CCL 2019讲习班

暨中国中文信息学会《前沿技术讲习班》(ATT)第19期

生成对抗网络

邱锡鹏

复旦大学

2019/10/18

参考

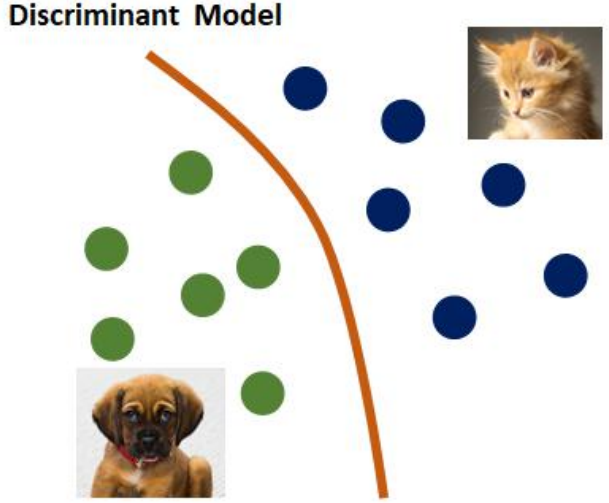
- 《神经网络与深度学习》 https://nndl.github.io/
  - 第8章 概率图模型
  - 第9章 无监督学习
  - 第13章 深度生成模型
  - 第15章 序列生成模型

- 一些例子来自于李宏毅《Introduction of Generative Adversarial Network (GAN)》

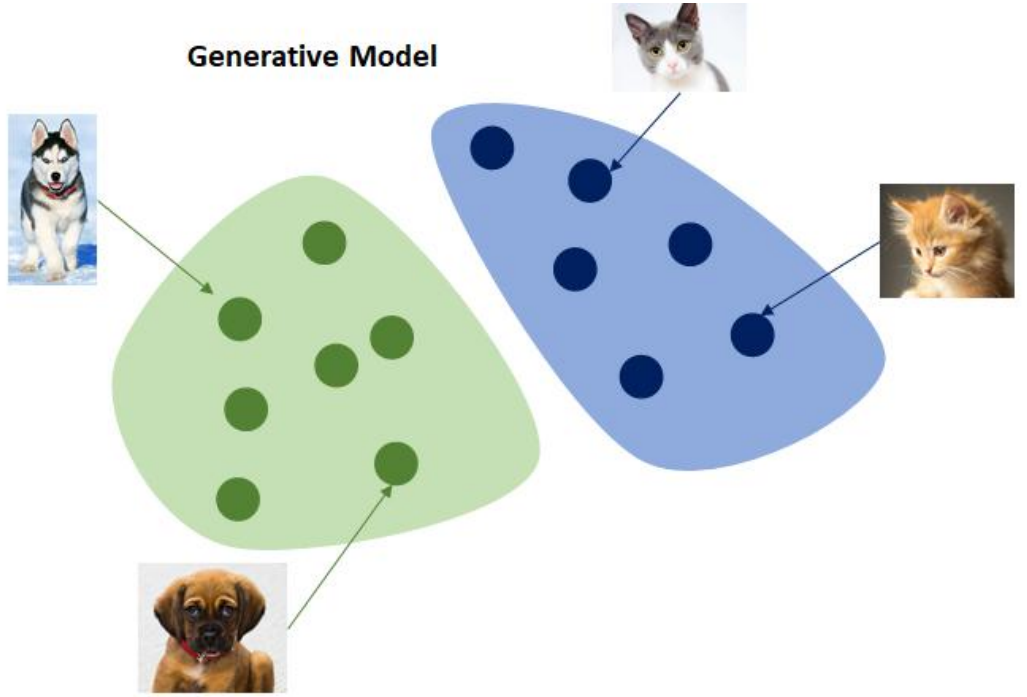
生成模型

机器学习的两种范式

- Discriminative VS Generative

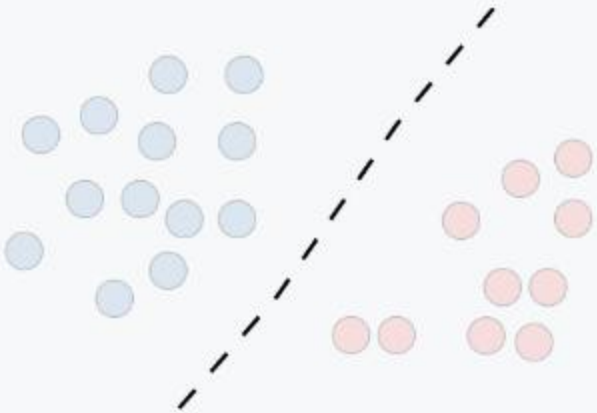
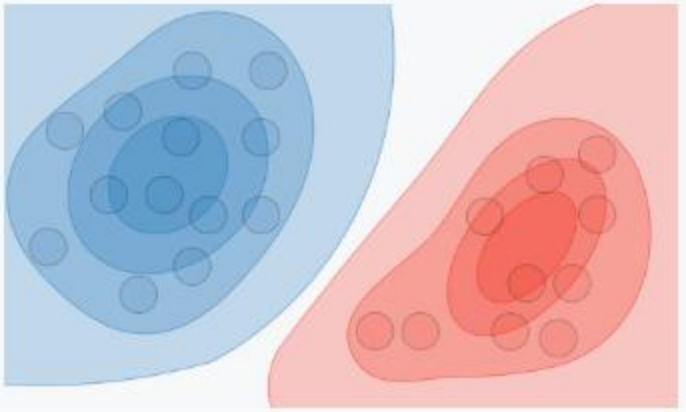


$p(y|x)$



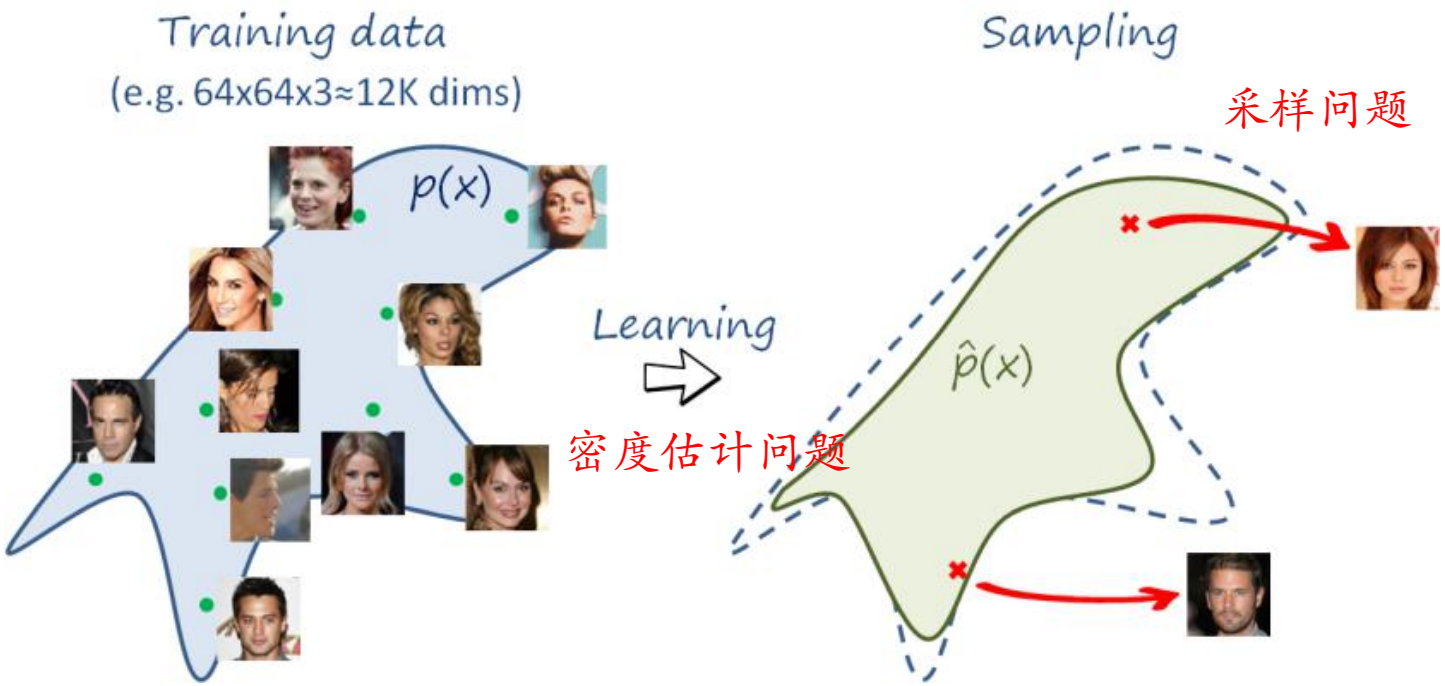
$p(x, y)$

机器学习的两种范式

	Discriminative model	Generative model
Goal	Directly estimate $P(y\|x)$	Estimate $P(x\|y)$ to then deduce $P(y\|x)$
What's learned	Decision boundary	Probability distributions of the data
Illustration		
Examples	Regressions, SVMs	GDA, Naive Bayes

（概率）生成模型：可以生成数据的模型

- 生成模型包含两部分：
  - 密度估计
  - 采样



（概率）密度估计

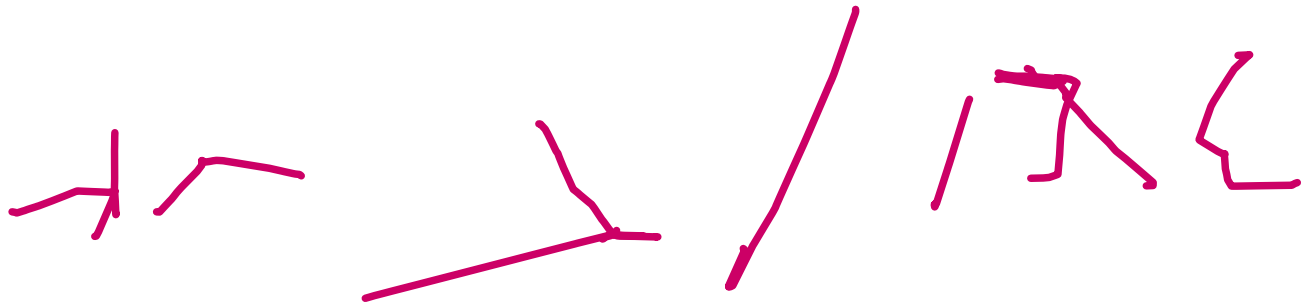
- 密度估计方法可以分为两类
  - 参数密度估计（Parametric Density Estimation）是根据先验知识假设随机变量服从某种分布，然后通过训练样本来估计分布的参数。
  - 非参数密度估计（Nonparametric Density Estimation）是不假设数据服从某种分布，通过将样本空间划分为不同的区域并估计每个区域的概率来近似数据的概率密度函数。

