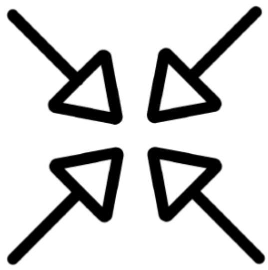
deeplearning.ai

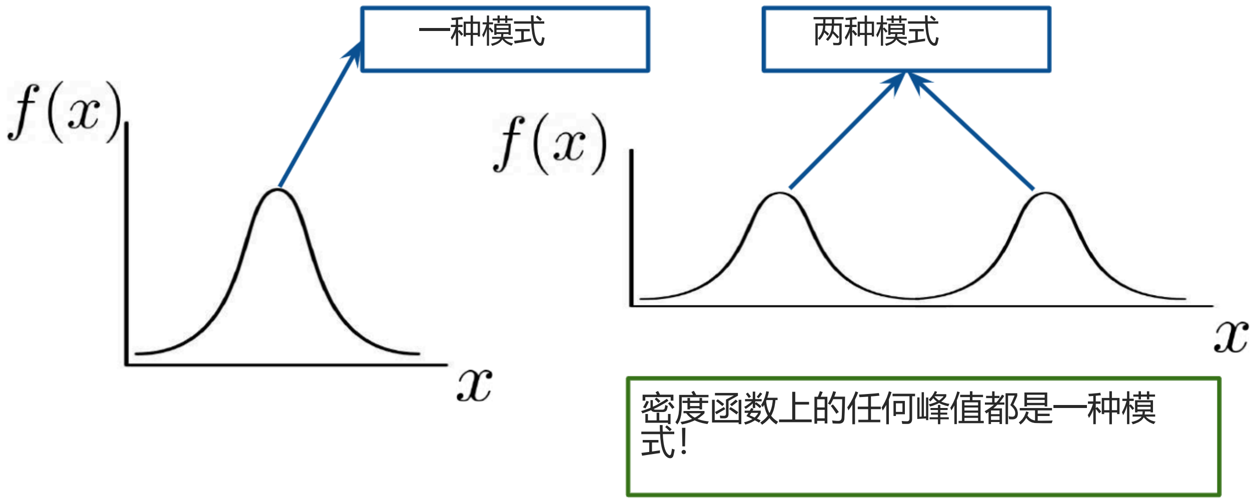
模式折叠

大纲

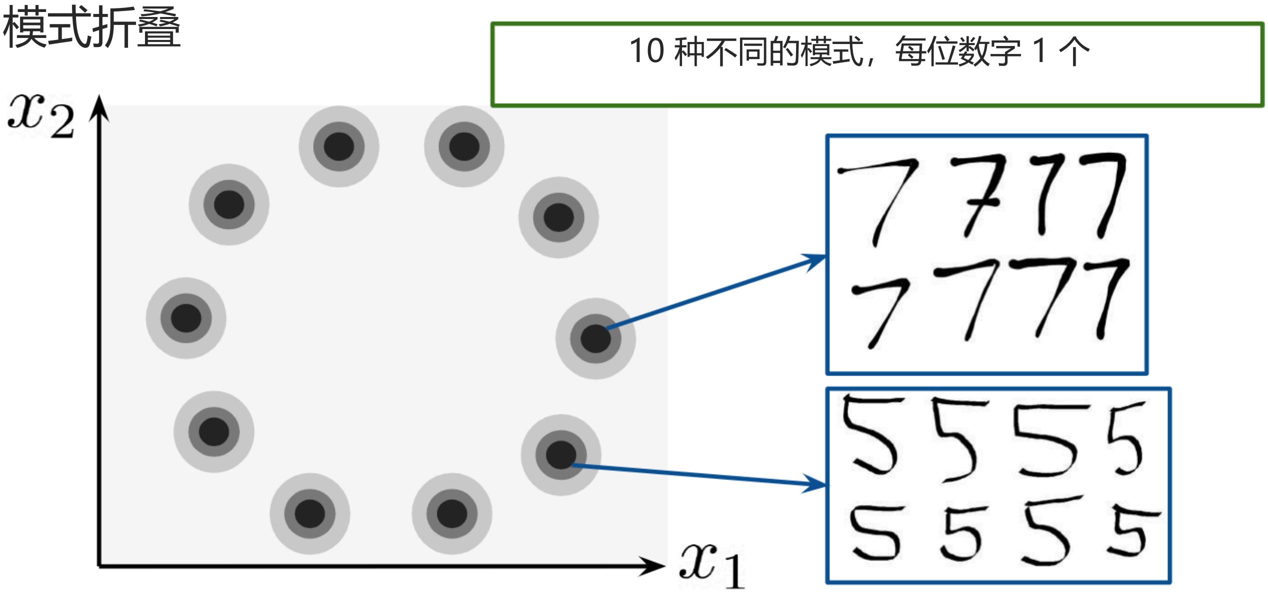
- 发行版中的模式
- GAN 中的模式崩溃
- 训练期间的直觉



模式折叠



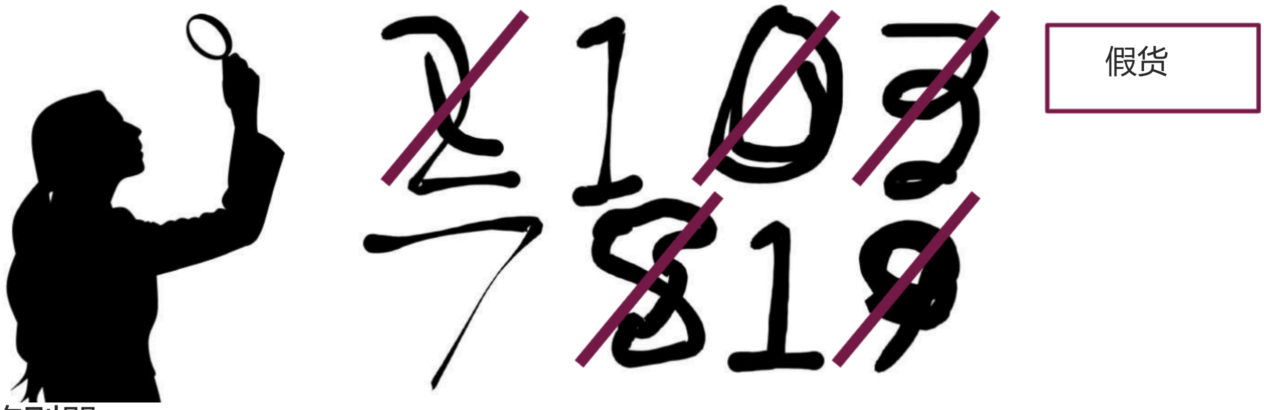
模式折叠



模式折叠

鉴别器

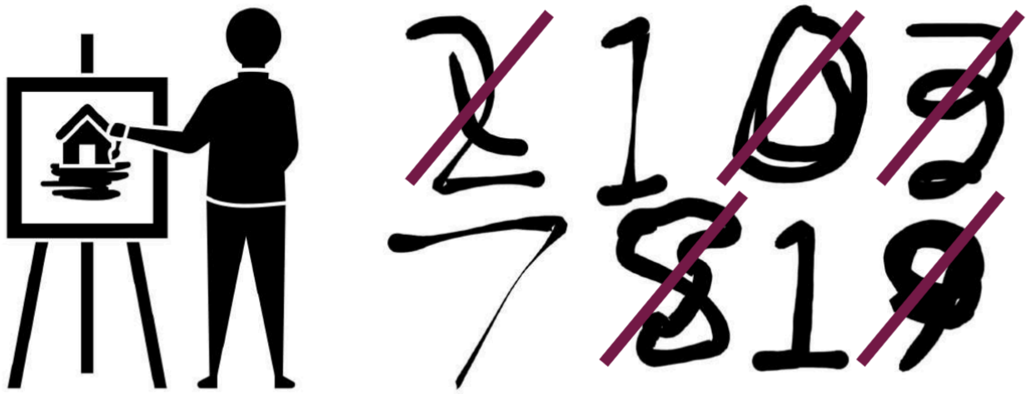
模式折叠



鉴别器

deeplearning.ai

