Improved Training of Wasserstein GANs

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, Aaron Courville

Outline

- Introduction
- Wasserstein GAN
- Gradient Penalty
- Experiments
- Conclusion

Introduction

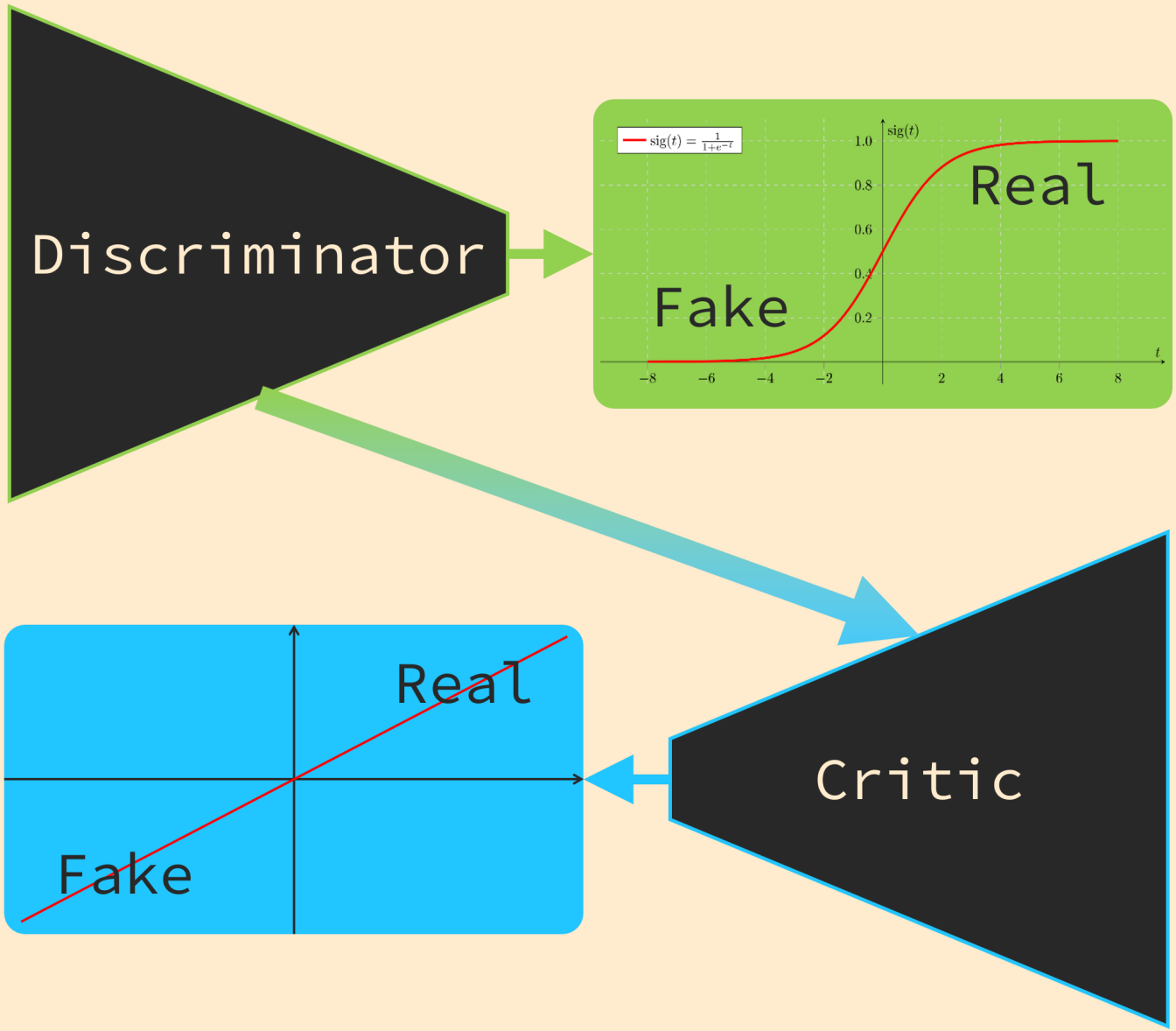
Although GAN is a powerful generative model, its training is unstable due to the gradient vanishing.

WGAN uses some simple changes to greatly reduce the gradient vanishing problem of the original GAN.

However, the weight clipping used in WGAN may still cause the gradient to vanish or explode.

For this reason, this paper uses the "gradient penalty" loss function to constrain the weights so that WGAN can converge stably.

