



# Wasserstein GAN

Xue Minglong

# CONTENT OF WEEK3

## 1 Wasserstein GAN 理论推导

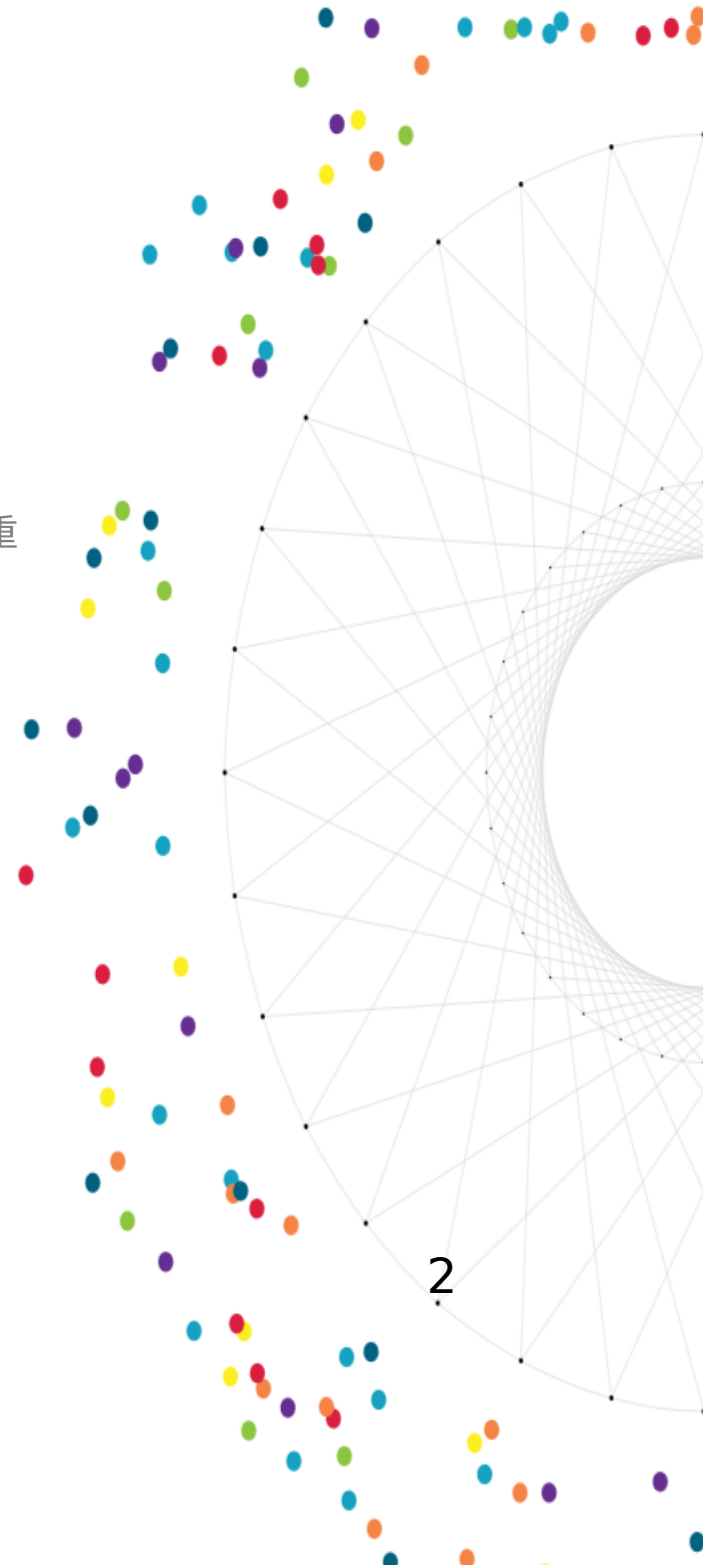
在上周提出的原始 GAN 模型出现的训练难、优化目标不合理、梯度不稳定三个问题的基础上，找到一种 WGAN 模型可以有效的解决上述问题。其主要思想是使用一种新的损失函数作为优化目标，使得两函数支撑集即使不重叠去，其目标函数也不会退化为常数。