最大似然估计 (Maximum Likelihood Estimation)

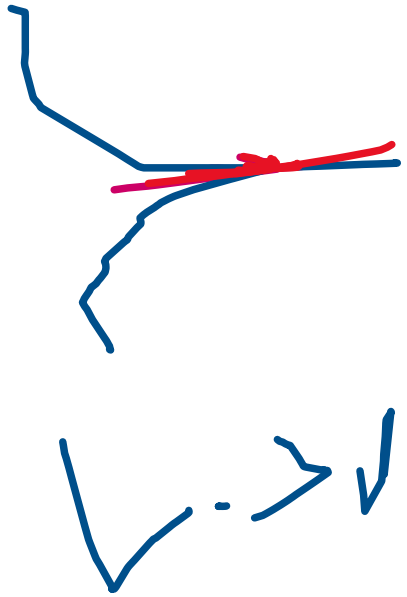
- 数据集$D = \{\boldsymbol{x}^{(n)}\}_{i=1}^{N}$
  - 从某个未知分布中独立抽取的N 个训练样本
- 假设$\boldsymbol{x}$服从概率密度函数为$p(\boldsymbol{x};\theta)$的分布，其对数似然函数为

$$\log p(\mathcal{D};\theta) = \sum_{n=1}^{N} \log p(\boldsymbol{x}^{(n)};\theta)$$

- 如何计算θ?
  - 最大似然估计



一个简单的例子：正态分布



假设样本 $\boldsymbol{x} \in \mathbb{R}^d$ 服从正态分布

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

$$\boldsymbol{\mu}^{ML} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{x}^{(n)},$$

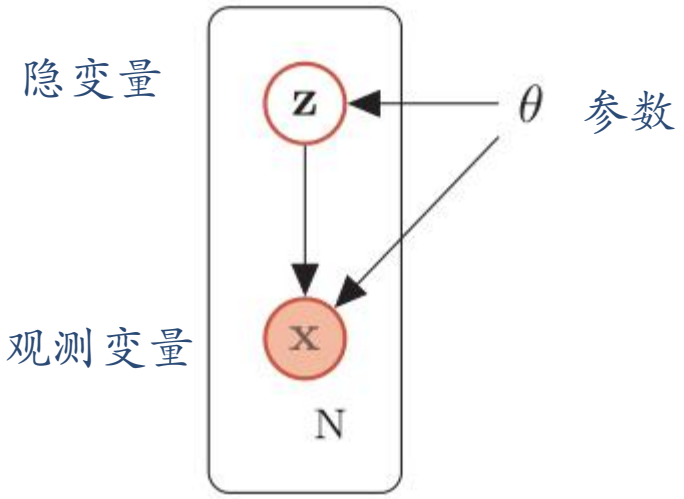
$$\Sigma^{ML} = \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^{\mathrm{T}}$$

含隐变量的概率生成模型

- 生成模型

$$p(x, z; \theta) = p(x|z; \theta)p(z; \theta)$$

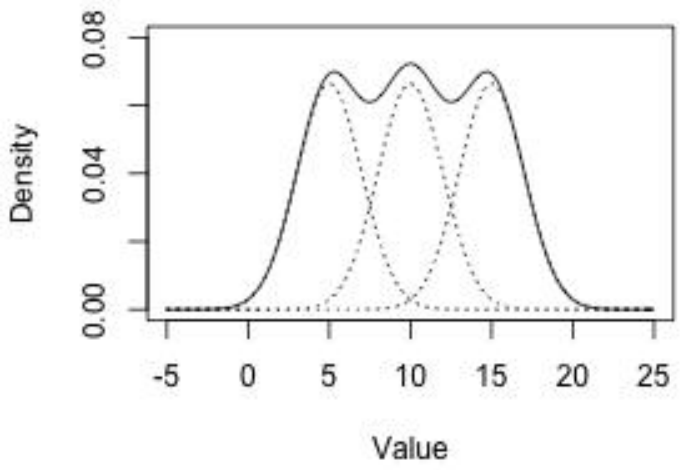
- 生成数据x的过程分为两步：
  - 根据先验分布$p(z; \theta)$采样得到z；
  - 根据条件分布$p(x|z; \theta)$采样得到x。
    - 当p(x|z;θ)比较复杂时，采样比较困难!

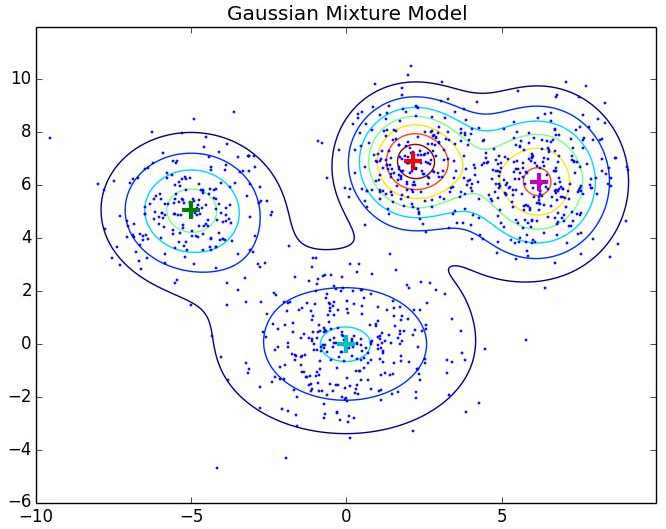


生成模型的概率图表示

一个简单的例子：高斯混合模型

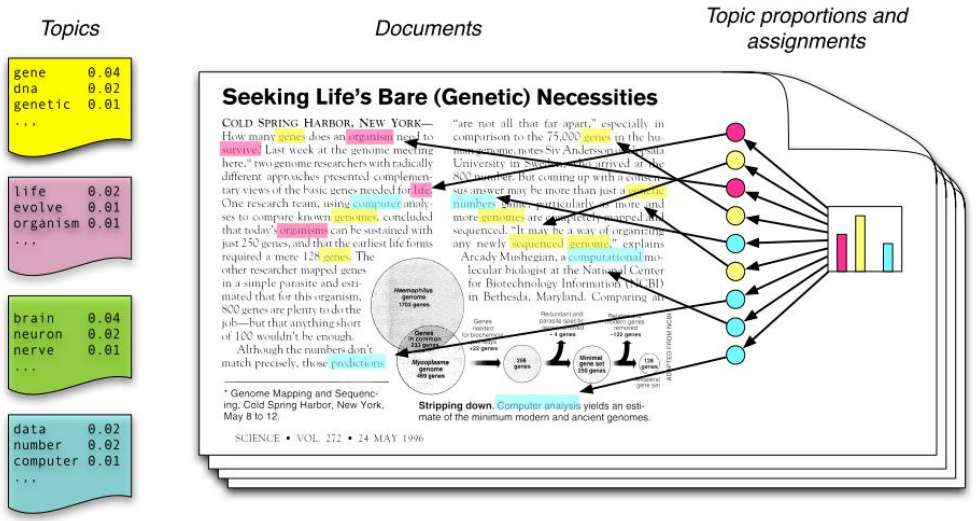
- 高斯混合模型（Gaussian Mixture Model，GMM）是由多个高斯分布组成的模型，其密度函数为多个高斯密度函数的加权组合。





另一个简单的例子：概率主题模型

- Probabilistic Topic Models
- 生成文档可以看成是如下两个过程：
  - 随机产生一个主题分布p(z)；
  - 对文档中的每个词：
    - (a) 根据主题分布p(z)，随机选择一个主题z；
    - (b) 根据主题z对应的词概率分布p(x|z)，随机选择一个词x。