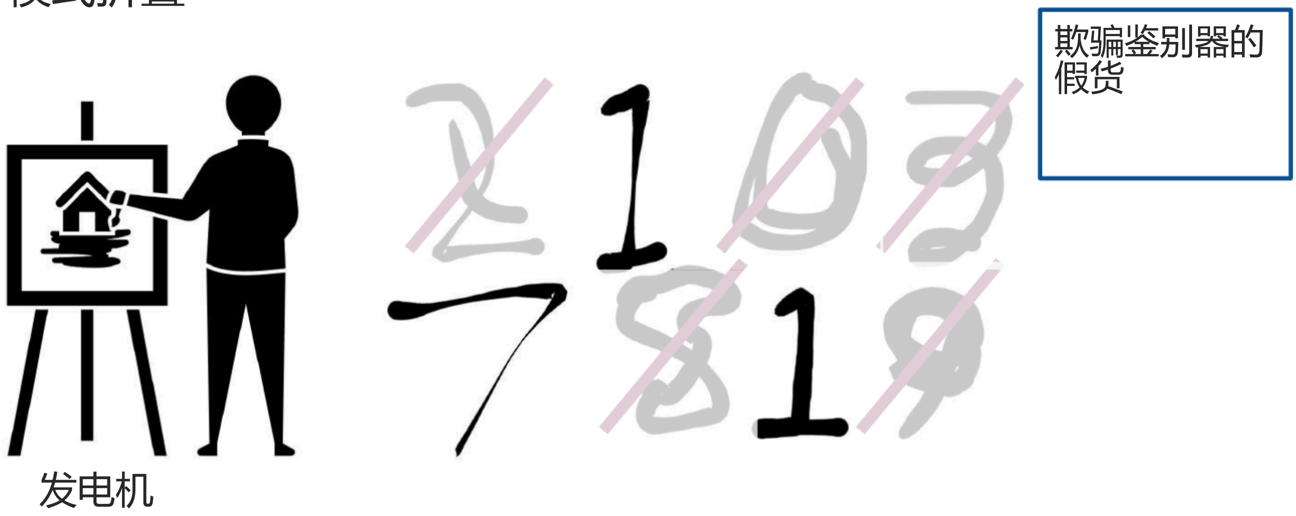
模式折叠



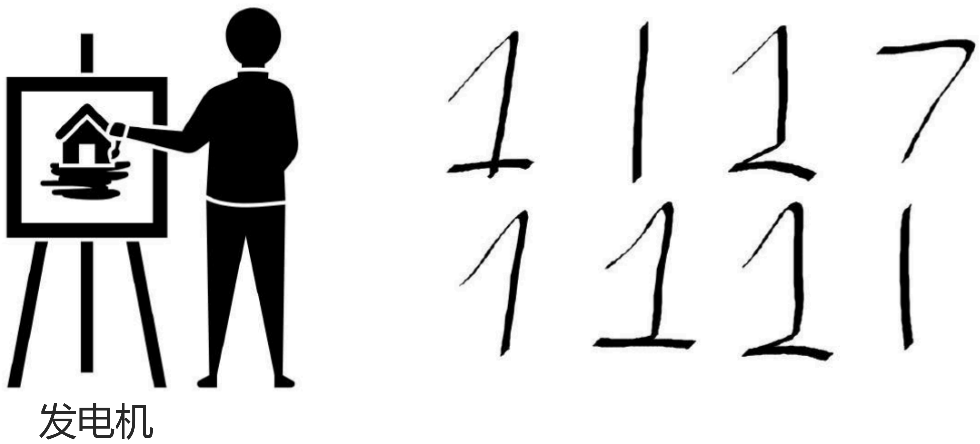
发电机

deeplearning.ai

模式折叠

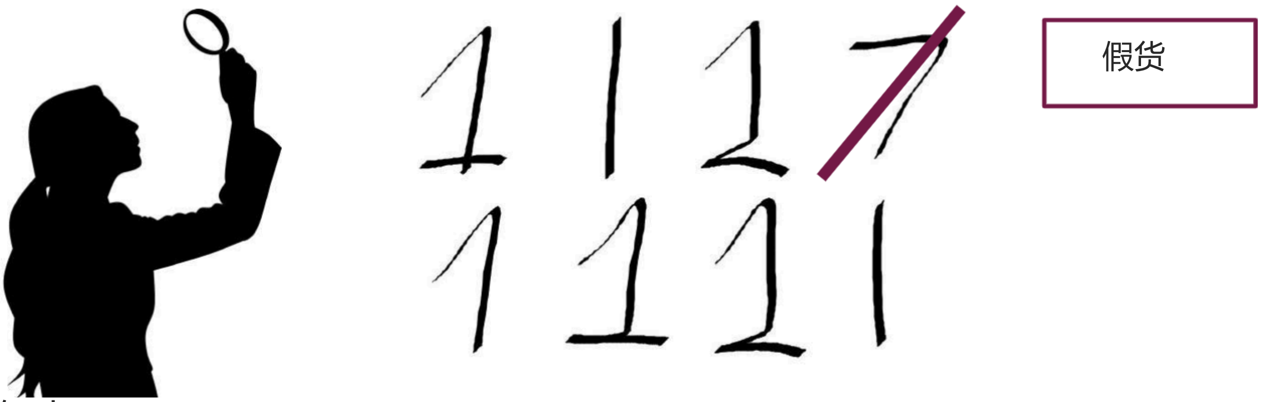


模式折叠



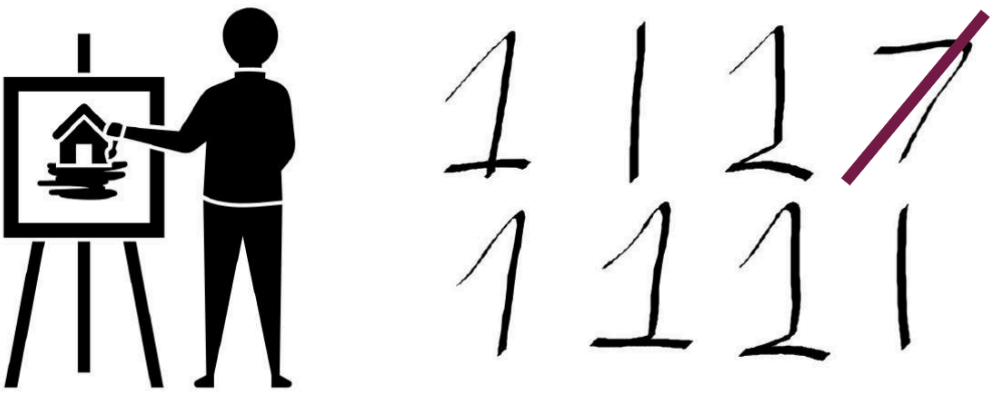
deeplearning.ai

模式折叠



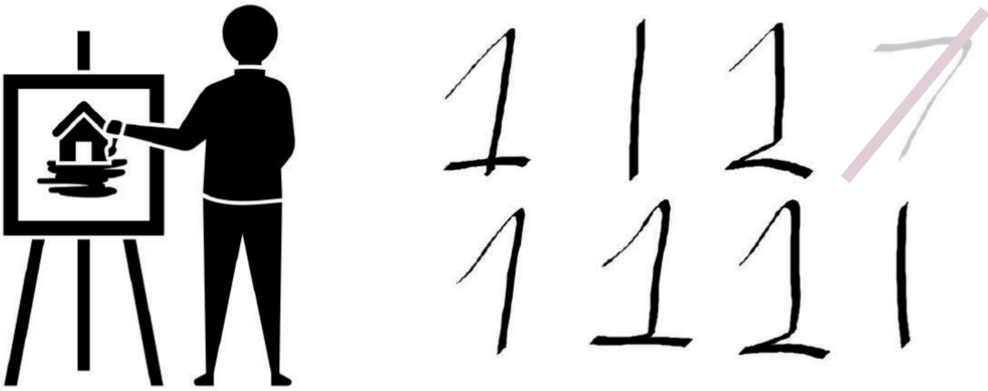
鉴别器

模式折叠



发电机

模式折叠

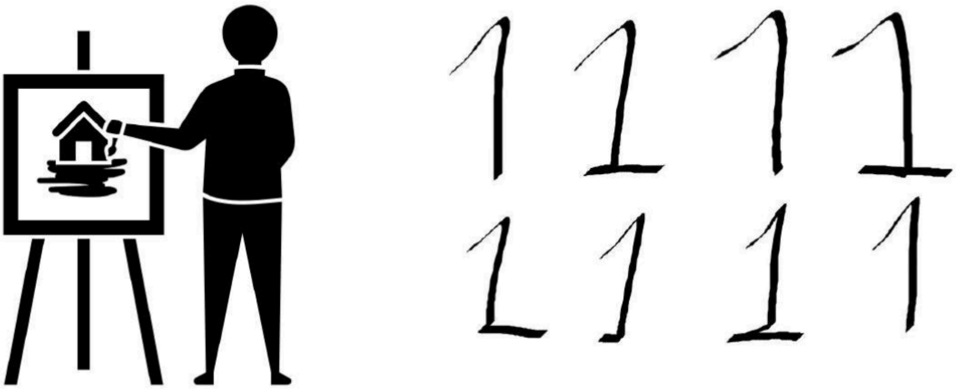


欺骗鉴别器的假货

发电机

deeplearning.ai

模式折叠



发电机

deeplearning.ai

总结

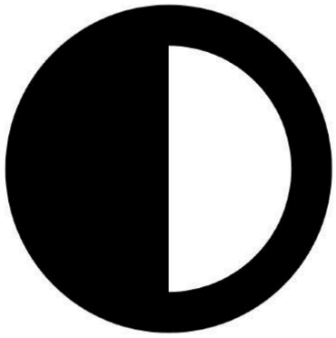
- 众数是特征分布中的峰值
- 真实数据集的典型情况
- 当生成器卡在一种模式下时，会发生模式崩溃

问题

BCE 损失

大纲

- GAN 中的 BCE 损失和最终目标
- BCE 损失问题



deeplearning.ai

GAN 中的 BCE 损失

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log h(x^{(i)},\theta) + (1-y^{(i)})\log(1-h(x^{(i)},\theta))]$$