## 2 搭建 WGAN 网络，进行对比实验

使用 tensorflow 搭建原始 GAN 模型与 WGAN，使用相同的网络结果，只改变其目标函数并进行对比实验便于观察实验结果。

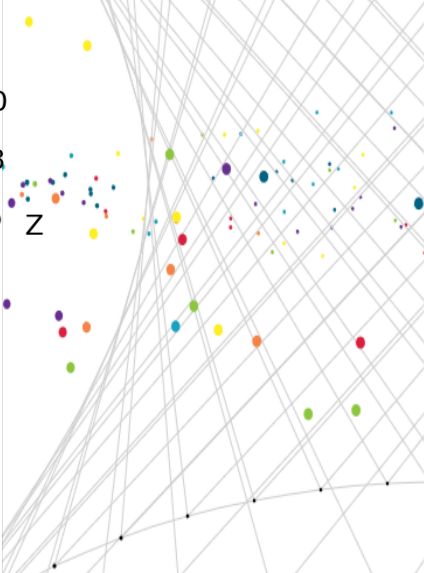
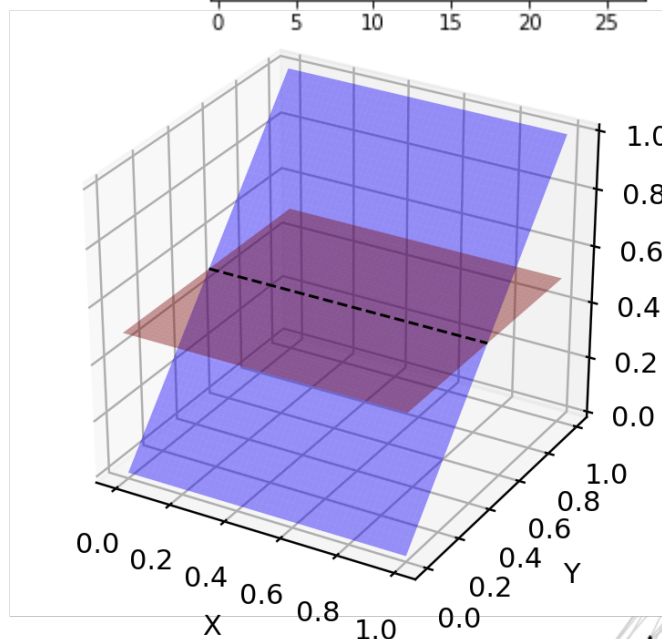
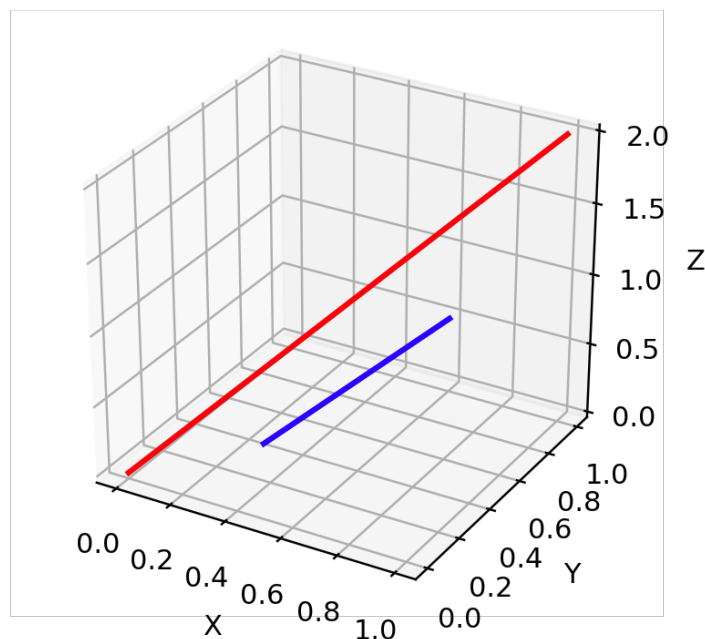
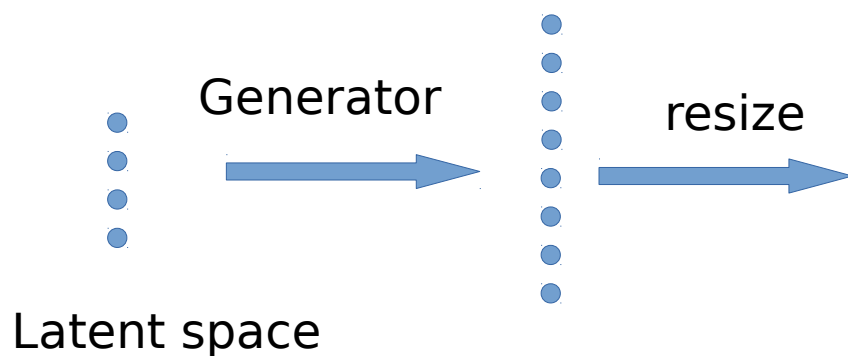
## 3 尝试在服务器上运行程序 ( 效果不理想 )