含隐变量的概率生成模型

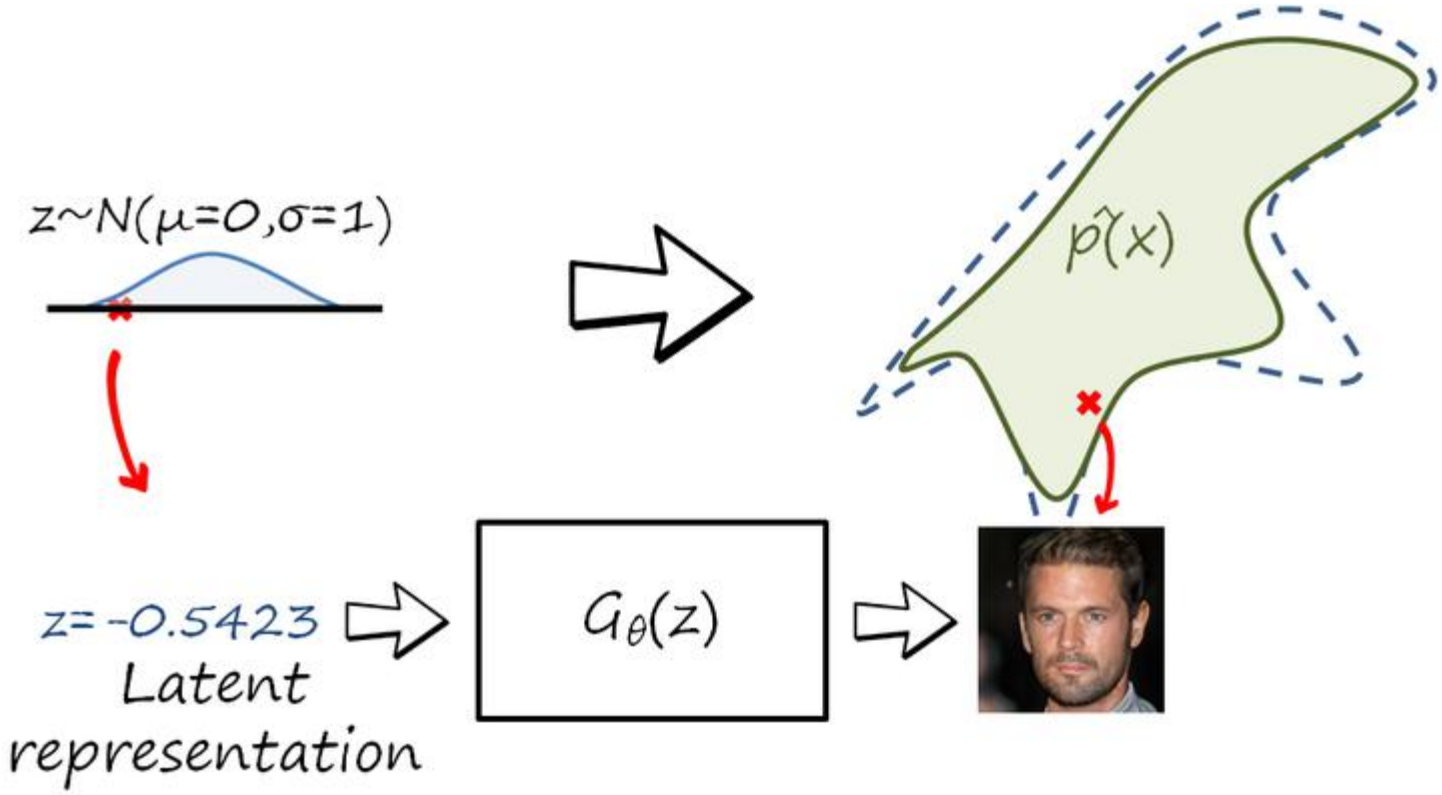
- 密度估计
  - EM算法
  - 变分法
  - 变分自编码器（VAE）

采样

- 随机采样方法
  - 直接采样：均匀分布
  - 间接采样
    - 逆函数法、拒绝采样、重要性采样、马尔可夫链蒙特卡罗（MCMC）方法

- 当分布比较复杂时，采样十分困难！

生成数据的另一种思路



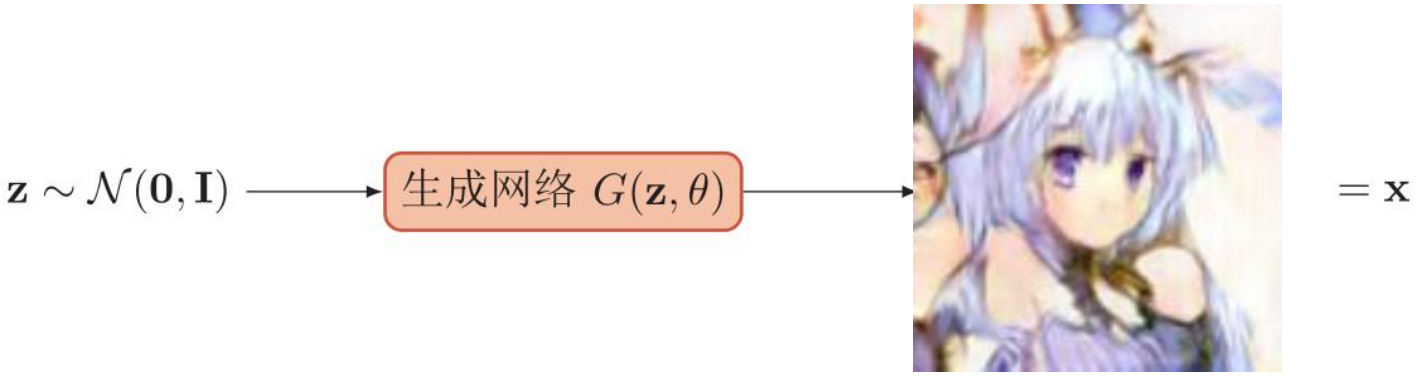
生成对抗网络

显式密度模型和隐式密度模型

- 显式密度模型
  - 显示地构建出样本的密度函数$p(x;\theta)$，并通过最大似然估计来求解参数；
  - 变分自编码器、深度信念网络
- 隐式密度模型
  - 不显示地估计出数据分布的密度函数
  - 但能生成符合数据分布$p_r(x)$的样本
  - 无法用最大似然估计

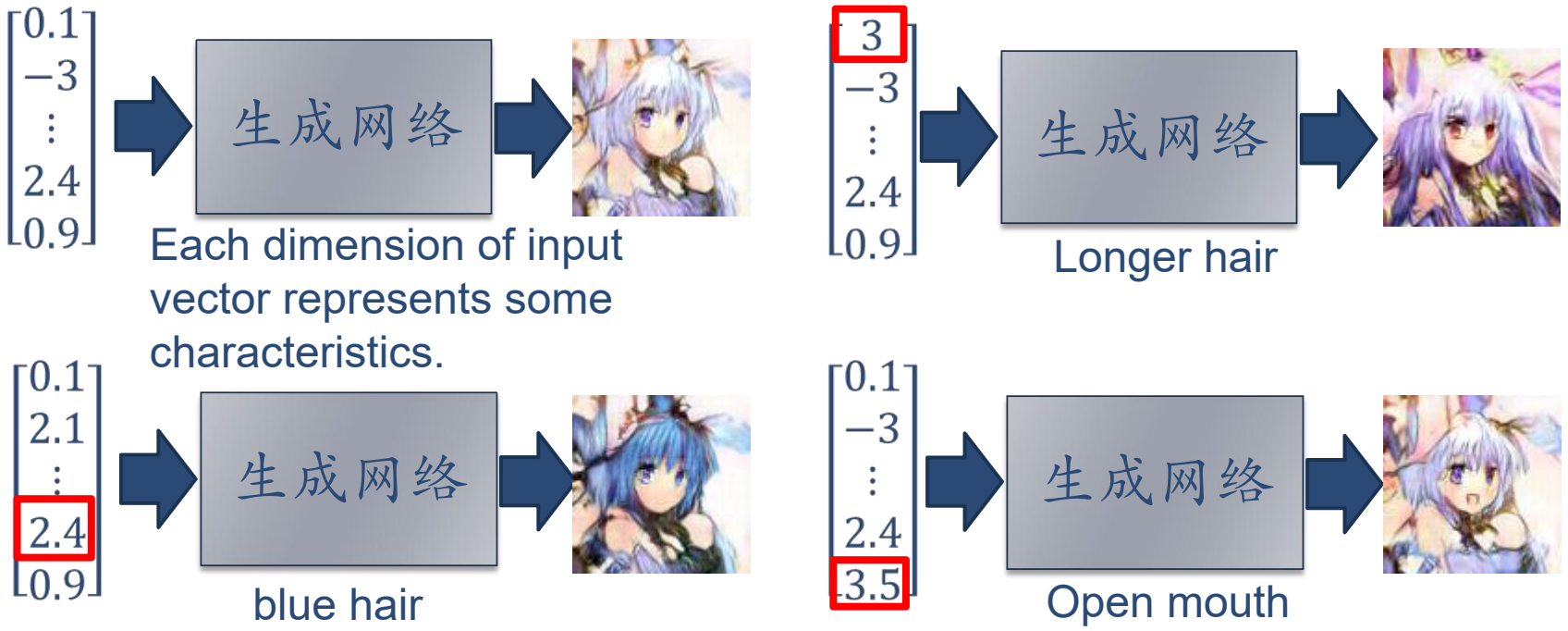
生成网络

- 生成网络从潜在空间（latent space）中随机采样作为输入，其输出结果需要尽量模仿训练集中的真实样本。



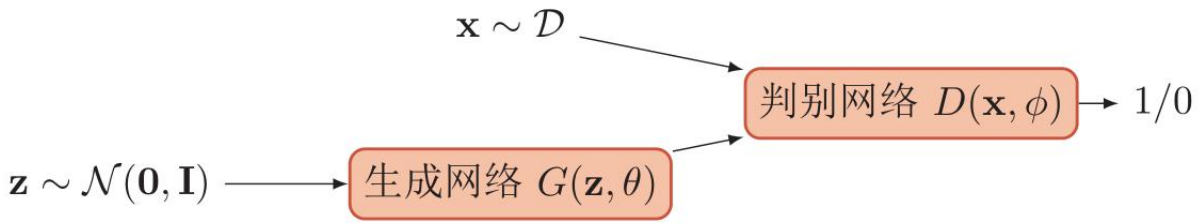
如何学习生成网络?

生成网络示例


0.1
−3
2.4
0.9
生成网络
Each dimension of input vector represents some characteristics.
3
−3
2.4
0.9
生成网络
Longer hair
0.1
2.1
2.4
0.9
生成网络
blue hair
0.1
−3
2.4
3.5
生成网络
Open mouth

判别网络

- 判别网络的输入则为真实样本或生成网络的输出，其目的是将生成网络的输出从真实样本中尽可能分辨出来。



MinMax Game

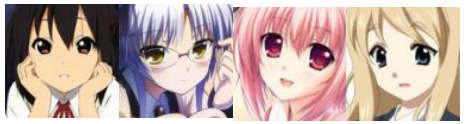
- 对抗训练
  - 生成网络要尽可能地欺骗判别网络。
  - 判别网络将生成网络生成的样本与真实样本中尽可能区分出来。

- 两个网络相互对抗、不断调整参数，最终目的是使判别网络无法判断生成网络的输出结果是否真实。

对抗过程

生成网络
(student)

判别网络
(teacher)



生成网络
v1

No eyes

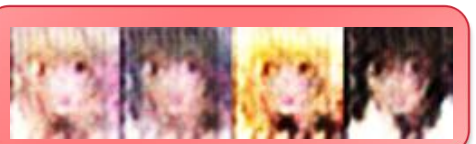
判别网络
v1

生成网络
v2

No mouth

判别网络
v2

生成网络
v3



MinMax Game

- 判别网络

$$\max_{\phi} \mathbb{E}_{\boldsymbol{x}\sim p_r(\boldsymbol{x})}\Big[\log D(\boldsymbol{x};\phi)\Big] + \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}\Big[\log(1 - D\Big(G(\boldsymbol{z};\theta);\phi\Big)\Big]$$

- 生成网络

$$\max_{\theta}\left(\mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}\Big[\log D\Big(G(\boldsymbol{z};\theta);\phi\Big)\Big]\right)$$

- Minimax Game

$$\min_{\theta}\max_{\phi}\left(\mathbb{E}_{\boldsymbol{x}\sim p_r(\boldsymbol{x})}\Big[\log D(\boldsymbol{x};\phi)\Big] + \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})}\Big[\log(1 - D\Big(G(\boldsymbol{z};\theta);\phi\Big)\Big]\right)$$