预测 标签 特征

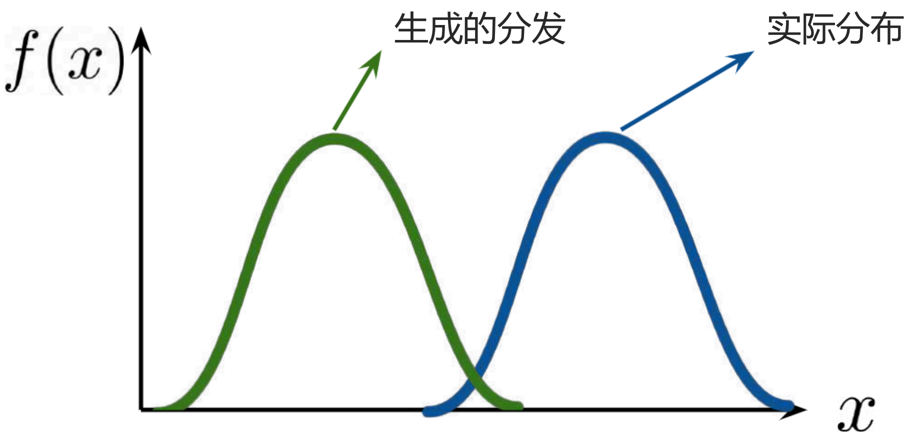
参数

最大化成本

最大限度地降低成本

deeplearning.ai

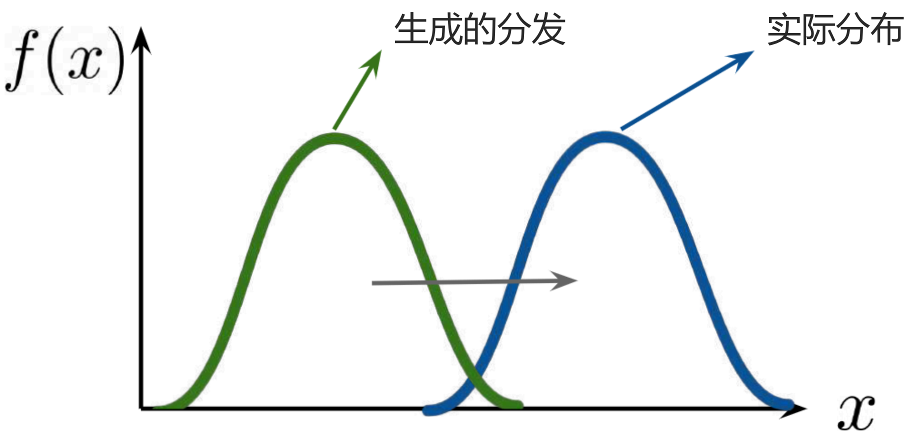
GAN 中的目标



deeplearning.ai

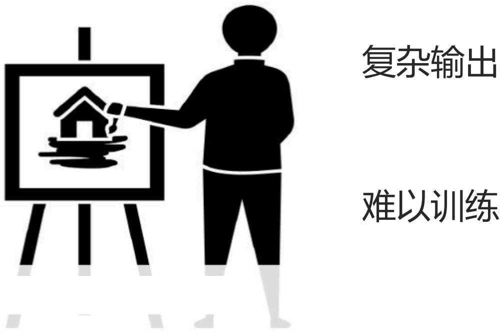
GAN 中的目标

使生成的分布和实际分布看起来相似



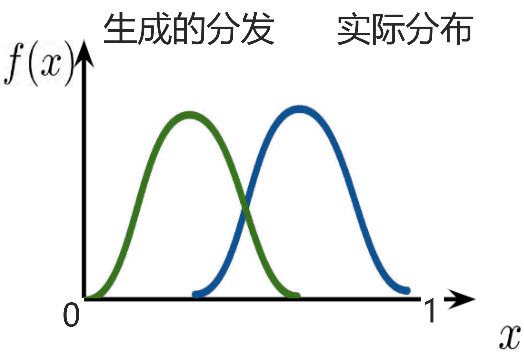
deeplearning.ai

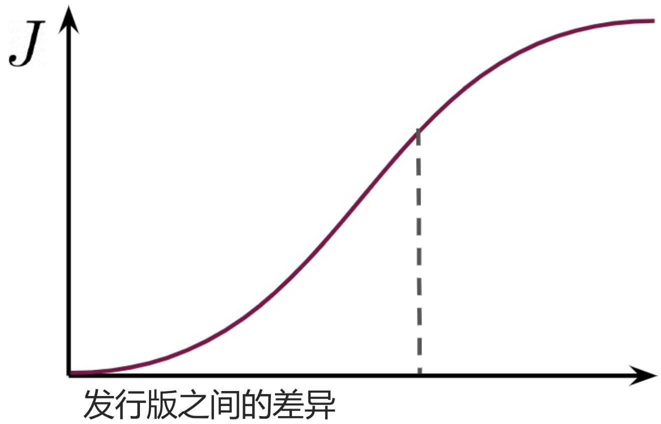
GAN 中的 BCE 损失批评更直接



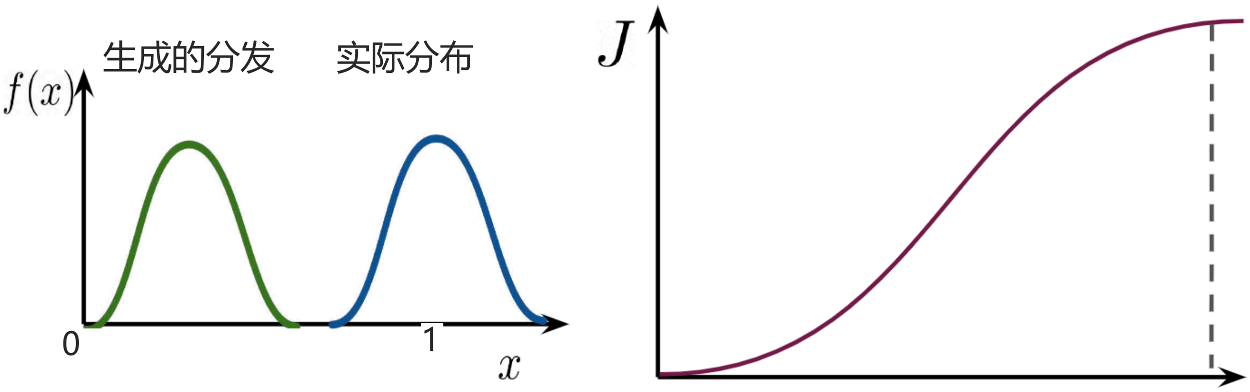
通常，判别器比生成器更好

BCE Loss 的问题

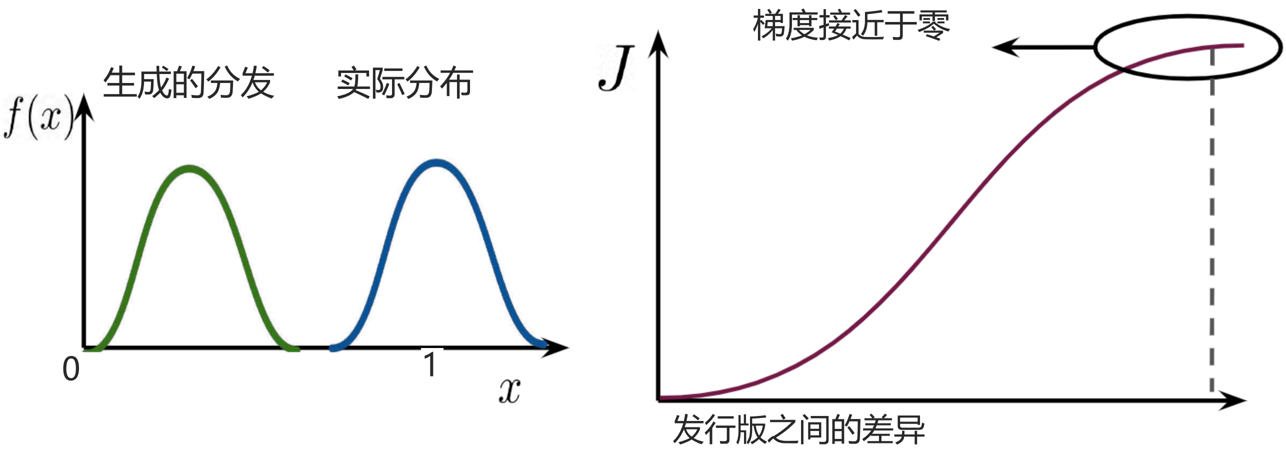




BCE Loss 的问题

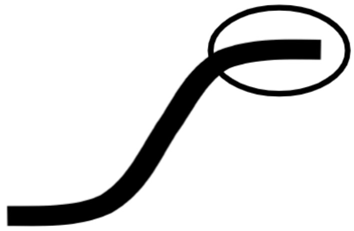


BCE Loss 的问题



总结

- GAN 尝试使真实分布和生成的分布看起来相似
- 当判别器改进太多时，由 BCE Loss 近似的函数将包含平坦区域
- 成本函数上的平坦区域 = 梯度消失



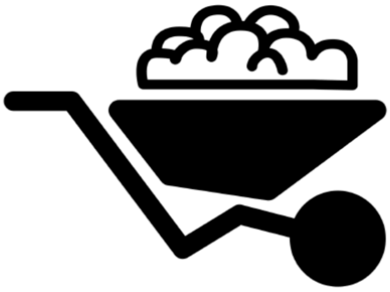
deeplearning.ai

deeplearning.ai

Earth Mover的距离

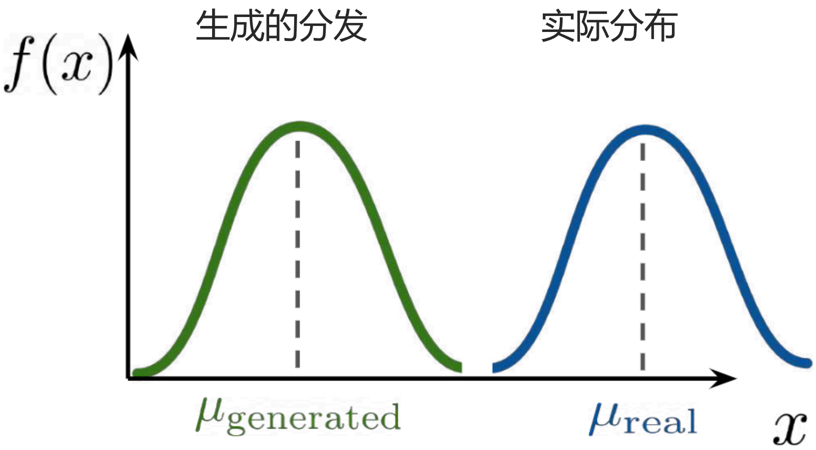
大纲

- 推土机距离 （EMD）
- 为什么它解决了 BCE Loss 的梯度消失问题



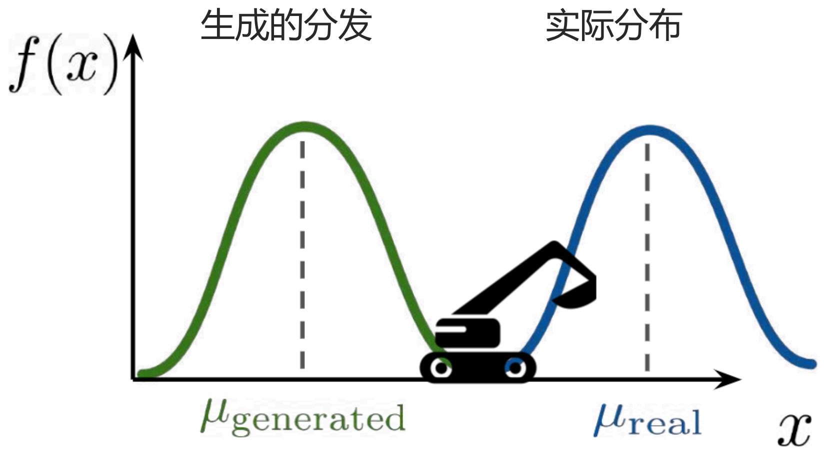
deeplearning.ai

推土机的距离



deeplearning.ai

推土机的距离

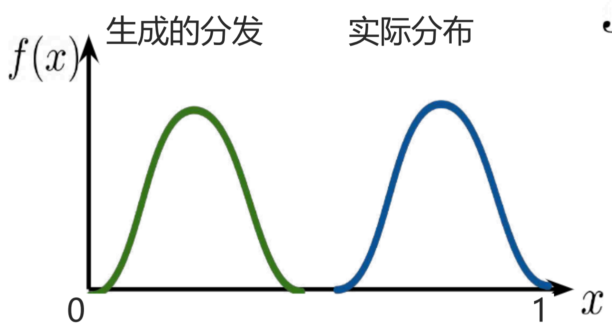


努力使生成的分布等于实际分布

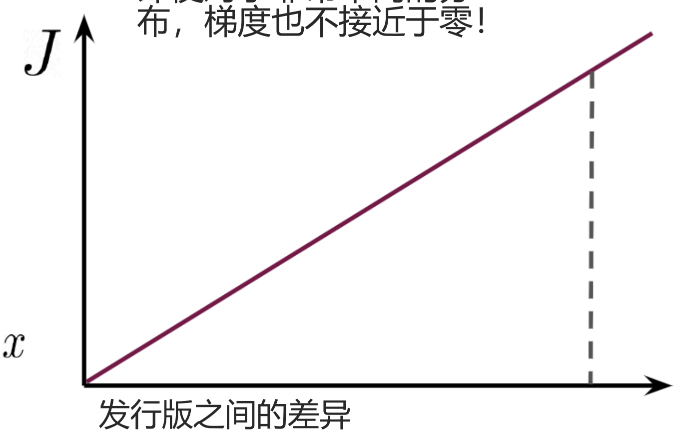
取决于
移动的距离和数量

deeplearning.ai

推土机的距离



即使对于非常不同的分布，梯度也不接近于零！



deeplearning.ai

总结

- 推土机距离 （EMD） 是数量和距离的函数