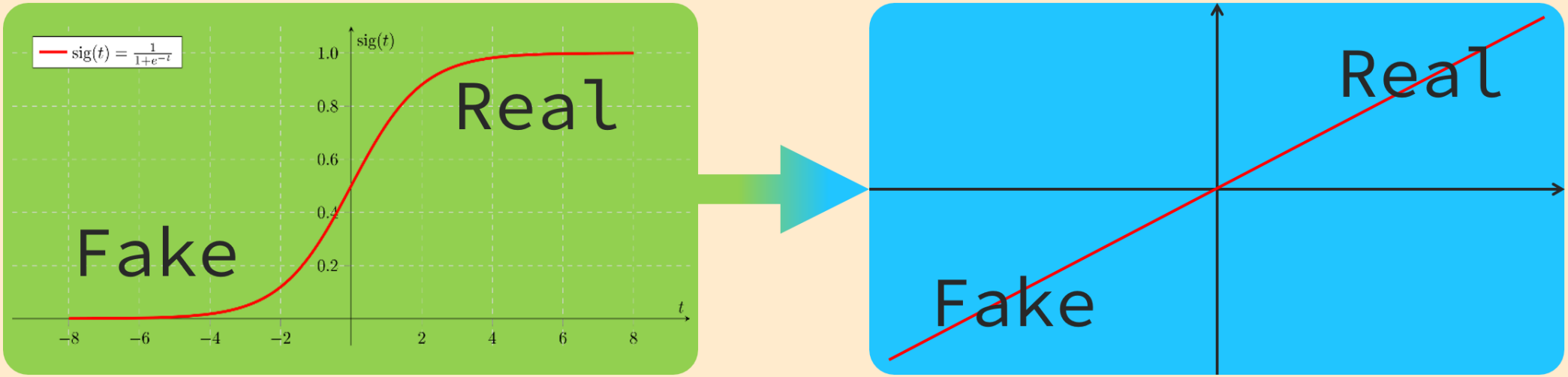
Wasserstein GAN



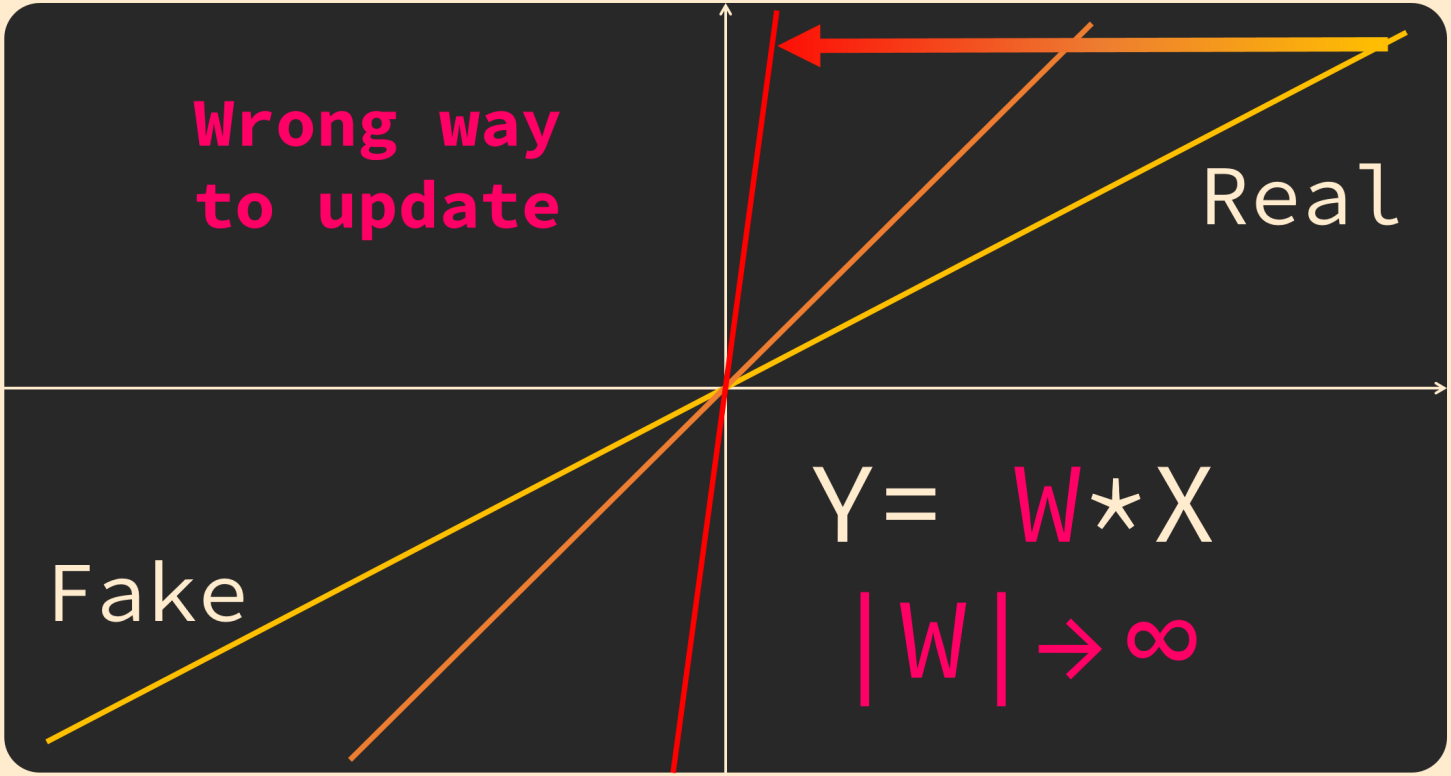
Discriminator => Critic

Solve the problem of vanishing gradient

1. Remove the Sigmoid in the output.
2. Remove the log in loss.