训练过程

算法 13.1: 生成对抗网络的训练过程

输入: 训练集 $\mathcal{D}$，对抗训练迭代次数 T，每次判别网络的训练迭代次数 K，小批量样本数量 M

随机初始化 θ, ϕ;

for $t \leftarrow 1$ **to** T **do**

  // 训练判别网络 $D(\mathbf{x}, \phi)$

  for $k \leftarrow 1$ **to** K **do**

    // 采集小批量训练样本

    从训练集 $\mathcal{D}$ 中采集 M 个样本 $\{\mathbf{x}^{(m)}\}, 1 \leq m \leq M$;

    从分布 $\mathcal{N}(\mathbf{0}, \mathbf{I})$ 中采集 M 个样本 $\{\mathbf{z}^{(m)}\}, 1 \leq m \leq M$;

    使用随机梯度上升更新 ϕ，梯度为

$$\frac{\partial}{\partial \phi}\left[\frac{1}{M}\sum_{m=1}^{M}\left(\log D(\mathbf{x}^{(m)}, \phi) + \log\left(1 - D\left(G(\mathbf{z}^{(m)}, \theta), \phi\right)\right)\right)\right];$$

  end

  // 训练生成网络 $G(\mathbf{z}, \theta)$

  从分布 $\mathcal{N}(\mathbf{0}, \mathbf{I})$ 中采集 M 个样本 $\{\mathbf{z}^{(m)}\}, 1 \leq m \leq M$;

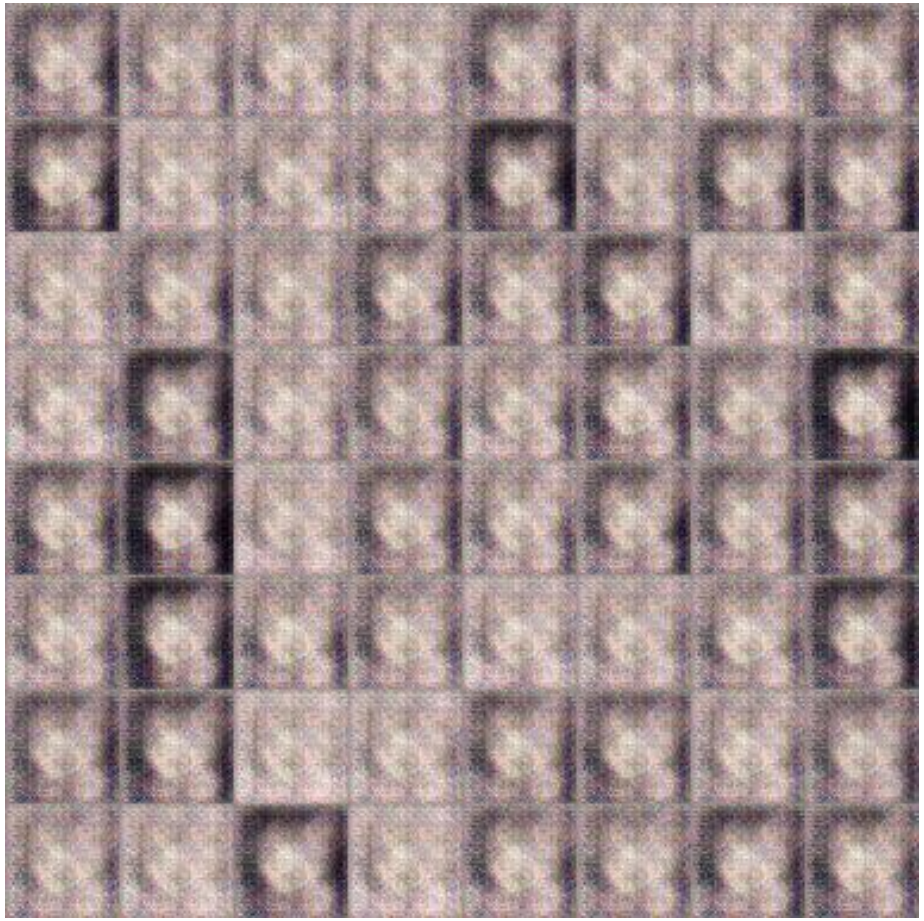
  使用随机梯度上升更新 θ，梯度为

$$\frac{\partial}{\partial \theta}\left[\frac{1}{M}\sum_{m=1}^{M} D\left(G(\mathbf{z}^{(m)}, \theta), \phi\right)\right];$$

end

输出: 生成网络 $G(\mathbf{z}, \theta)$

Anime Face Generation



100 updates

1000 updates

Anime Face Generation

2000 updates

5000 updates

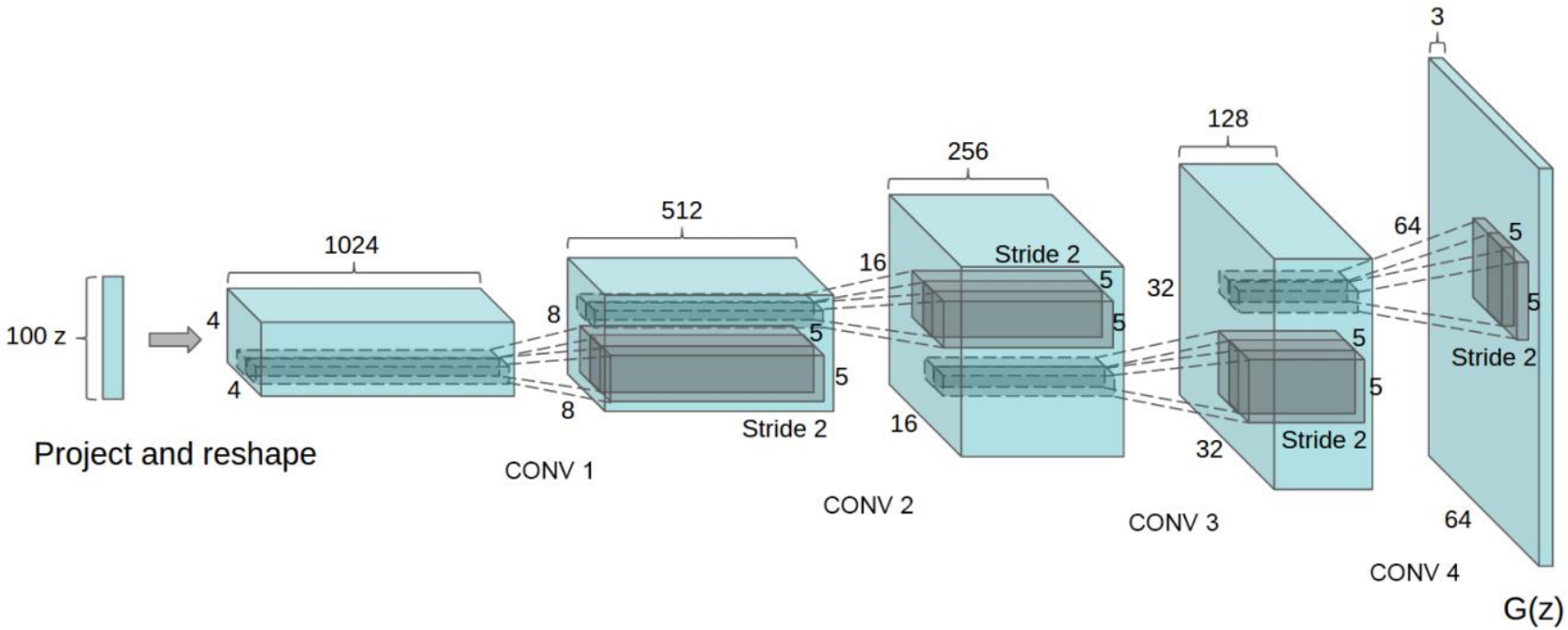
Anime Face Generation

10,000
updates

50,000
updates

一个具体的模型：DCGANs

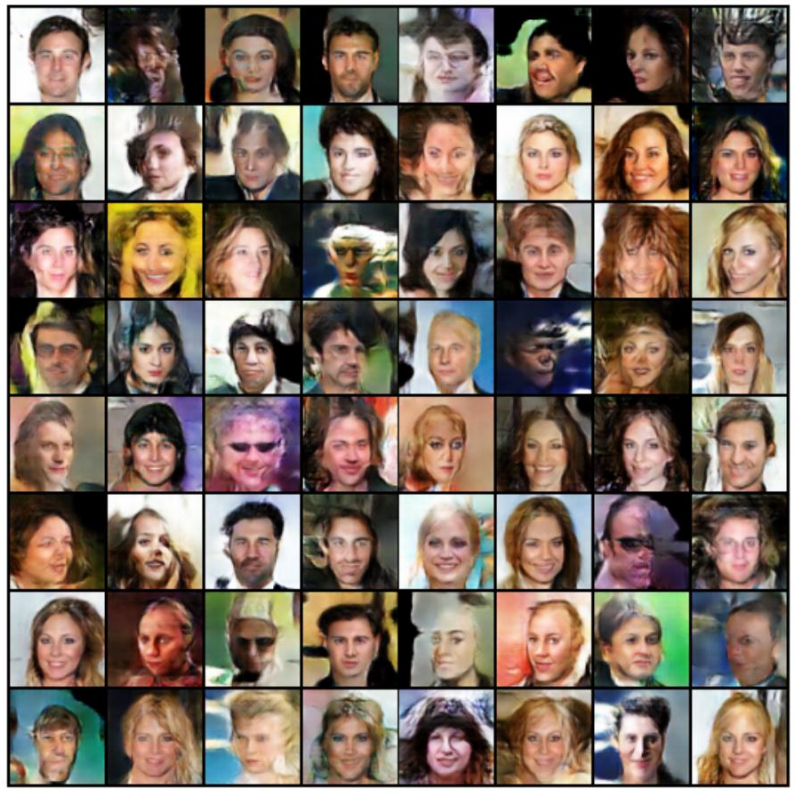
- 判别网络是一个传统的深度卷积网络，但使用了带步长的卷积来实现下采样操作，不用最大汇聚（pooling）操作。
- 生成网络使用一个特殊的深度卷积网络来实现使用微步卷积来生成64×63大小的图像。