- 当分布差异很大时，没有平坦的区域
- 近似 EMD 解决了与 BCE 相关的问题

deeplearning.ai

deeplearning.ai

水垢

损失

大纲

- 简化的 BCE 损失
- W-Loss 及其与 BCE Loss 的比较

简化的 BCE 损失

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log h(x^{(i)},\theta) + (1-y^{(i)})\log(1-h(x^{(i)},\theta))]$$

$$\min_{d}\max_{g}$$

鉴别器 发电机 最大限度地

最大化成本

简化的 BCE 损失

$$J(\theta) = \boxed{-\frac{1}{m}\sum_{i=1}^{m}}\left[\boxed{y^{(i)}\log h(x^{(i)},\theta)} + (1-y^{(i)})\log(1-h(x^{(i)},\theta))\right]$$

$$\min_{d}\max_{g} -[\mathbb{E}(\log(d(x))) + \mathbb{E}(\qquad\qquad)]$$

最大限度地
降低成本

最大化成
本

deeplearning.ai

简化的 BCE 损失

$$J(\theta) = \boxed{-\frac{1}{m}\sum_{i=1}^{m}}\left[\boxed{y^{(i)}\log h(x^{(i)},\theta)} + \boxed{(1-y^{(i)})\log(1-h(x^{(i)},\theta))}\right]$$

$$\min_{d}\max_{g} -[\mathbb{E}(\log(d(x))) + \mathbb{E}(1-\log(d(g(z))))]$$

最大限度地
降低成本

最大化成
本

deeplearning.ai

W 损失

W-Loss 近似于地球移动器的距离

deeplearning.ai

W 损失

W-Loss 近似于地球移动器的距离

$$\min_{g} \max_{c} \ \mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))$$

deeplearning.ai

W 损失

W-Loss 近似于地球移动器的距离

$$\min_{g} \max_{c} \ \mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))$$