在服务器上使用 GPU 运行程序时，运行速度比在本地运行慢。其原因可能是因为没有正确使用 pipeline 策略，使得每次选择训练集中数据块进行梯度下降时，程序运行效率低，或者没有正确调用 tensorflow 的 GPU API。





# 问题 1 : Low dimension manifold



$$L(G) = 2JS(P_r||P_g) - 2\log 2$$

(1)

4

$JS(P_r||P_g) = 0$ ,  $L(G)$  退化为常数



使用 JS 距离作为生成图像与真实图像的分布差异指标不合理 !!!!!!!

## 生成模型常用的分布相似度指标

- The Kullback-Leibler (KL) divergence is

$$KL(P_r \| P_g) = \int_x \log \left( \frac{P_r(x)}{P_g(x)} \right) P_r(x) dx \quad (2)$$

This isn't symmetric. The reverse KL divergence is defined as  $KL(P_g \| P_r)$ .

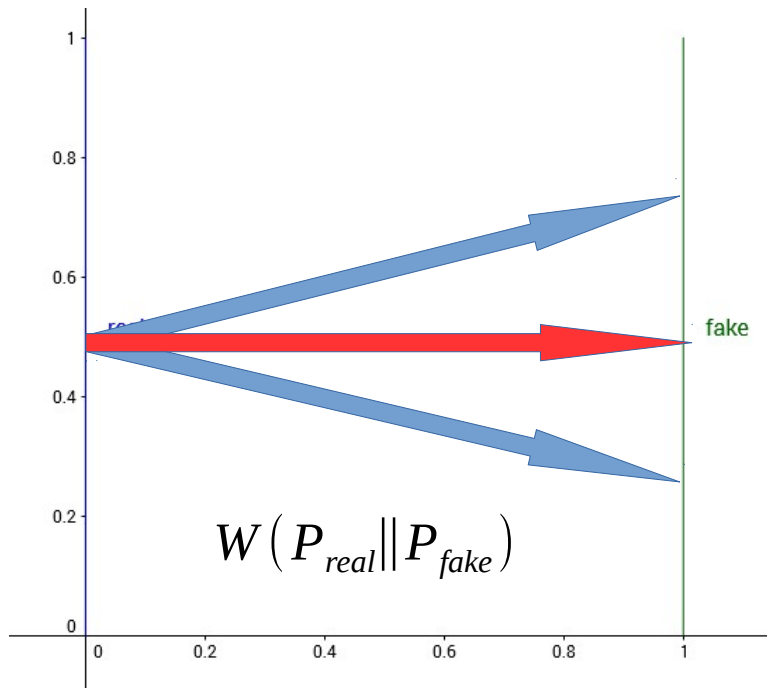
- The Jensen-Shannon (JS) divergence: Let  $M$  be the mixture distribution  $M = P_r/2 + P_g/2$ . Then

$$JS(P_r, P_g) = \frac{1}{2} KL(P_r \| P_m) + \frac{1}{2} KL(P_g \| P_m) \quad (3)$$