DCGANs

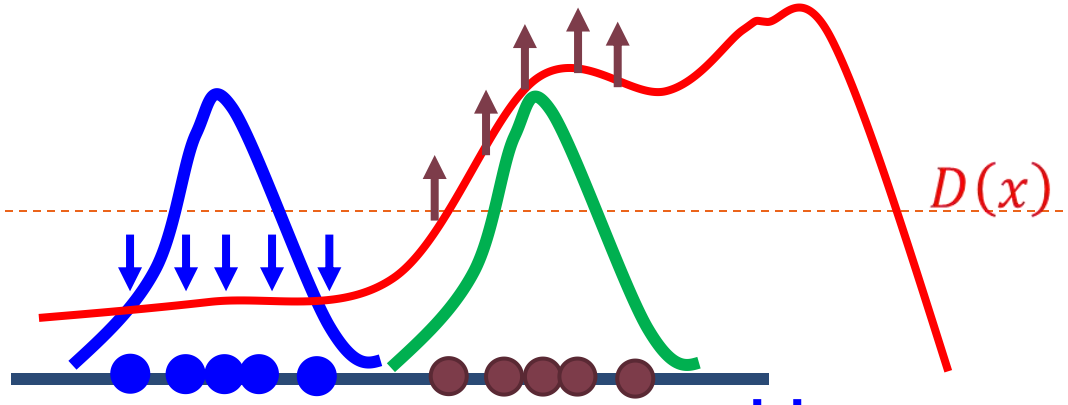
Real Images

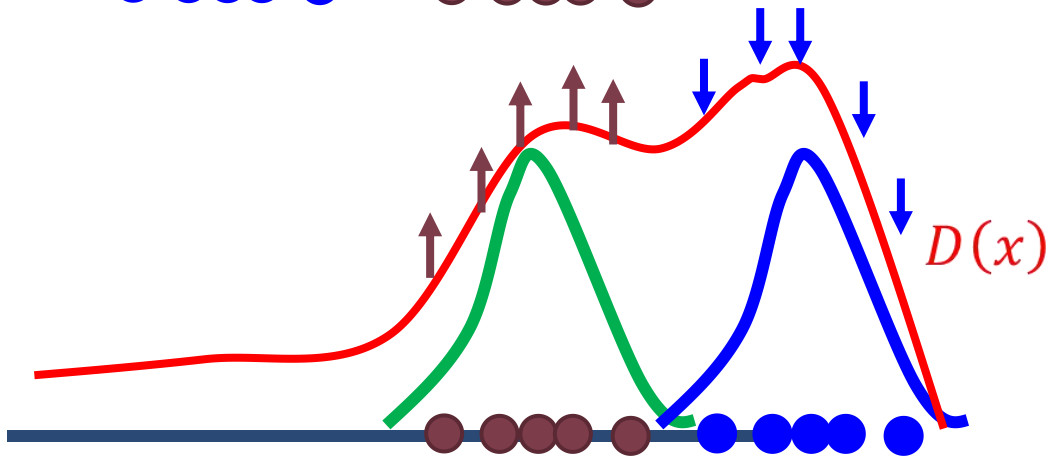
Fake Images



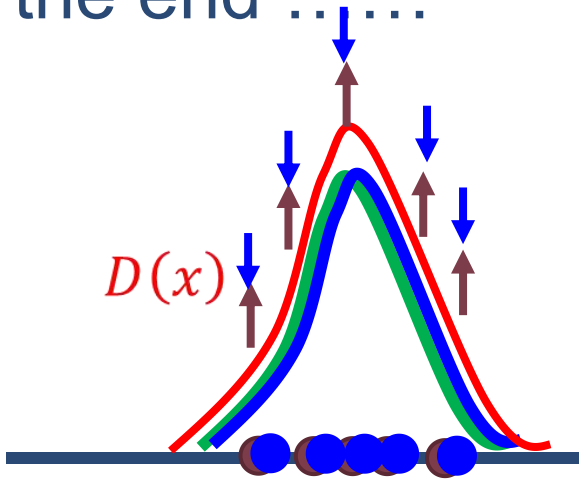
模型分析

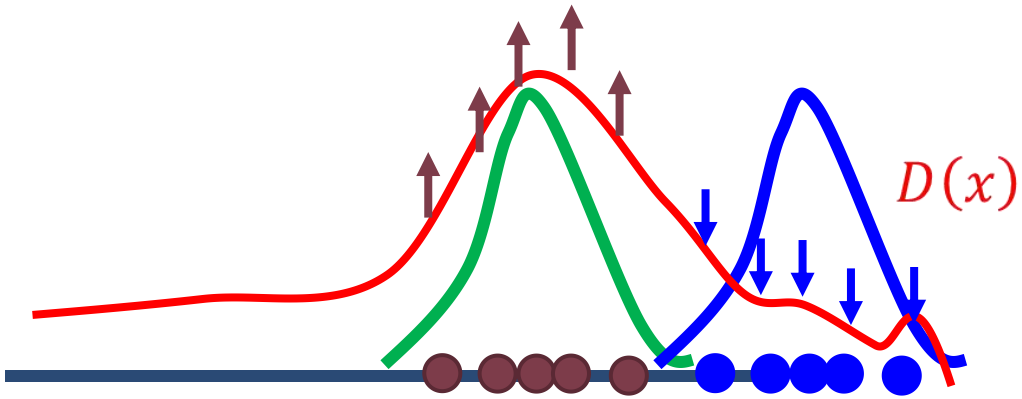
数据分布





In the end ……





模型分析

- 假设$p_r(x)$和$p_\theta(x)$已知，则最优的判别器为

$$D^\star(\boldsymbol{x}) = \frac{p_r(\boldsymbol{x})}{p_r(\boldsymbol{x}) + p_\theta(\boldsymbol{x})}$$

- 目标函数变为

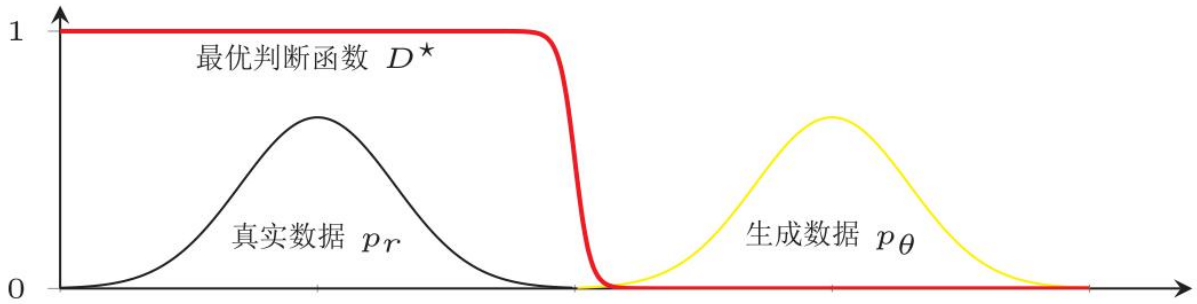
$$\begin{aligned}
\mathcal{L}(G|D^\star) &= \mathbb{E}_{\boldsymbol{x}\sim p_r(\boldsymbol{x})}\Big[\log D^\star(\boldsymbol{x})\Big] + \mathbb{E}_{\boldsymbol{x}\sim p_\theta(\boldsymbol{x})}\Big[\log(1 - D^\star(\boldsymbol{x}))\Big] \\
&= \mathbb{E}_{\boldsymbol{x}\sim p_r(\boldsymbol{x})}\left[\log \frac{p_r(\boldsymbol{x})}{p_r(\boldsymbol{x}) + p_\theta(\boldsymbol{x})}\right] + \mathbb{E}_{\boldsymbol{x}\sim p_\theta(\boldsymbol{x})}\left[\log \frac{p_\theta(\boldsymbol{x})}{p_r(\boldsymbol{x}) + p_\theta(\boldsymbol{x})}\right] \\
&= D_{\mathrm{KL}}\Big(p_r \| p_a\Big) + D_{\mathrm{KL}}\Big(p_\theta \| p_a\Big) - 2\log 2 \\
&= 2D_{\mathrm{JS}}(p_r \| p_\theta) - 2\log 2,
\end{aligned}$$

不稳定性：生成网络的梯度消失

$$\min_{\theta}\max_{\phi}\left(\mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\log D(\mathbf{x},\phi)\right]+\mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}\left[\log(1-D\Big(G(\mathbf{z},\theta),\phi\Big)\right]\right)$$

在生成对抗网络中，当判断网络为最优时，生成网络的优化目标是最小化真实分布$p_r(x)$和模型分布$p_\theta(x)$之间的JS散度。
当两个分布相同时，JS散度为0，最优生成网络对应的损失为-2log2。

使用JS散度来训练生成对抗网络的一个问题是当两个分布没有重叠时，它们之间的JS散度恒等于常数log2。对生成网络来说，目标函数关于参数的梯度为0。



模型坍塌：生成网络的“错误”目标

- 生成网络的目标函数

$$
\begin{aligned}
\mathcal{L}'(G|D^\star) &= \mathbb{E}_{\boldsymbol{x}\sim p_\theta(\boldsymbol{x})}\Big[\log D^\star(\boldsymbol{x})\Big] \\
&= \mathbb{E}_{\boldsymbol{x}\sim p_\theta(\boldsymbol{x})}\left[\log \frac{p_r(\boldsymbol{x})}{p_r(\boldsymbol{x})+p_\theta(\boldsymbol{x})}\cdot\frac{p_\theta(\boldsymbol{x})}{p_\theta(\boldsymbol{x})}\right] \\
&= -\mathbb{E}_{\boldsymbol{x}\sim p_\theta(\boldsymbol{x})}\left[\log \frac{p_\theta(\boldsymbol{x})}{p_r(\boldsymbol{x})}\right] + \mathbb{E}_{\boldsymbol{x}\sim p_\theta(\boldsymbol{x})}\left[\log \frac{p_\theta(\boldsymbol{x})}{p_r(\boldsymbol{x})+p_\theta(\boldsymbol{x})}\right] \\
&= -D_{\mathrm{KL}}\Big(p_\theta\|p_r\Big) + \mathbb{E}_{\boldsymbol{x}\sim p_\theta(\boldsymbol{x})}\Big[\log\big(1-D^\star(\boldsymbol{x})\big)\Big] \\
&= -D_{\mathrm{KL}}\Big(p_\theta\|p_r\Big) + 2D_{\mathrm{JS}}(p_r\|p_\theta) - 2\log 2 - \mathbb{E}_{\boldsymbol{x}\sim p_r(\boldsymbol{x})}\Big[\log D^\star(\boldsymbol{x})\Big]
\end{aligned}
$$

- 其中后两项和生成网络无关，因此

$$
\arg\max_\theta \mathcal{L}'(G|D^\star) = \arg\min_\theta D_{\mathrm{KL}}\Big(p_\theta\|p_r\Big) - 2D_{\mathrm{JS}}(p_r\|p_\theta)
$$

