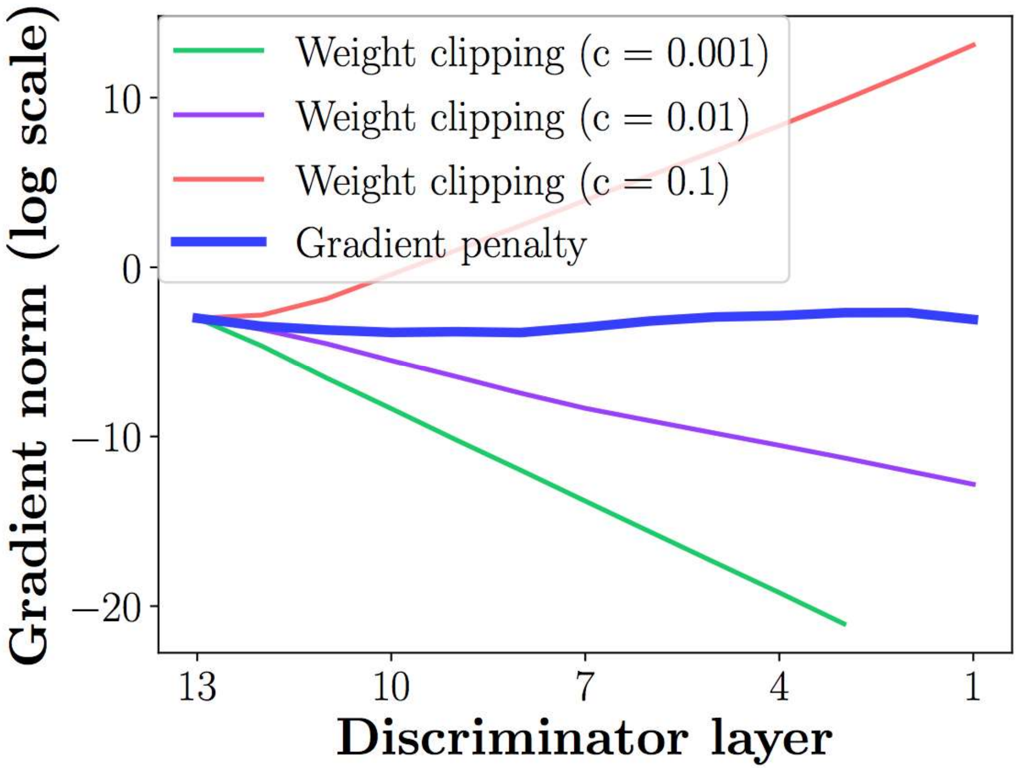
$$L = \boxed{\underset{\tilde{x}\sim\mathbb{P}_g}{\mathbb{E}}[D(\tilde{x})] - \underset{x\sim\mathbb{P}_r}{\mathbb{E}}[D(x)]} + \boxed{\lambda \underset{\hat{x}\sim\mathbb{P}_{\hat{x}}}{\mathbb{E}}\left[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2\right]}$$