- Finally, the Earth Mover (EM) or Wasserstein distance: Let  $\Pi(P_r, P_g)$  be the set of all joint distributions  $\gamma$  whose marginal distributions are  $P_r$  and  $P_g$ . Then.

$$W(P_r, P_g) = \inf_{\gamma \in \Pi(P_r, P_g)} \mathbb{E}_{(x,y) \sim \gamma} [ \|x - y\| ] \quad (4)$$

# 举例分析



$$KL(P_{real} || P_{fake}) = \int P_{real} \log \frac{P_{real}}{P_{fake}} dx$$

$$\rightarrow +\infty (P_{real} \rightarrow 1 \wedge P_{fake} \rightarrow 0) \quad (5)$$

$$JS(P_{real} || P_{fake}) = \int P_{real} \log \frac{P_{real}}{1/2(P_{fake} + P_{real})} dx$$

$$+ \int P_{fake} \log \frac{P_{fake}}{1/2(P_{fake} + P_{real})} dx$$

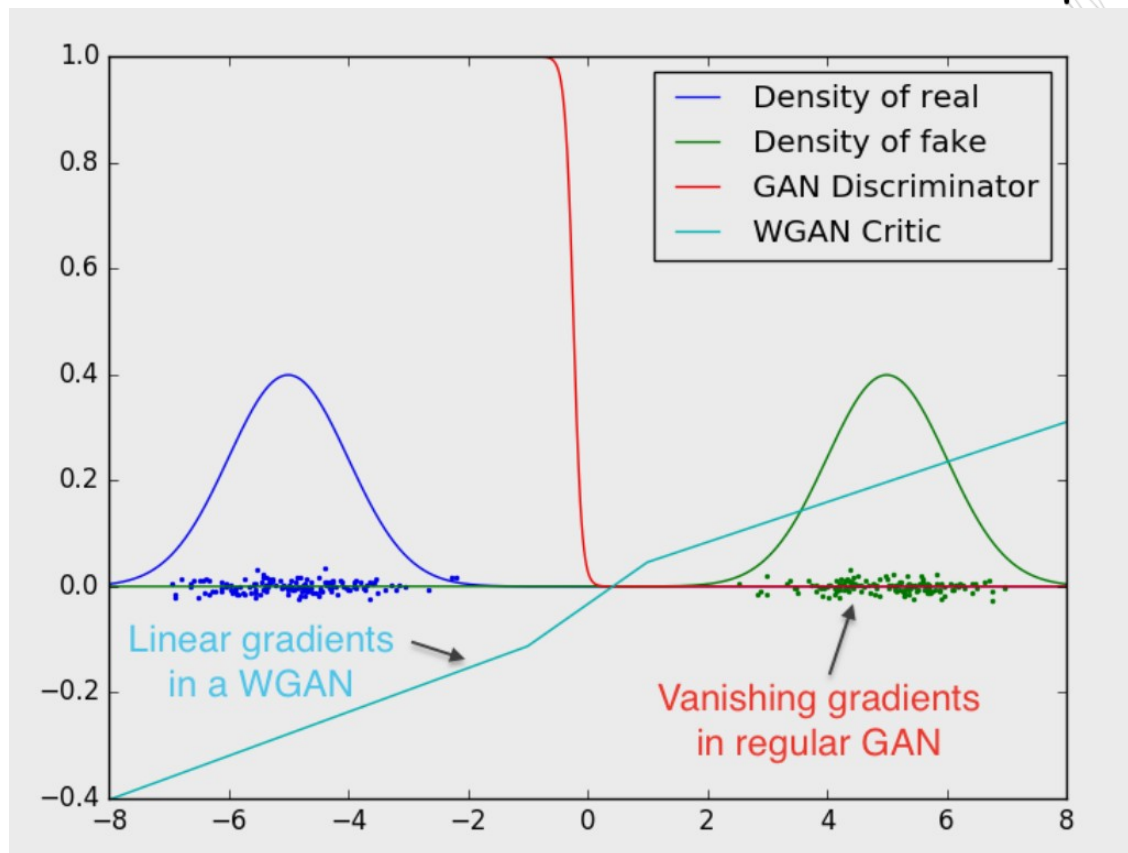
$$\rightarrow \log 2 \quad (6)$$

$$W(P_{real} || P_{fake}) = \inf_{\gamma \sim \Pi(P_r, P_g)} E_{(x,y) \sim \gamma} [\|(x-y)\|] \quad (7)$$

Consider probability distributions defined over  $\mathbb{R}^2$ . Let the true data distribution be  $(0, y)$ , with  $y$  sampled uniformly from  $U[0, 1]$ . Consider the family of distributions  $P_\theta$ , where  $P_\theta = (\theta, y)$ , with  $y$  also sampled from  $U[0, 1]$ .



# Wasserstein distance and JS divergence



解决了问题 1、2

- W 距离即使在两分布不重叠或重叠部分可忽略不计的条件下其导数为常数使得生成器可以正常进行训练
- JS 散度在两分布不重叠或重叠部分可忽略不计的条件下时退化为常数，其导数为 0，使得生成器无法继续训练