前向和逆向KL散度

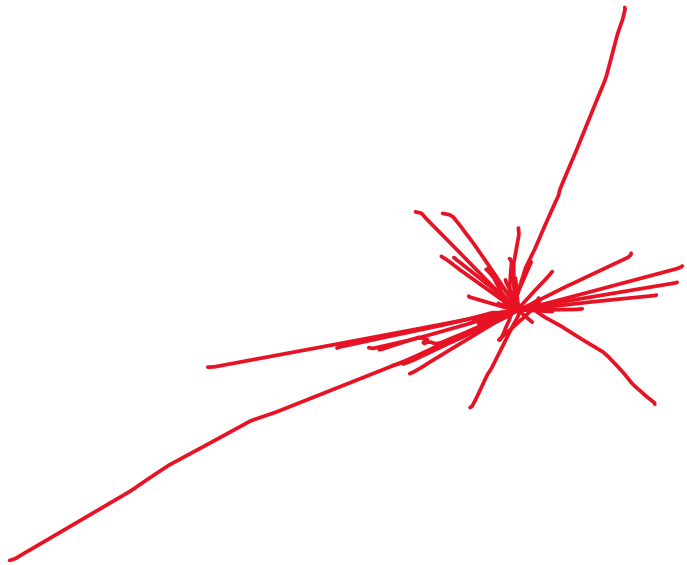
- KL散度是一种非对称的散度，在计算真实分布$p_r(x)$和模型分布$p_\theta(x)$之间的KL散度时，按照顺序不同，有两种KL散度：

前向KL散度

$$D_{\mathrm{KL}}(p_r \| p_\theta) = \int p_r(\boldsymbol{x}) \log \frac{p_r(\boldsymbol{x})}{p_\theta(\boldsymbol{x})} d\boldsymbol{x}$$

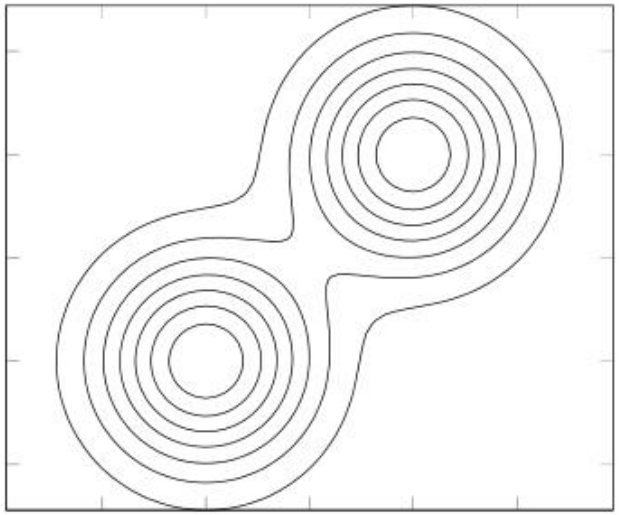
逆向KL散度

$$D_{\mathrm{KL}}(p_\theta \| p_r) = \int p_\theta(\boldsymbol{x}) \log \frac{p_\theta(\boldsymbol{x})}{p_r(\boldsymbol{x})} d\boldsymbol{x}$$

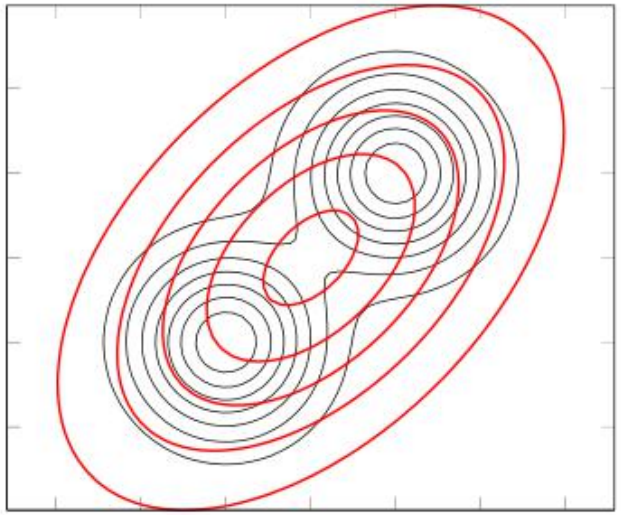


前向和逆向KL散度

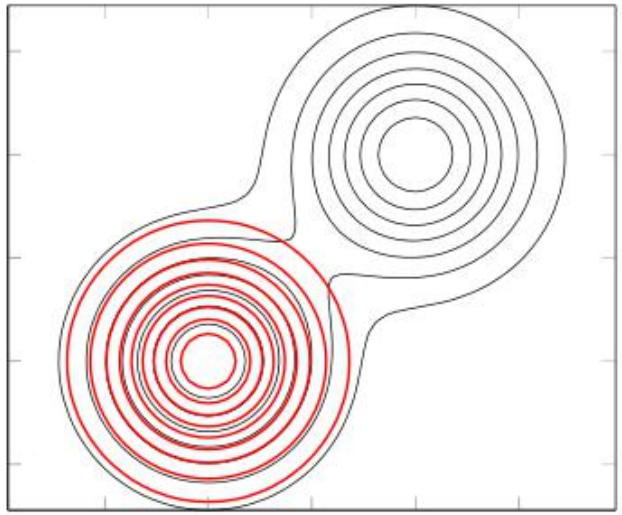
真实分布 p_r



前向 KL 散度 $D_{KL}(p_r\|p_\theta)$



逆向 KL 散度 $D_{KL}(p_\theta\|p_r)$

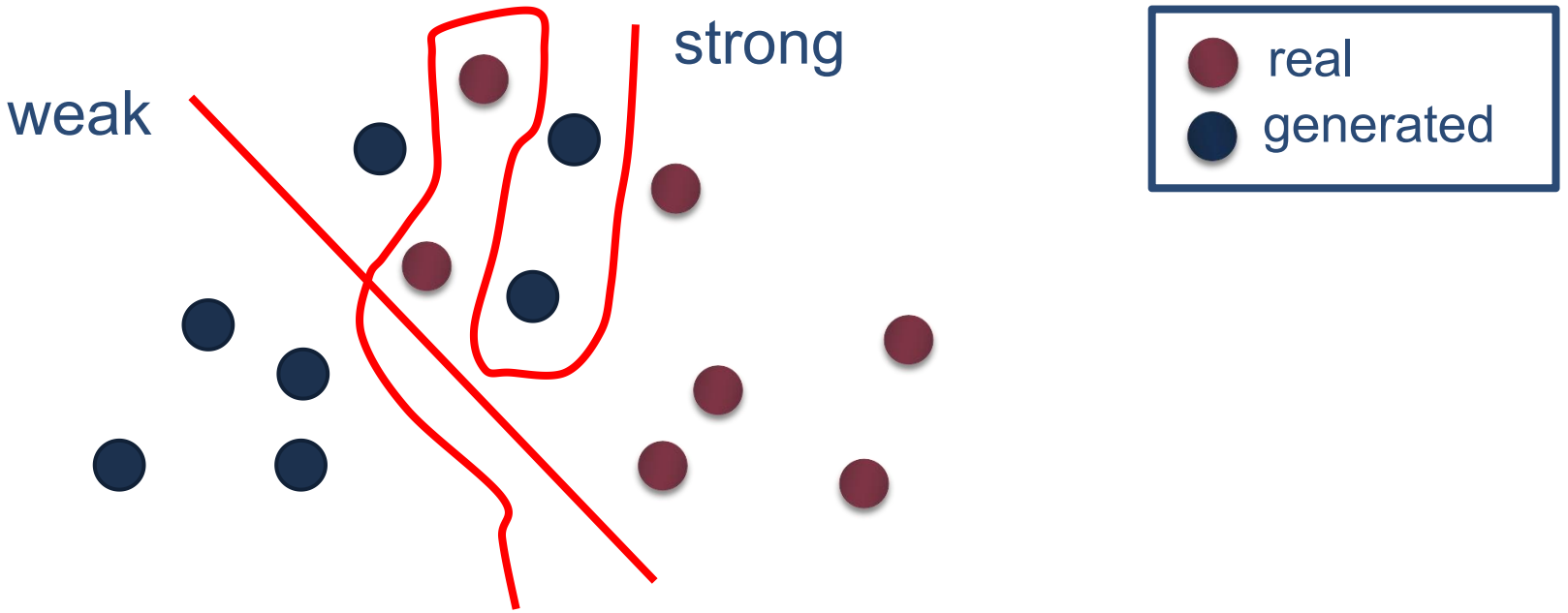


$$D_{\mathrm{KL}}(p_r\|p_\theta)=\int p_r(\boldsymbol{x})\log\frac{p_r(\boldsymbol{x})}{p_\theta(\boldsymbol{x})}d\boldsymbol{x}$$

$$D_{\mathrm{KL}}(p_\theta\|p_r)=\int p_\theta(\boldsymbol{x})\log\frac{p_\theta(\boldsymbol{x})}{p_r(\boldsymbol{x})}d\boldsymbol{x}$$