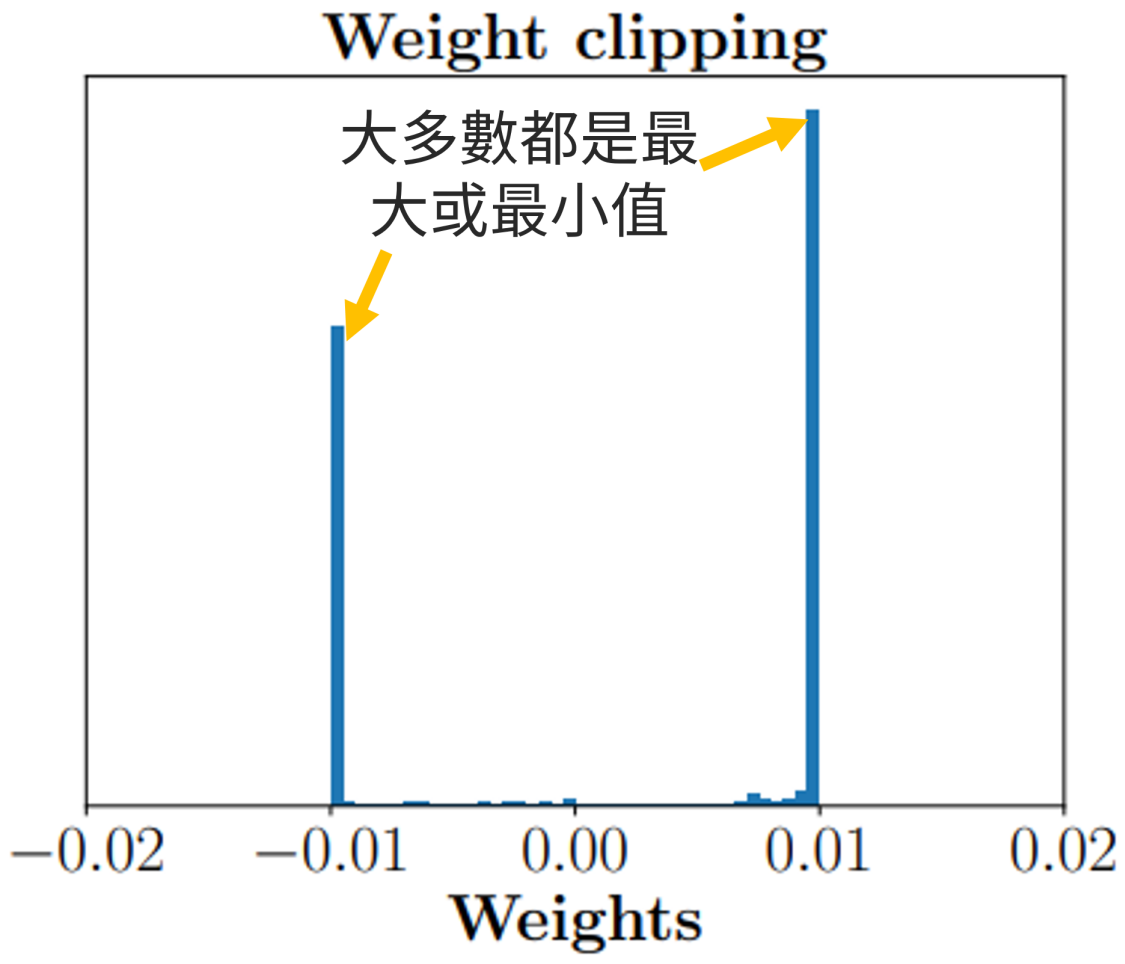


Discriminator => Critic



Without restrictions, it may lead to unlimited weight growth.

Weight Clipping



Use Weight Clipping in the original WGAN paper to limit the weight.