# WGAN 改动 1，使用 Wasserstein 距离作为评价生成图片与真实图片的相似度指标

$$W(P_{real} \| P_{fake}) = \inf_{\gamma \sim \prod(P_r, P_g)} E_{(x, y) \sim \gamma} [\|x - y\|] \quad (8)$$

Kantorovich-Rubinstein duality

$$W(P_{real} \| P_{fake}) = s \sup_{\|f\|_L \leq 1} E_{x \sim P_r}[f(x)] - E_{x \sim P_\theta}[f(x)] \quad (9)$$

根据 lipschitz 条件：

$$\|f(a) - f(b)\| \leq K \|a - b\| \quad (10)$$

$$K \cdot W(P_{real} \| P_{fake}) = s \sup_{\|f\|_L \leq K} E_{x \sim P_r}[f(x)] - E_{x \sim P_\theta}[f(x)] \quad (11)$$

我们最终可用带参函数来估计上述方程的上界：

$$\max_{w \in W} E_{x \sim P_r}[f_w(x)] - E_{x \sim P_\theta}[f_w(x)] \leq s \sup_{\|f\|_L \leq K} E_{x \sim P_r}[f(x)] - E_{x \sim P_\theta}[f(x)] \quad (12)$$

最终的生成器优化目标为：

$$\max_{w \in W} E_{x \sim P_r}[f_w(x)] - E_{x \sim P_\theta}[f_w(x)] \quad (13)^8$$



## WGAN 改动 2，使用 weights clamping 来使得神经网络满足 lipschitz 连续性要求

$$K \cdot W(P_{real} \| P_{fake}) = \sup_{\|f\|_L \leq K} E_{x \sim P_r}[f(x)] - E_{x \sim P_\theta}[f(x)] \quad (14)$$

the ‘critic’  $f_w$ . In order to have parameters  $w$  lie in a compact space, something simple we can do is clamp the weights to a fixed box (say  $\mathcal{W} = [-0.01, 0.01]^l$ ) after each gradient update. The Wasserstein Generative Adversarial Network (WGAN) procedure is described in Algorithm 1