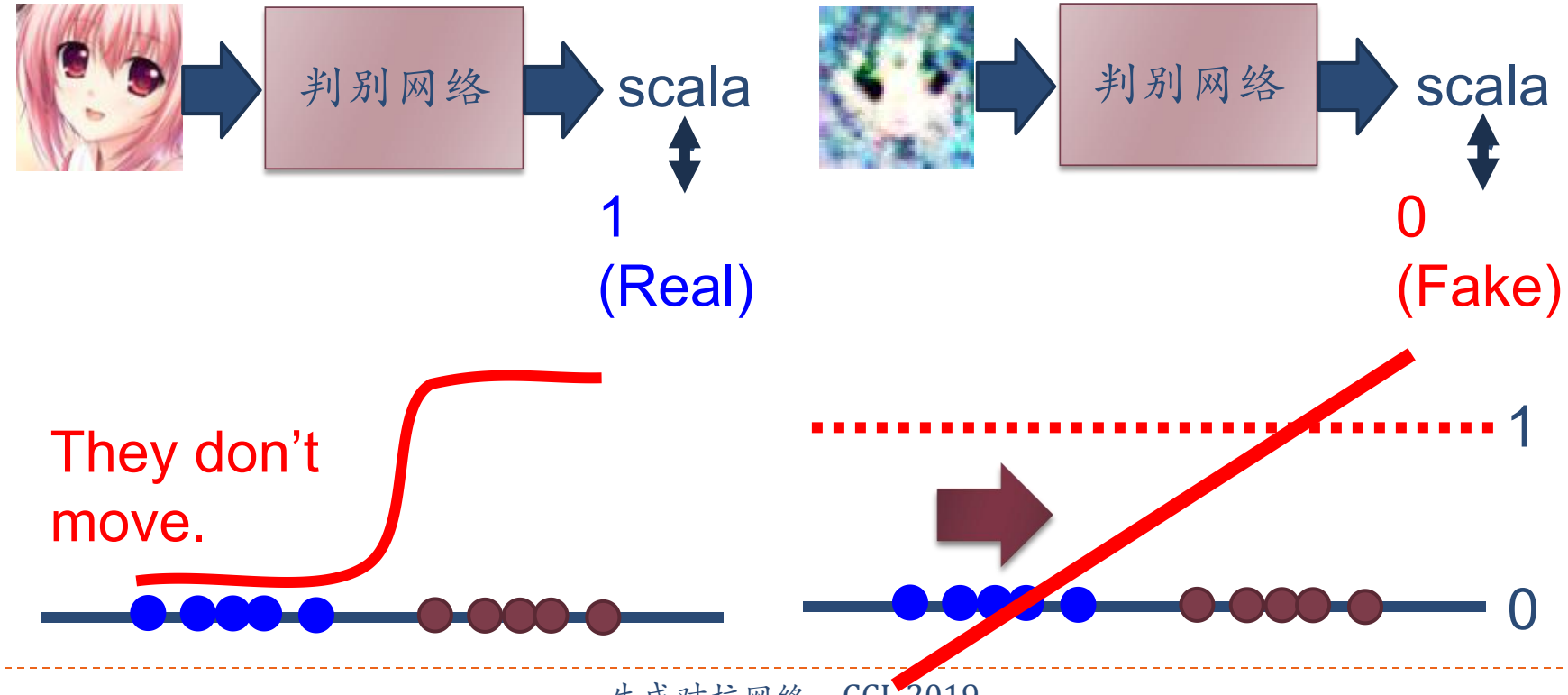
改进

- 弱化判别器
- 使用更好的损失函数



Least Square GAN (LSGAN)

- Replace sigmoid with linear (replace classification with regression)



f-GAN

- f-divergences $D_f(P||Q)$

Name	$D_f(P\|Q)$	Generator $f(u)$	$T^*(x)$
Kullback-Leibler	$\int p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x$	$u \log u$	$1 + \log \frac{p(x)}{q(x)}$
Reverse KL	$\int q(x) \log \frac{q(x)}{p(x)} \mathrm{d}x$	$-\log u$	$-\frac{q(x)}{p(x)}$
Pearson χ^2	$\int \frac{(q(x)-p(x))^2}{p(x)} \mathrm{d}x$	$(u-1)^2$	$2(\frac{p(x)}{q(x)} - 1)$
Squared Hellinger	$\int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 \mathrm{d}x$	$(\sqrt{u}-1)^2$	$(\sqrt{\frac{p(x)}{q(x)}} - 1) \cdot \sqrt{\frac{q(x)}{p(x)}}$
Jensen-Shannon	$\frac{1}{2} \int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \mathrm{d}x$	$-(u+1) \log \frac{1+u}{2} + u \log u$	$\log \frac{2p(x)}{p(x)+q(x)}$
GAN	$\int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \mathrm{d}x - \log(4)$	$u \log u - (u+1) \log(u+1)$	$\log \frac{p(x)}{p(x)+q(x)}$

f-GAN

- 目标函数

$$F(\theta, \omega) = \mathbb{E}_{x \sim P}\left[T_\omega(x)\right] - \mathbb{E}_{x \sim Q_\theta}\left[f^*(T_\omega(x))\right]$$

- f-divergences

Name	Output activation g_f	dom_{f^*}	Conjugate $f^*(t)$	$f'(1)$
Kullback-Leibler (KL)	v	$\mathbb{R}$	$\exp(t-1)$	1
Reverse KL	$-\exp(-v)$	$\mathbb{R}_-$	$-1-\log(-t)$	-1
Pearson χ^2	v	$\mathbb{R}$	$\frac{1}{4}t^2+t$	0
Squared Hellinger	$1-\exp(-v)$	$t<1$	$\frac{t}{1-t}$	0
Jensen-Shannon	$\log(2)-\log(1+\exp(-v))$	$t<\log(2)$	$-\log(2-\exp(t))$	0
GAN	$-\log(1+\exp(-v))$	$\mathbb{R}_-$	$-\log(1-\exp(t))$	$-\log(2)$

Wasserstein GAN

Wasserstein距离

- Wasserstein距离用于衡量两个分布之间的距离。

$$W_p(q_1, q_2) = \left(\inf_{\gamma(x,y) \in \Gamma(q_1, q_2)} \mathbb{E}_{(x,y)\sim\gamma(x,y)}[d(x,y)^p] \right)^{\frac{1}{p}}$$

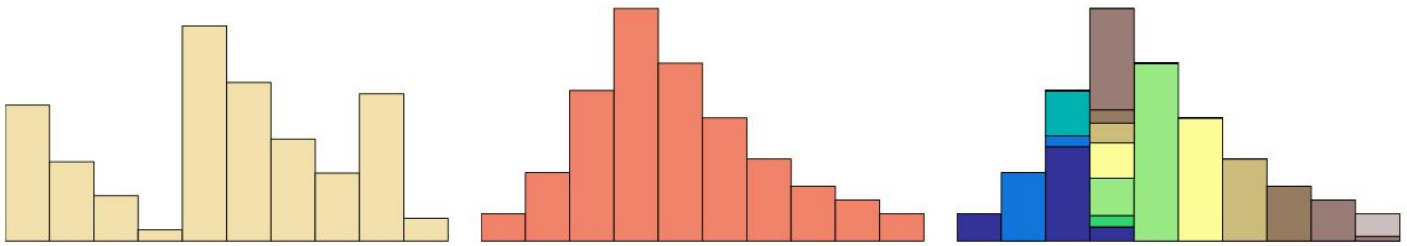
  - 其中$\Gamma(q_1, q_2)$是边际分布为q_1, q_2的所有可能的联合分布集合，d(x,y)为x和y的距离，比如ℓ_p距离等。

- Wasserstein距离相比KL散度和JS散度的优势在于：即使两个分布没有重叠或者重叠非常少，Wasserstein距离仍然能反映两个分布的远近。

Wasserstein距离

- 如果将两个分布看作是两个土堆，联合分布γ(x,y)看作是从土堆q_1的位置x到土堆q_2的位置y的搬运土的数量。

- Wasserstein距离可以理解为搬运土堆的最小工作量，也称为推土机距离（Earth-Mover's Distance，EMD）。



(a) $q_1(x)$　(b) $q_2(x)$　(c) q_1 到 q_2 最优的运输方案

Kantorovich-Rubinstein 对偶定理

This formula is highly intractable

$$\mathrm{EMD}(P_r, P_\theta) = \inf_{\gamma \in \Pi(P_r, P_\theta)} \sum_{x,y} \|x - y\| \gamma(x, y) = \inf_{\gamma \in \Pi(P_r, P_\theta)} \mathbb{E}_{(x,y)\sim\gamma} \|x - y\|$$

Kantorovich-Rubinstein Duality

$$\mathrm{EMD}(P_r, P_\theta) = \sup_{\|f\|_L \leq 1} \mathbb{E}_{x\sim P_r} f(x) - \mathbb{E}_{x\sim P_\theta} f(x).$$

1-Lipschitz Constraint

Lipschitz连续函数

数学小知识 | Lipschitz 连续函数

在数学中，对于一个实数函数 $f : \mathbb{R} \to \mathbb{R}$，如果满足函数曲线上任意两点连线的斜率一致有界，即任意两点的斜率都小于常数 $K > 0$，

$$|f(x_1) - f(x_2)| \leq K|x_1 - x_2|, \tag{13.54}$$

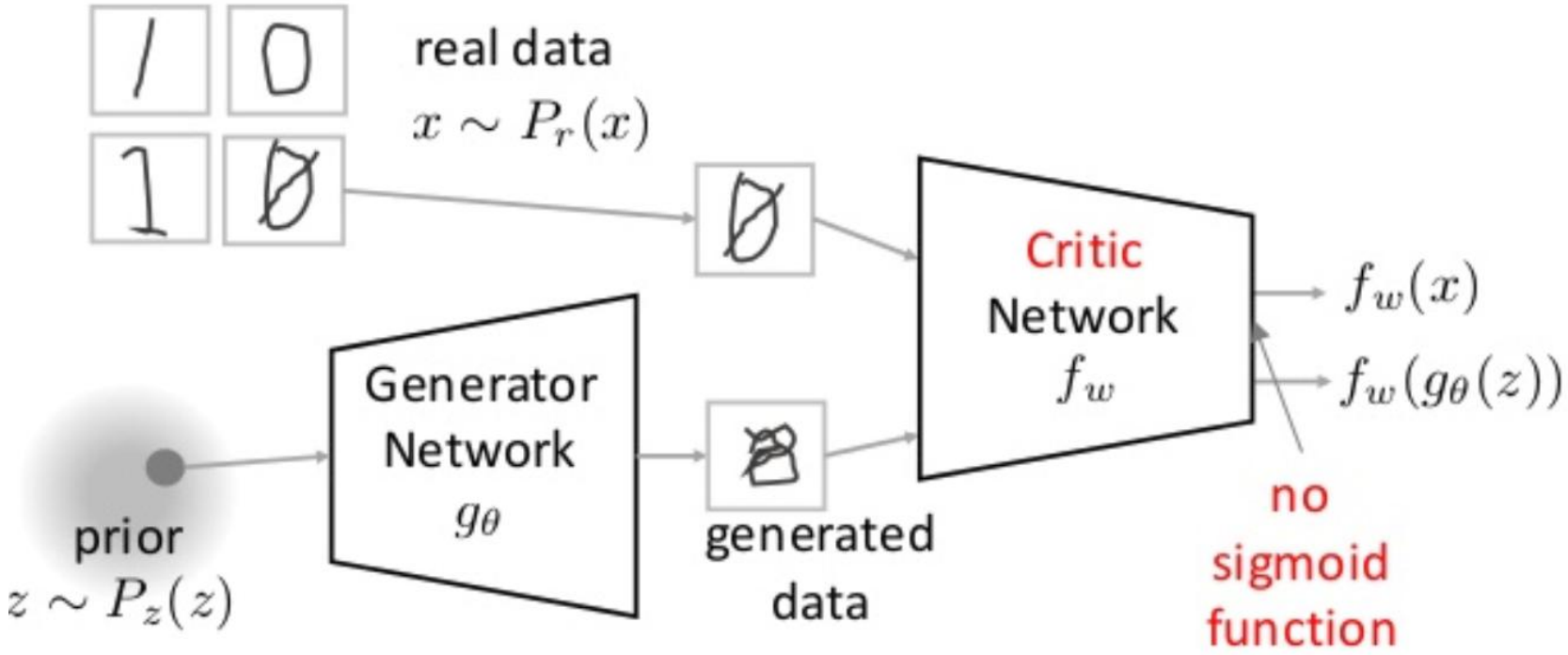
则函数 f 就称为 *K-Lipschitz 连续函数*，K 称为Lipschitz 常数。