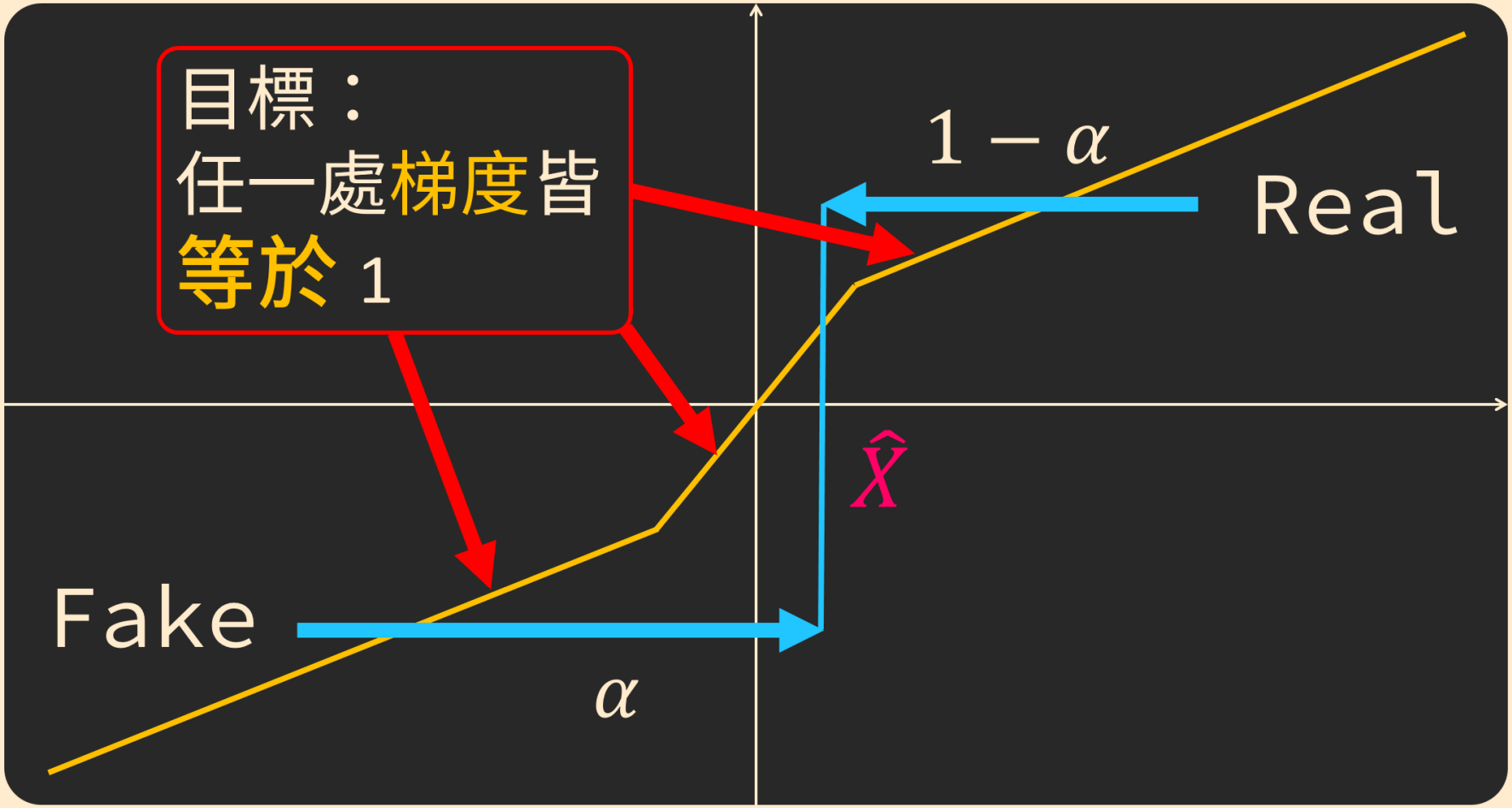
But it will cause the gradient to vanish or explode.

Gradient Penalty

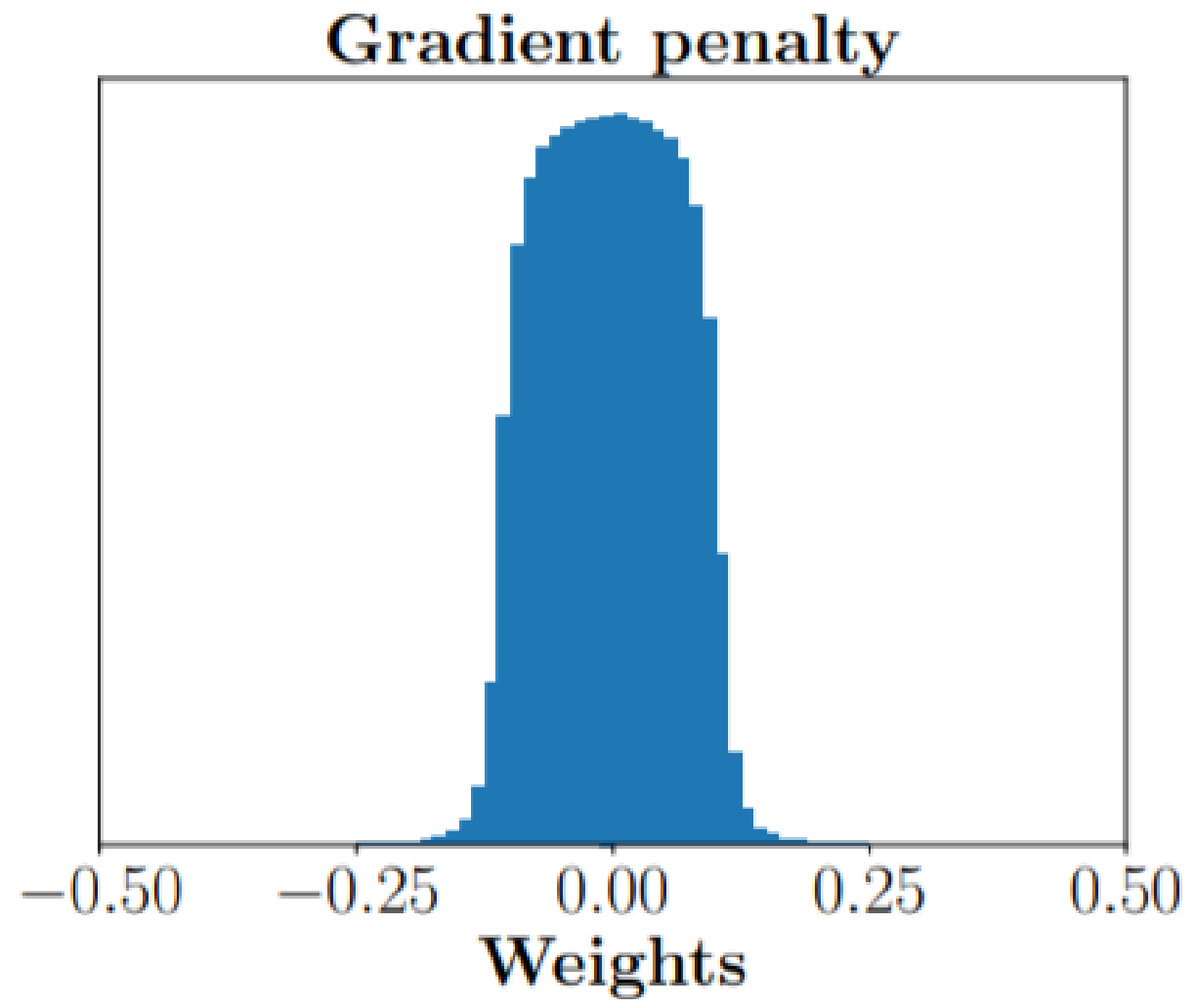
$$\hat{X} = (1 - \alpha)X + \alpha\tilde{X}$$