最大化距离

deeplearning.ai

W 损失

W-Loss 近似于地球移动器的距离

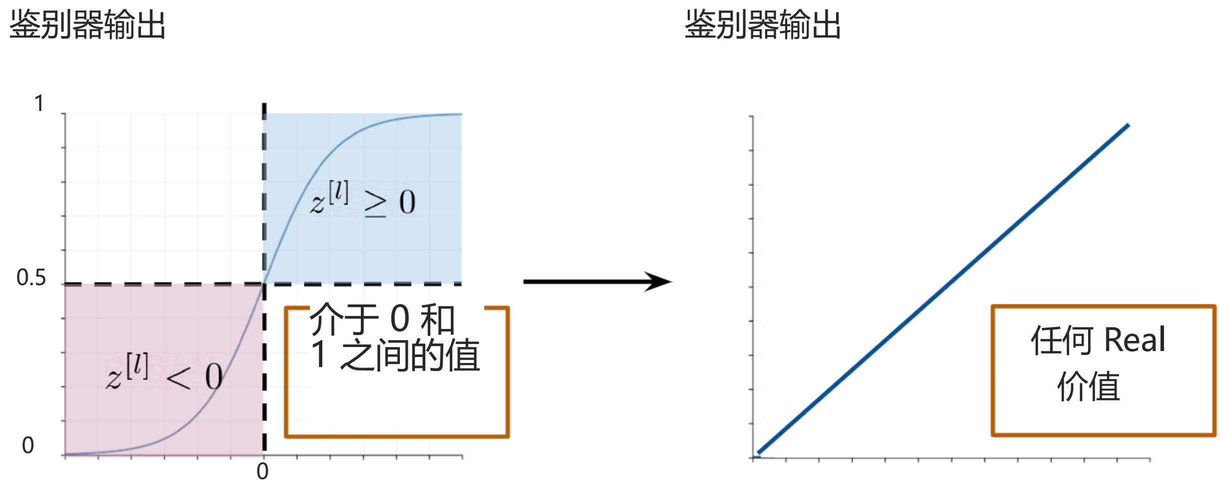
$$\min_{g} \max_{c} \ \mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))$$

最小化距离

最大化距离

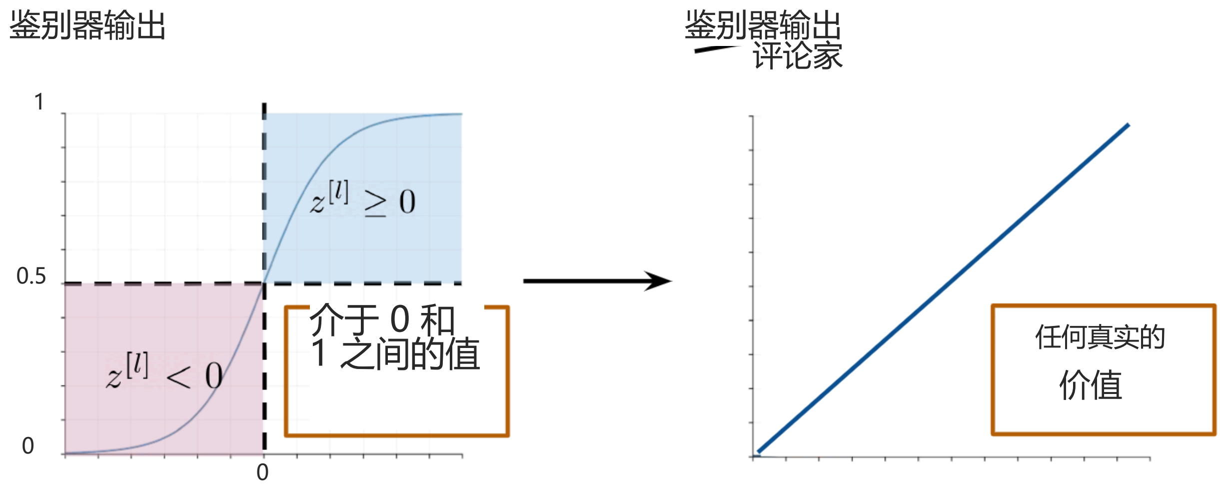
deeplearning.ai

甄别器输出



deeplearning.ai

甄别器输出



deeplearning.ai

W 损失与 BCE 损失

BCE 损失	W 损失
鉴别器输出介于 0 和 1 之间	Critic 输出任意数字
$-[\mathbb{E}(\log(d(x))) + \mathbb{E}(1 - \log(d(g(z))))]$	$\mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))$

W-Loss 有助于解决模式崩溃和梯度消失问题

deeplearning.ai

总结

- W-Loss 看起来与 BCE Loss 非常相似
- W-Loss 可防止模式崩溃和梯度消失问题

deeplearning.ai

deeplearning.ai

Wasserstein Critic 的条件

大纲

- 评论家神经网络上的连续性条件
- 为什么这种情况很重要

deeplearning.ai

W 损失的条件

$$\min_g \max_c \; \mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))$$

deeplearning.ai

W 损失的条件

$$\min_g \max_c \; \mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))$$

deeplearning.ai

W 损失的条件

$$\min_{g} \max_{c} \ \mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))$$

deeplearning.ai

W 损失的条件

$$\min_{g} \max_{\boxed{c}} \ \mathbb{E}(\boxed{c}(x)) - \mathbb{E}(\boxed{c}(g(z)))$$