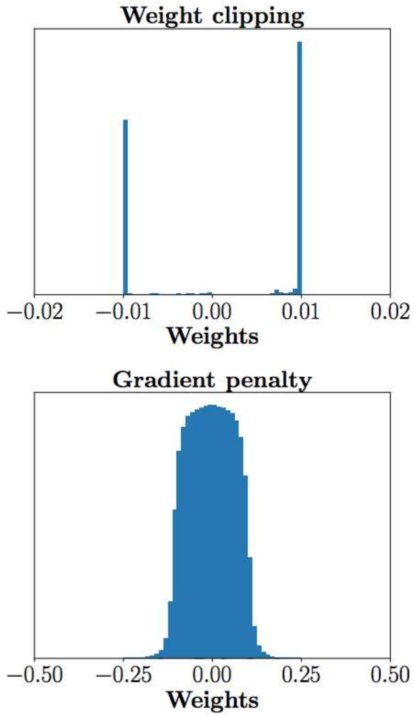
Original Discriminator Loss

Gradient penalty

Improved WGAN

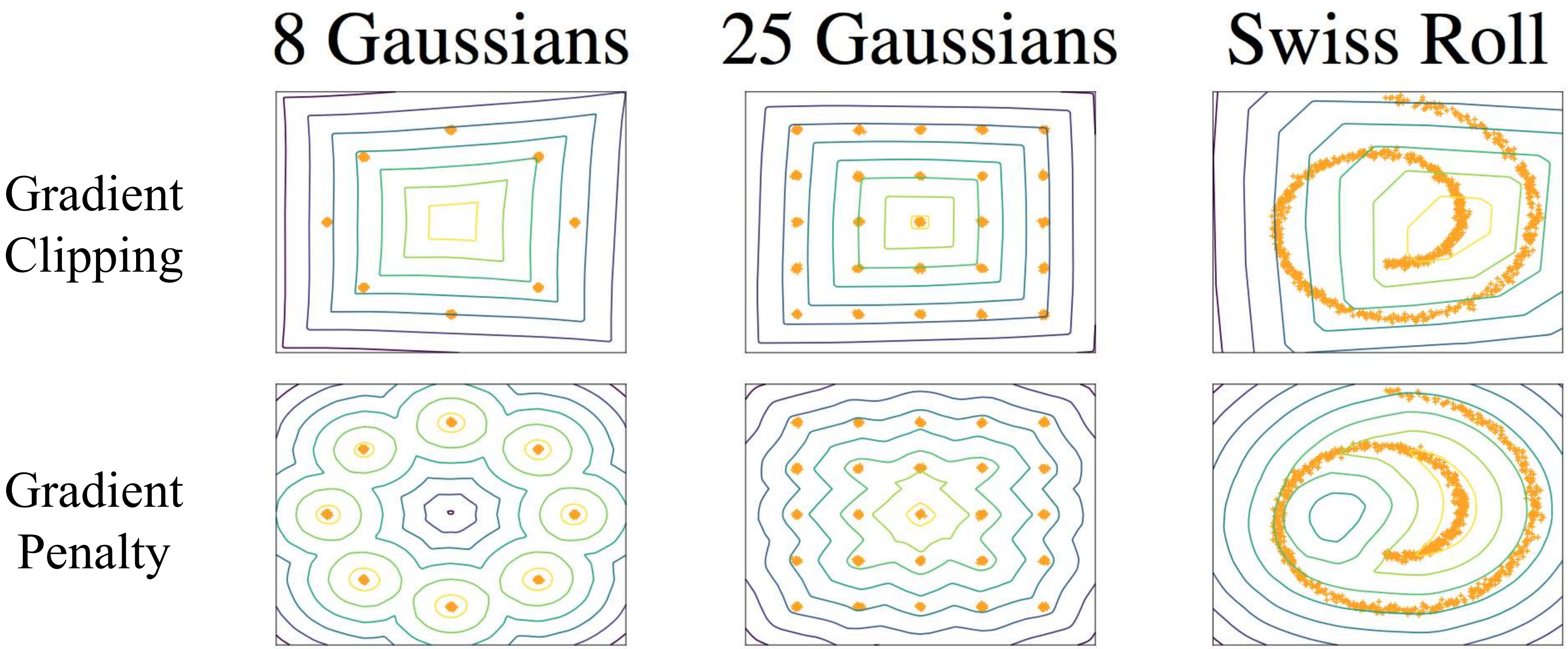
- **Gradient penalty**