$$(\|\nabla_{\hat{x}} Critic(\hat{x})\|_2 - 1)^2$$

Gradient Penalty

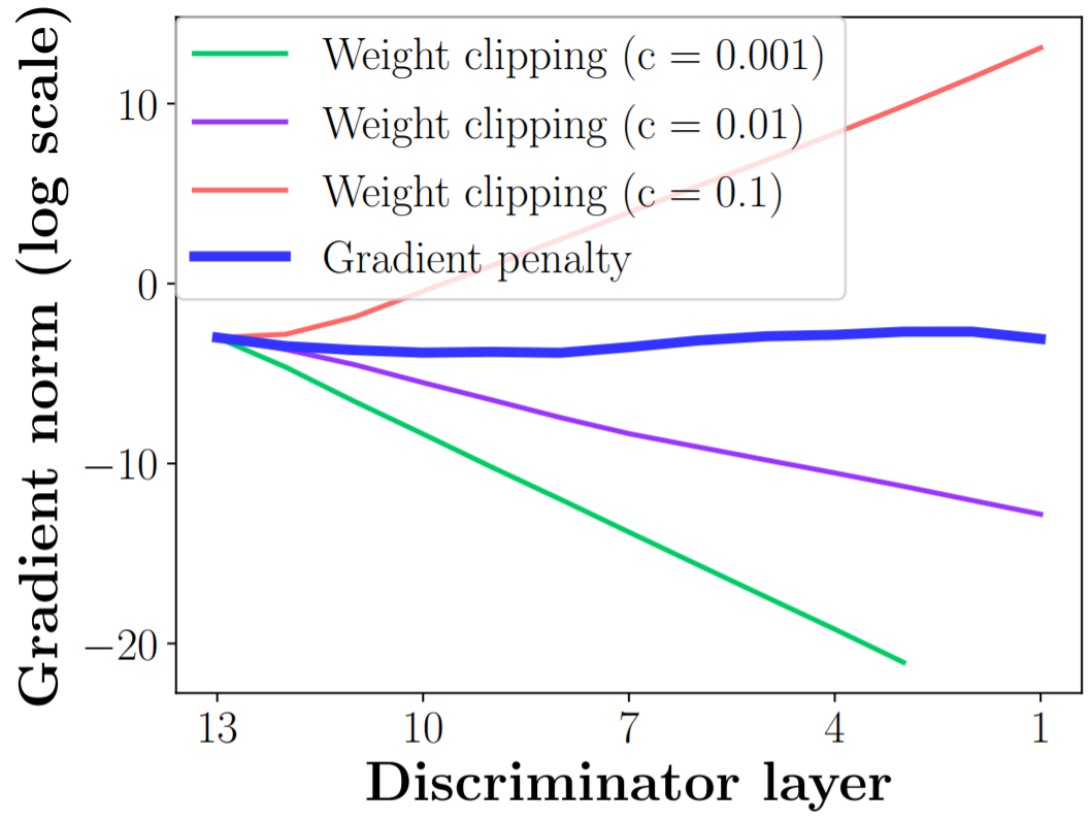


Gradient Penalty



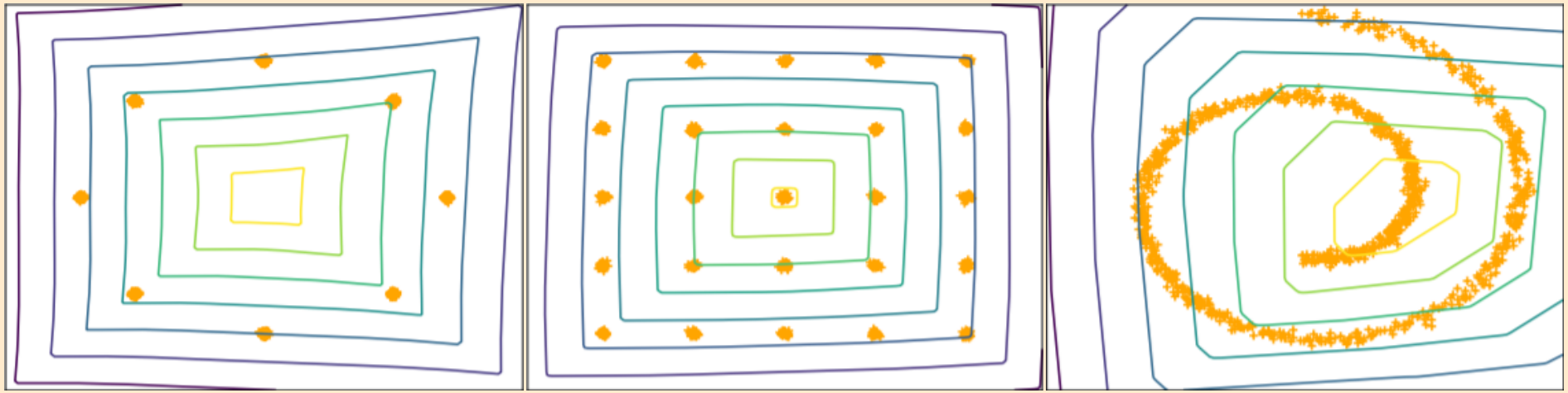
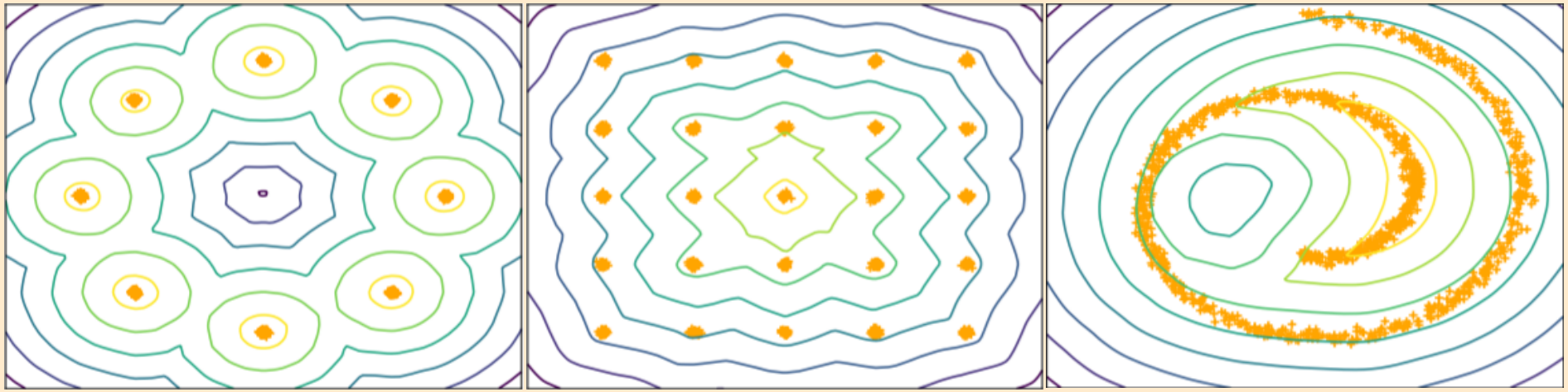
The weights are evenly distributed around 0, which makes the model more robust.

Gradient Penalty



Even in a deep model, Gradient Penalty can keep the gradient stable without vanishing or exploding like Weight Clipping.

Gradient Penalty

	8 Gaussians	25 Gaussians	Swiss Roll