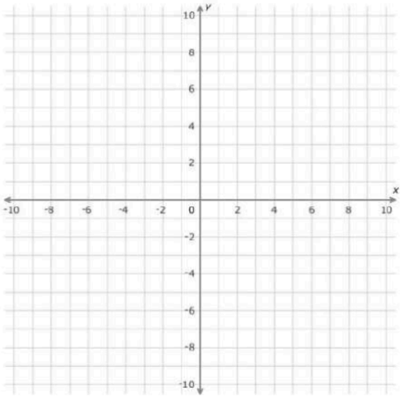
需要为 1-Lipschitz 连续

deeplearning.ai

W 损失的条件

Critic 需要为 1 L 连续

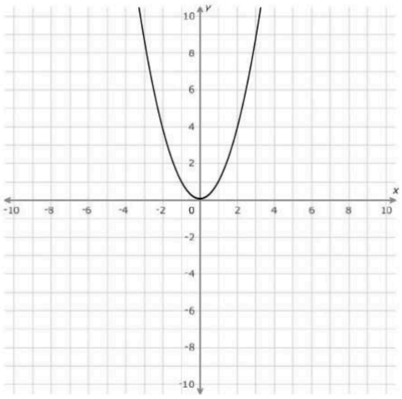
对于每个点，梯度的范数应最多为 1



W 损失的条件

Critic 需要为 1 L 连续

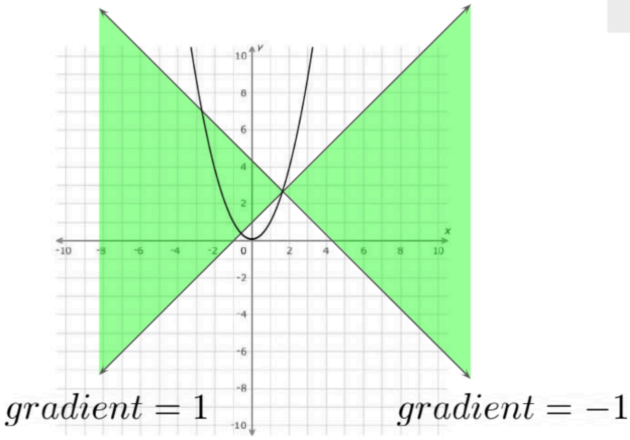
对于每个点，梯度的范数应最多为 1



W 损失的条件

Critic 需要为 1 L 连续

对于每个点，梯度的范数应最多为 1

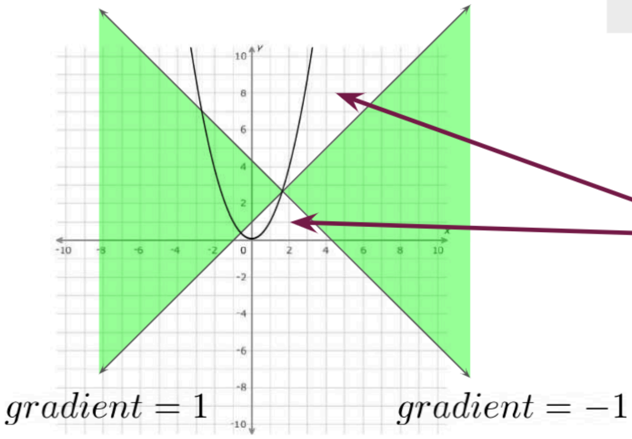


deeplearning.ai

W 损失的条件

Critic 需要为 1 L 连续

梯度的范数应为
每个点最多 1 个



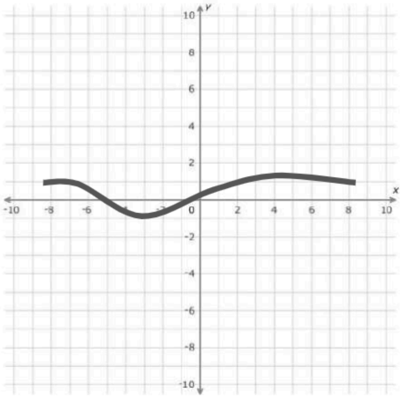
不是 1 L 连续

deeplearning.ai

W 损失的条件

Critic 需要为 1 L 连续

对于每个点，梯度的范数应最多为 1

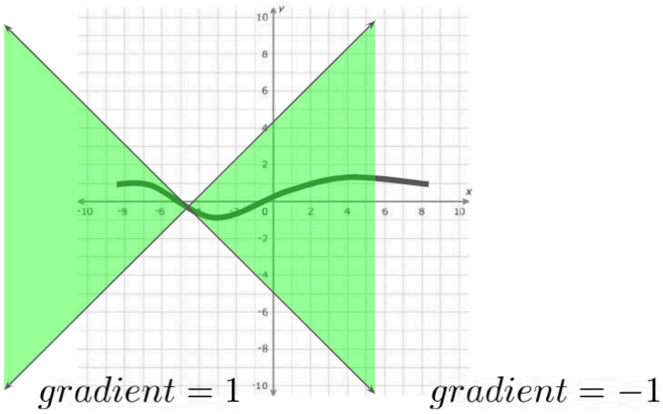


deeplearning.ai

W 损失的条件

Critic 需要为 1 L 连续

对于每个点，梯度的范数应最多为 1

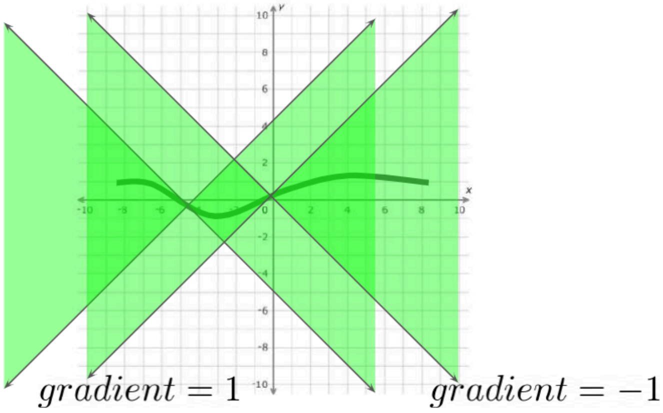


deeplearning.ai

W 损失的条件

Critic 需要为 1 L 连续

对于每个点，梯度的范数应最多为 1

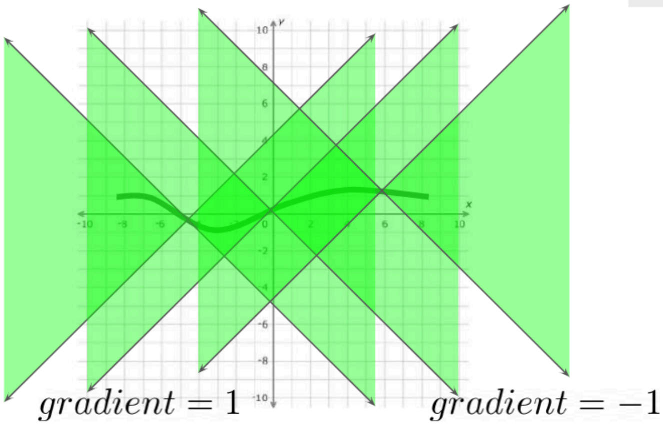


deeplearning.ai

W 损失的条件

Critic 需要为 1 L 连续

对于每个点，梯度的范数应最多为 1

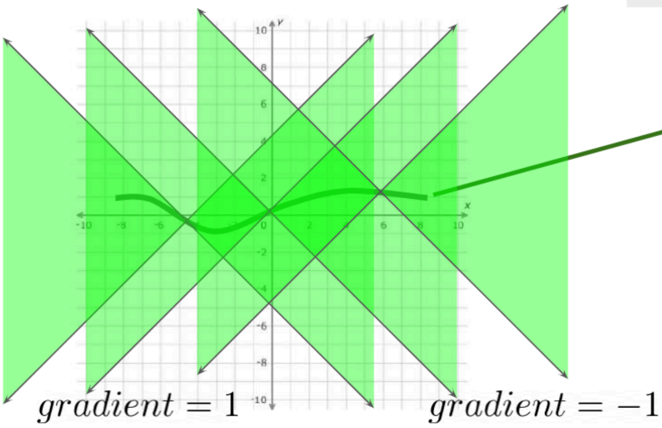


deeplearning.ai

W 损失的条件

Critic 需要为 1 L 连续

对于每个点，梯度的范数应最多为 1



1 L 连续

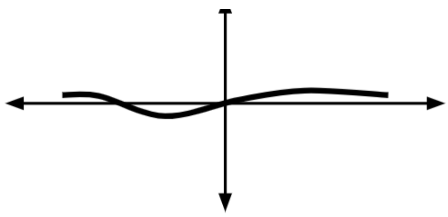
W 损失有效

使用 W-Loss 训练稳定的神经网络时需要

deeplearning.ai

总结

- 使用 W-Loss 时，Critic 的神经网络需要为 1-L Continuous
- 此条件可确保 W-Loss 有效地近似地接近地球移动器的距离



deeplearning.ai

deeplearning.ai

1-Lipschitz 连续性执行

大纲

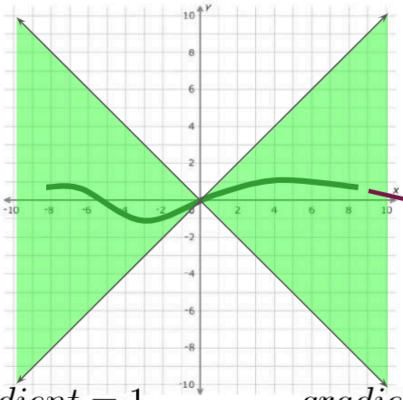
- 权重裁剪和渐变惩罚
- 梯度惩罚的优点

deeplearning.ai

1-L 强制执行

Critic 需要为 1 L 连续

梯度的范数最多 1



$$||\nabla f(x)||_2 \leq 1$$

函数的斜率
最多 1 个

1 L 强制：重量夹

权重剪切将评论家的权重强制到固定的间隔

梯度下降到
更新权重