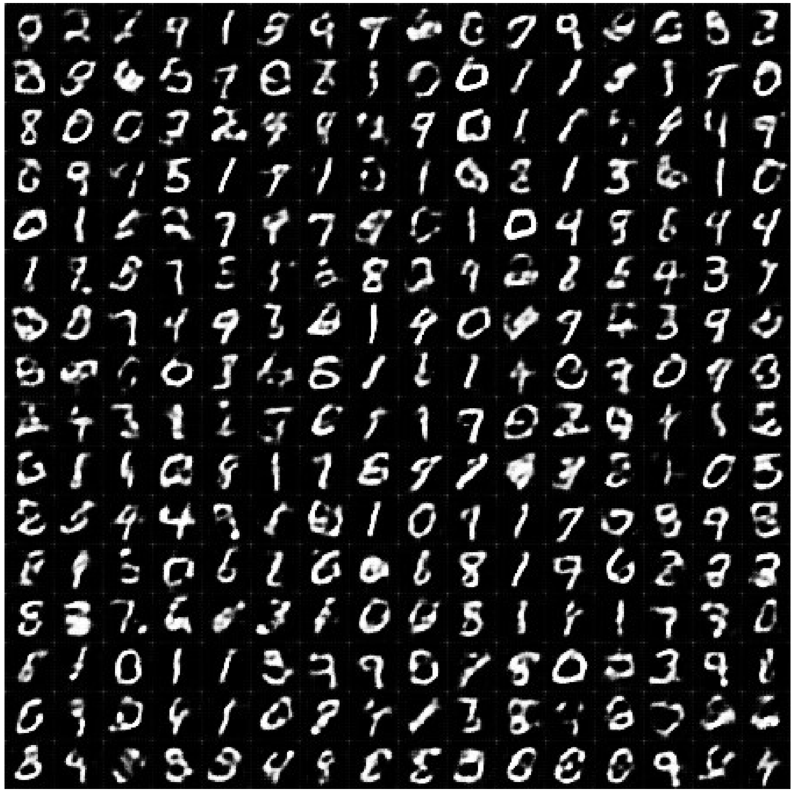


Improved WGAN

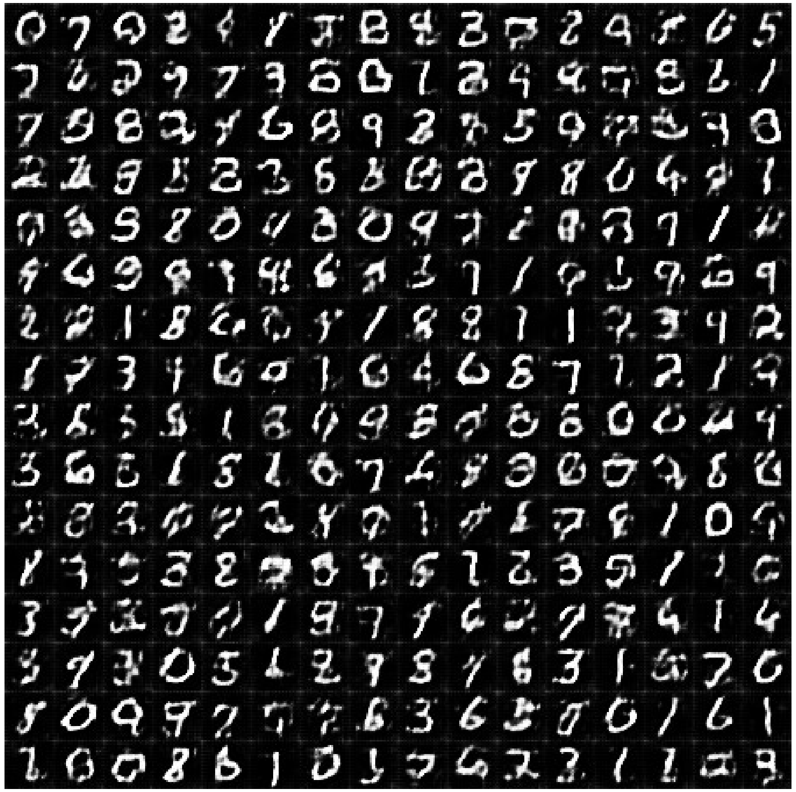
- **Gradient penalty**
  - Discriminator capacity:



Improved WGAN

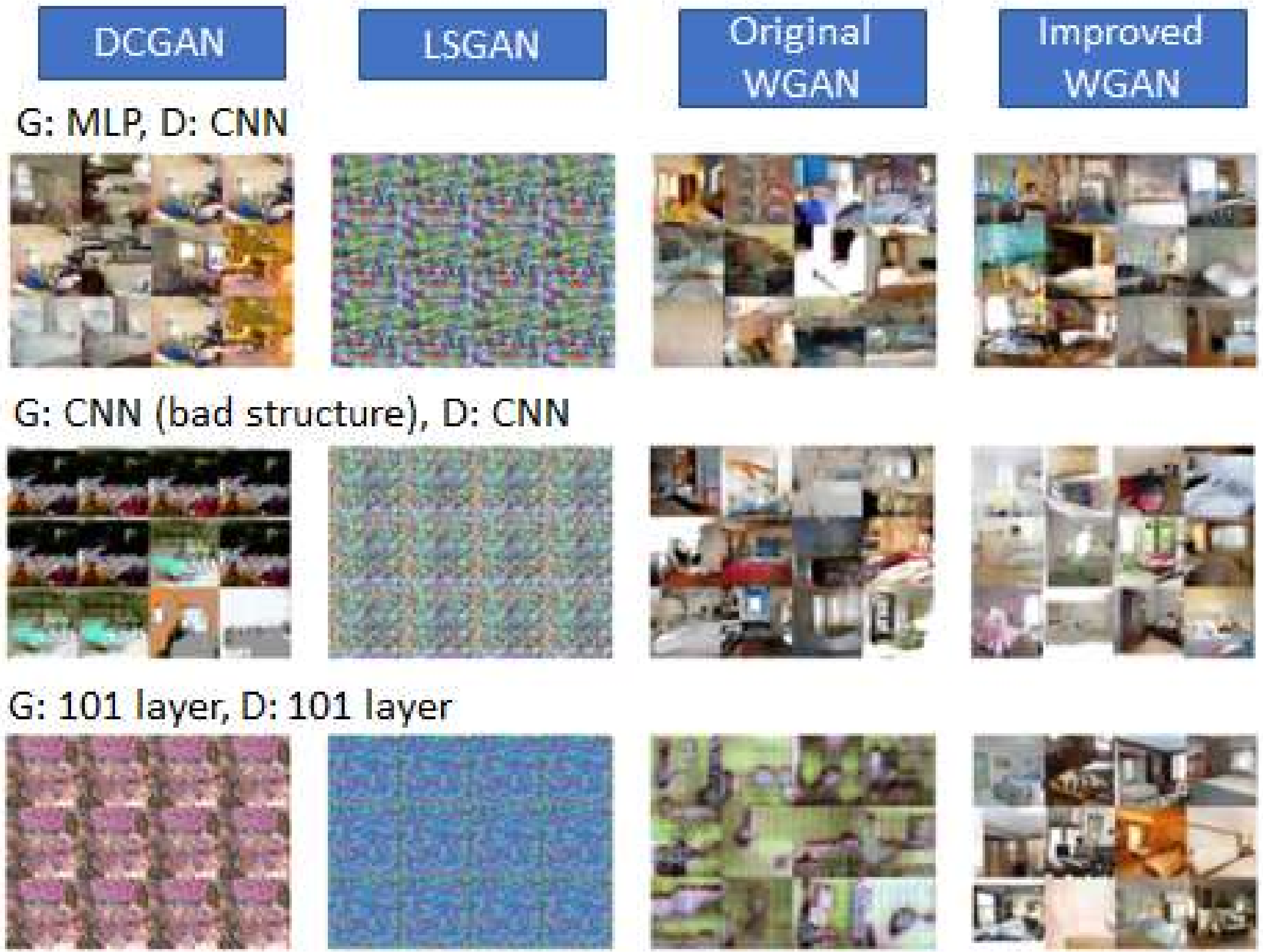


WGAN



WGAN-GP

Improved WGAN



References

- https://lilianweng.github.io/lil-log/2017/08/20/from-GAN-to-WGAN.html
- https://zhuanlan.zhihu.com/p/25071913
- https://medium.com/@jonathan_hui/gan-wasserstein-gan-wgan-gp-6a1a2aa1b490
- https://zhuanlan.zhihu.com/p/52799555

Assignment

- **Loss**
  - Improved WGAN (WGAN-GP)

- **Dataset**
  - CelebA (aligned)

Assignment

- **Examples**

Assignment

- **Requirements**
  - Implementation of Improved WGAN (WGAN-GP).
  - Build dataset to read and resize image to $\mathbf{64 \times 64}$ for training.
  - Training loop(s) / routine(s) for GAN. Pre-trained models are not allowed.
  - Show at least $\mathbf{8 \times 8}$ animated image of training and some best generated samples.
  - Draw the curve of discriminator loss and generator loss during training process in a single image.
  - Brief report about what you have done.

Assignment

- **Requirements**

- **Submission**
  - Upload notebook and attachments to google drive and submit the link to iLMS.
  - Your notebook should be named after "**Lab14-2_{student id}.ipynb**".
  - Deadline : **2019/12/12 23:59**