具体做法是在每次判别器更新后对其模型参数进行裁剪使其限制在我们预设的范围内，此范围  $c$  是一个可调参数（hyper parameter）

$$\begin{aligned} g_w &\leftarrow \nabla_w \left[ \frac{1}{m} \sum_{i=1}^m f_w(x^{(i)}) - \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)})) \right] \\ w &\leftarrow w + \alpha \cdot \text{RMSProp}(w, g_w) \\ w &\leftarrow \text{clip}(w, -c, c) \end{aligned}$$

## WGAN 改动 3，判别器目的是增大生成图片与真实图片分布之间的距离，由做分类问题改为做回归问题

原始 GAN 的目标函数是由极大似然估计推导而来：

$$L(x) = E_{x \sim P_r} \log(D(x)) + E_{x \sim P_g} \log(1 - D(x)) \quad (15)$$

WGAN 的目标函数是由 wasserstein 距离所确定的：

$$L(x) = \max_{w \in W} E_{x \sim P_r} [f_w(x)] - E_{x \sim P_g} [f_w(x)] \quad (16)$$

**WGAN 的目标函数去掉了 log 项，其目的由判断输入图片是否为真改为尽量增加生成分布与真实分布之间的 wasserstein 距离，改为回归模型**

**Algorithm 1** WGAN, our proposed algorithm. All experiments in the paper used the default values  $\alpha = 0.00005$ ,  $c = 0.01$ ,  $m = 64$ ,  $n_{\text{critic}} = 5$ .

**Require:** :  $\alpha$ , the learning rate.  $c$ , the clipping parameter.  $m$ , the batch size.  $n_{\text{critic}}$ , the number of iterations of the critic per generator iteration.

**Require:** :  $w_0$ , initial critic parameters.  $\theta_0$ , initial generator's parameters.

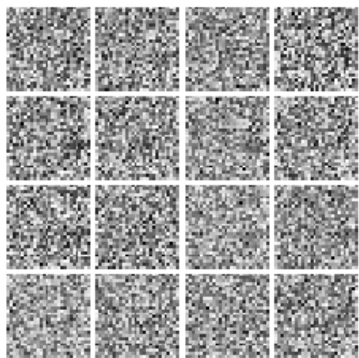
```
1: while  $\theta$  has not converged do
2:   for  $t = 0, \dots, n_{\text{critic}}$  do
3:     Sample  $\{x^{(i)}\}_{i=1}^m \sim \mathbb{P}_r$  a batch from the real data.
4:     Sample  $\{z^{(i)}\}_{i=1}^m \sim p(z)$  a batch of prior samples.
5:      $g_w \leftarrow \nabla_w \left[ \frac{1}{m} \sum_{i=1}^m f_w(x^{(i)}) - \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)})) \right]$ 
6:      $w \leftarrow w + \alpha \cdot \text{RMSPProp}(w, g_w)$ 
7:      $w \leftarrow \text{clip}(w, -c, c)$ 
8:   end for
9:   Sample  $\{z^{(i)}\}_{i=1}^m \sim p(z)$  a batch of prior samples.
10:   $g_\theta \leftarrow -\nabla_\theta \frac{1}{m} \sum_{i=1}^m f_w(g_\theta(z^{(i)}))$ 
11:   $\theta \leftarrow \theta - \alpha \cdot \text{RMSPProp}(\theta, g_\theta)$ 
12: end while
```

- 判别器最后一层去掉sigmoid
- 生成器和判别器的loss不取log
- 每次更新判别器的参数之后把它们的绝对值截断到不超过一个固定常数c
- 不要用基于动量的优化算法（包括momentum和Adam），推荐RMSPProp，SGD也行