WC			
GP			

Loss

Generate Loss

$$-Critic(G(z))$$

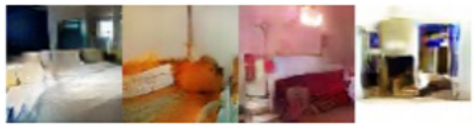
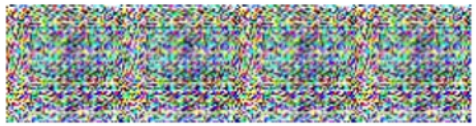
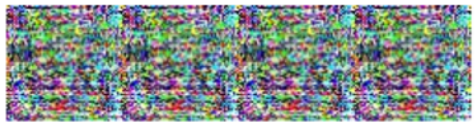
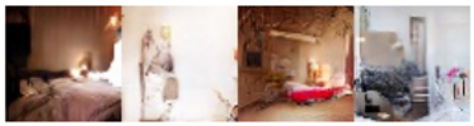
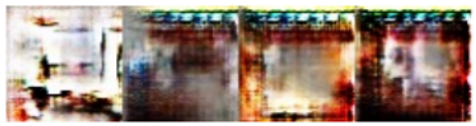
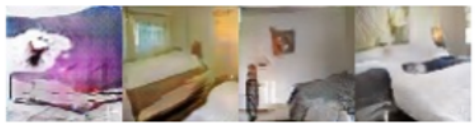
越大越好

Critic Loss

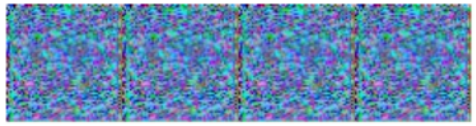
越小越好　　越大越好

$$Critic(G(z)) - Critic(x) + \lambda(\|\nabla_{\hat{x}} Critic(\hat{x})\|_2 - 1)^2$$