剪辑评论家的权重

限制了
评论家

1-L 执行：梯度惩罚

$$\min_{g} \max_{c} \; \mathbb{E}(c(x)) - \mathbb{E}(c(g(z))) + \lambda \mathrm{reg}$$

正则化
评论家的渐变

deeplearning.ai

1-L 执行：梯度惩罚

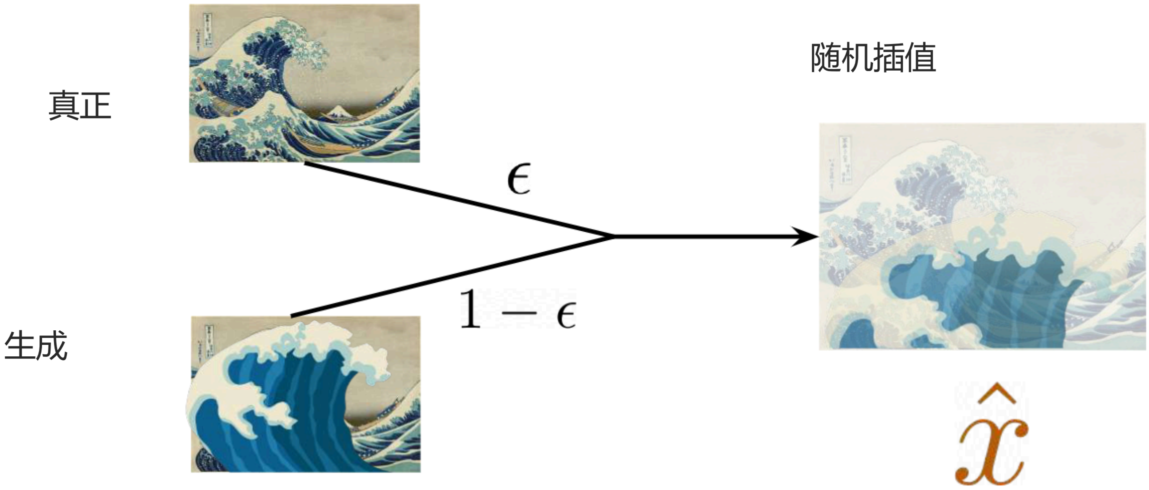
真正

ϵ

随机插值

deeplearning.ai

1-L 执行：梯度惩罚



deeplearning.ai

1-L 执行：梯度惩罚

$$\mathbb{E}(||\nabla c(\hat{x})||_2 - 1)^2$$

正则化项

deeplearning.ai

1-L 执行：梯度惩罚

$$\mathbb{E}(||\nabla c(\boxed{\hat{x}})||_2 - 1)^2$$

正则化项

1-L 执行：梯度惩罚

$$\mathbb{E}(||\nabla c(\boxed{\hat{x}})||_2 - 1)^2$$

正则化项

$$\epsilon x + (1 - \epsilon) g(z)$$

插值

1-L 执行：梯度惩罚

$$\mathbb{E}(||\nabla c(\boxed{\hat{x}})||_2 - 1)^2$$

正则化项

$$\epsilon \boxed{x} + (1 - \epsilon)g(z)$$

插值

真正

deeplearning.ai

1-L 执行：梯度惩罚

$$\mathbb{E}(||\nabla c(\boxed{\hat{x}})||_2 - 1)^2$$

正则化项

$$\epsilon \boxed{x} + (1 - \epsilon)\boxed{g(z)}$$

插值

真正

生成

deeplearning.ai

把它们放在一起

$$\min_{g} \max_{c} \mathbb{E}(c(x)) - \mathbb{E}(c(g(z))) + \lambda \mathbb{E}(||\nabla c(\hat{x})||_2 - 1)^2$$

把它们放在一起

$$\min_{g} \max_{c} \boxed{\mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))} + \lambda \mathbb{E}(||\nabla c(\hat{x})||_2 - 1)^2$$

使 GAN 不易出现模式折叠和渐变消失

把它们放在一起

$$\min_{g} \max_{c} \boxed{\mathbb{E}(c(x)) - \mathbb{E}(c(g(z)))} \boxed{+\lambda \mathbb{E}(||\nabla c(\hat{x})||_2 - 1)^2}$$

使 GAN 不易出现模式折叠和渐变消失

尝试使 critic 为 1-L Continuous，使损失函数为连续且可微分

deeplearning.ai

总结

- 砝码削缩和梯度惩罚是强制 1 L 连续性的方法
- 梯度惩罚往往效果更好

deeplearning.ai