## 模型收敛速度

WGAN



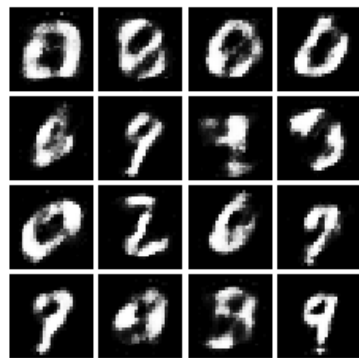
Step 0



Step 1000

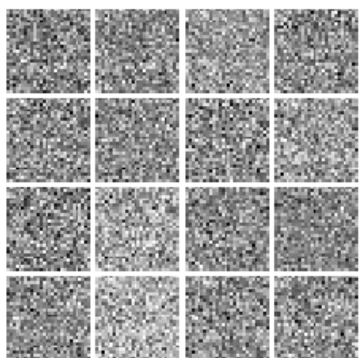


Step 5000

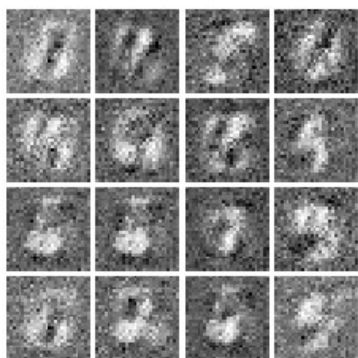


Step 10000

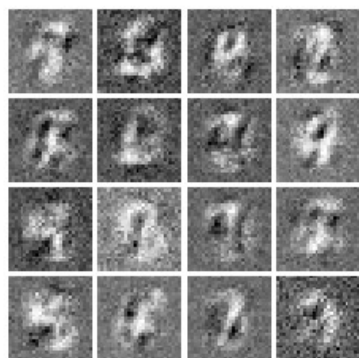
GAN



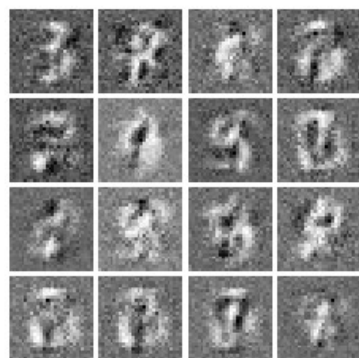
Step 0



Step 1000



Step 5000



Step 10000

WGAN 很快收敛， GAN 收敛速度慢



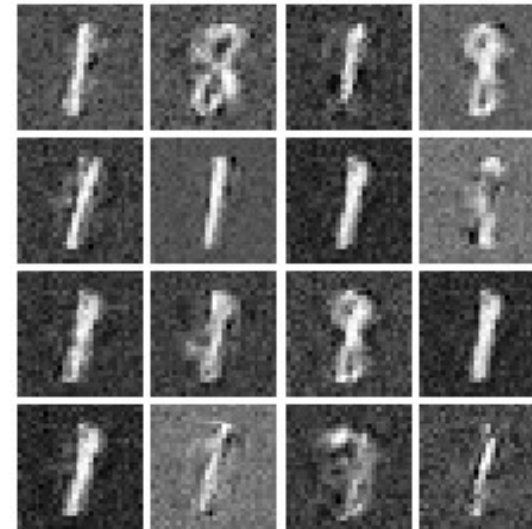
## 最终生成图片效果

WGAN



Step 10000

GAN



Step 418000

- WGAN 生成图片清晰且种类繁多
- GAN 生成图片模糊且种类单一

解决了问题 3 ， model collapse<sup>14</sup>





# 资源|17类对抗网络经典论文及开源代码（附源码）

2017-04-10 12:55

 人工智能 / 谷歌 / 技术

全球人工智能

文章来源：Github

对抗网络专题文献集

第一篇论文

[Generative Adversarial Nets](the first paper about it)

[Paper] : <https://arxiv.org/abs/1406.2661>

[Code] : <https://github.com/goodfeli/adversarial>

根据目前已有的 GAN 开源项目，看他们发表的论文，并在公共数据集上实现算法，并进行实验结果对比。大约需要两周时间。

The background features a complex geometric pattern of thin, light gray lines that form a hyperboloid of two sheets, creating a sense of depth and perspective. Scattered throughout the scene are numerous small, semi-transparent dots in various colors, including red, blue, green, yellow, orange, and purple. These dots are concentrated in clusters around the central text and along the edges of the hyperboloid's sheets.

**THANK YOU FOR WATCHING**