Robustness

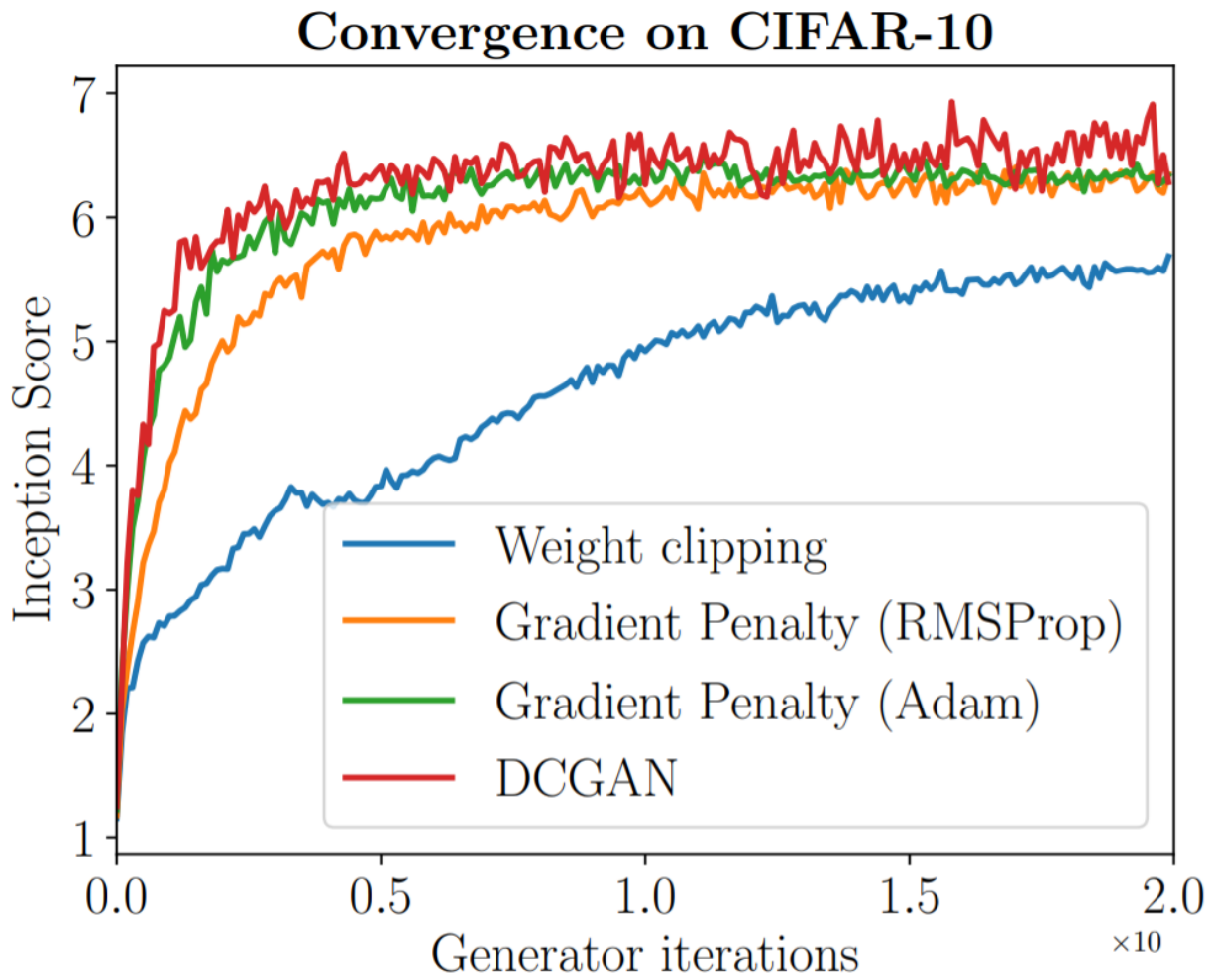
DCGAN	LSGAN	WGAN (clipping)	WGAN-GP (ours)
Baseline (G: DCGAN, D: DCGAN)				
G: No BN and a constant number of filters, D: DCGAN				
G: 4-layer 512-dim ReLU MLP, D: DCGAN				
No normalization in either G or D				

Robustness

DCGAN	LSGAN	WGAN (clipping)	WGAN-GP (ours)
Gated multiplicative nonlinearities everywhere in G and D			
			
tanh nonlinearities everywhere in G and D			
			
101-layer ResNet G and D			
			

Use Gradient Penalty to apply WGAN to each model architecture.

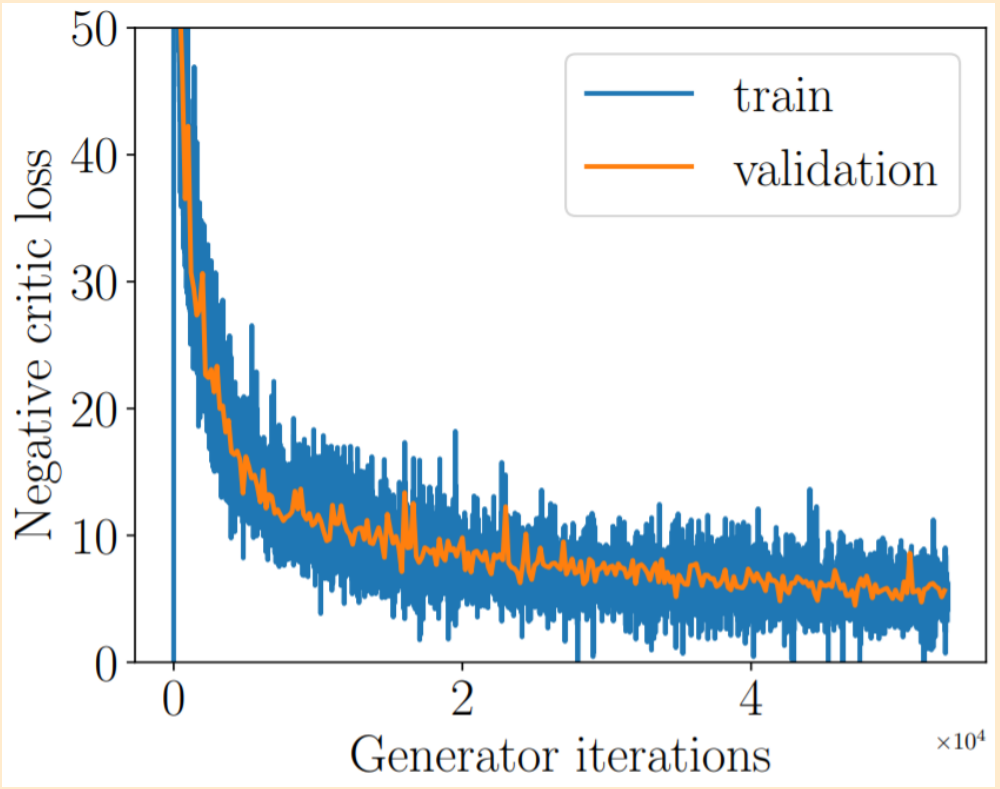
Inception Score



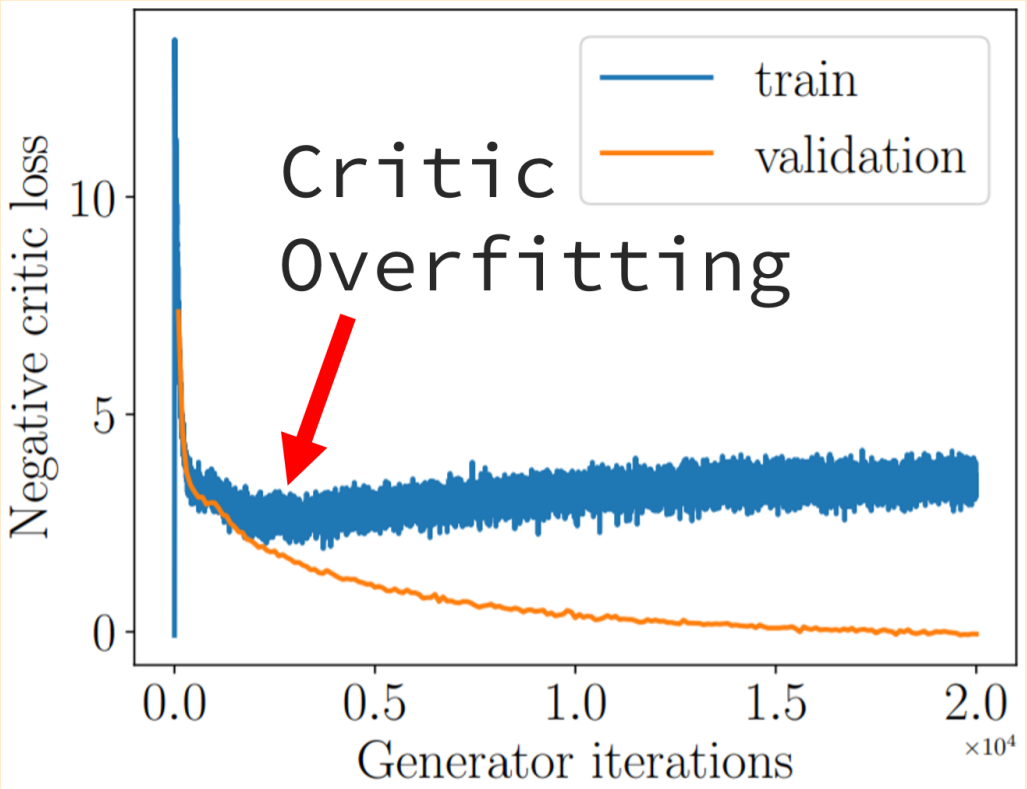
It can reach a level equivalent to DCGCN, and the training convergence is not limited to a certain model architecture.

Overfitting

Non Overfitting



Overfitting



When using WGAN-GP, if there are too few real samples, Critic will overfitting faster than Generator.

Conclusion

- WGAN uses simple techniques to solve the gradient vanishing problem caused by the discriminator.
- The Gradient Penalty brings higher robustness to the model, allowing WGAN training to converge without being affected by the model architecture.

Note

- In WGAN-GP, all Batch Norm are replaced with Layer Norm.
- The paper mentioned that the effect of the two-way Gradient Penalty is better than the one-way Gradient Penalty (which only penalizes the gradient greater than 1).
- Non-linear non-smooth function (e.g. ELU if α != 1) cannot train WGAN-GP.