Lipschitz连续要求函数在无限的区间上不能有超过线性的增长。如果一个函数可导，并满足Lipschitz连续，那么导数有界。如果一个函数可导，并且导数有界，那么函数为Lipschitz连续。

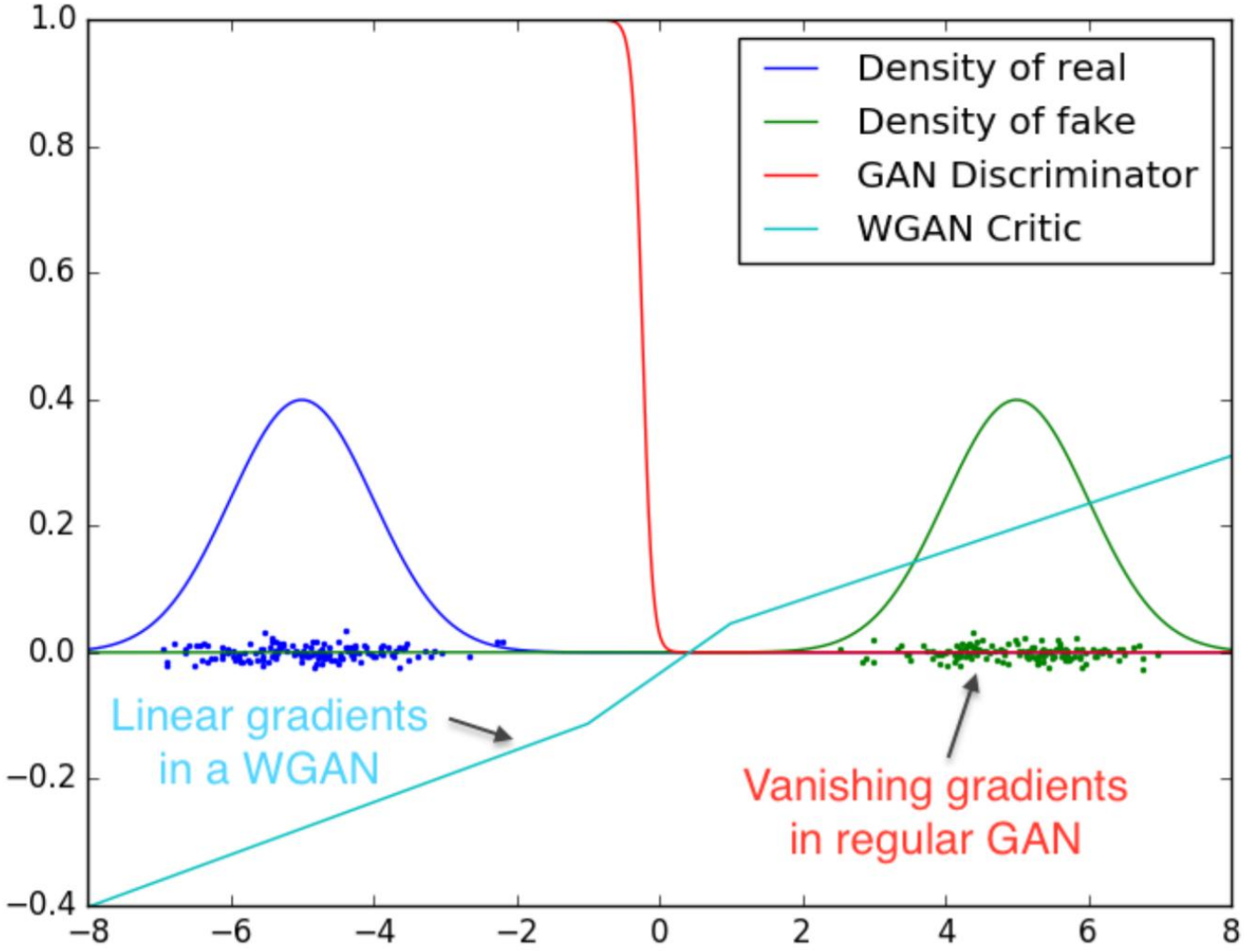
Wasserstein GAN

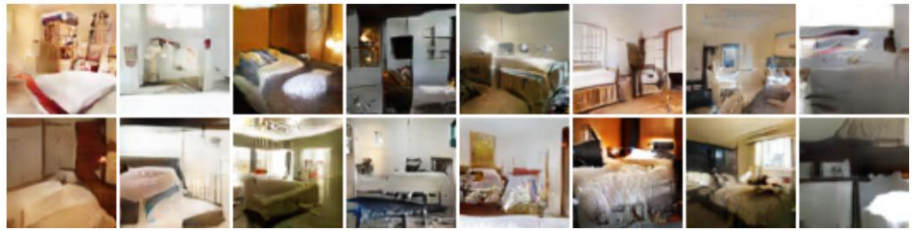
$$\min_{\theta} \max_{w \in [-k,k]^l} \mathbb{E}_{x \sim P_r}[f_w(x)] - \mathbb{E}_{z \sim P_z}[f_w(g_\theta(z))]$$

k-Lipschitz Constraint

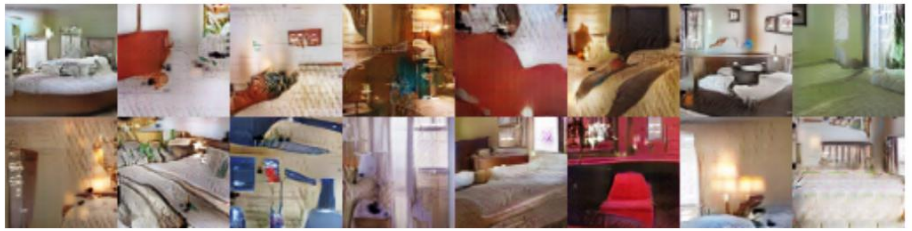


梯度问题


Density of real
Density of fake
GAN Discriminator
WGAN Critic
Linear gradients
in a WGAN
Vanishing gradients
in regular GAN
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−8
−6
−4
−2
0
2
4
6
8

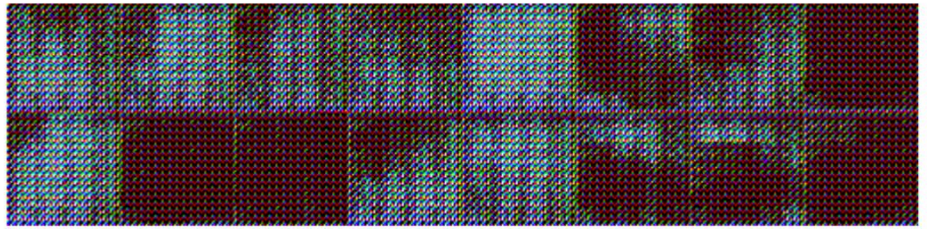


WGAN



DCGAN

batch normalization
constant number of filters at every layer



DCGAN

LSGAN

Original WGAN

Improved WGAN

G: CNN, D: CNN

weight clipping

gradient penalty

G: CNN (no normalization), D: CNN (no normalization)

G: CNN (tanh), D: CNN(tanh)

DCGAN

LSGAN

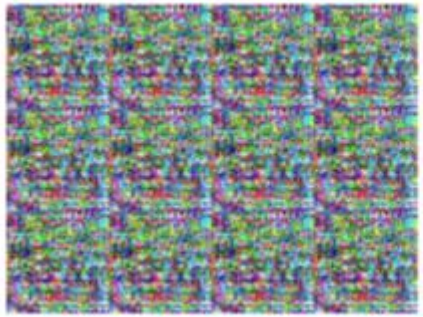
Original WGAN

weight clipping

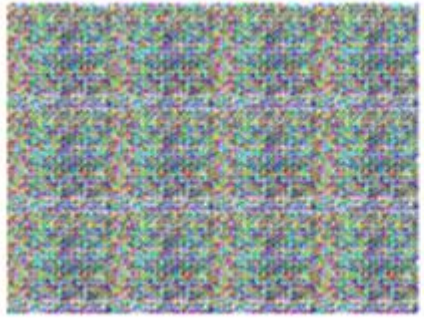
Improved WGAN

gradient penalty

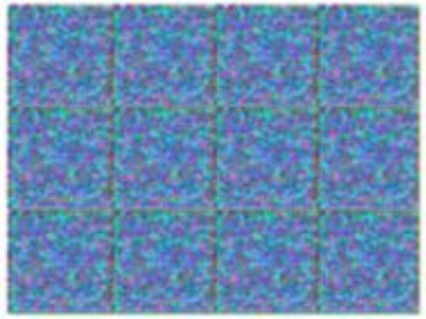
G: MLP, D: CNN



G: CNN (bad structure), D: CNN



G: 101 layer, D: 101 layer



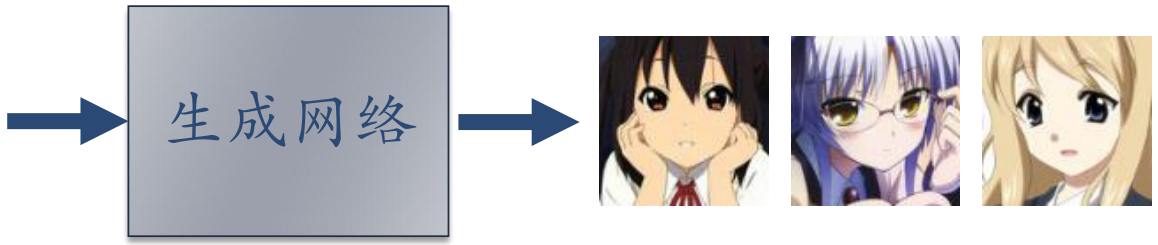
GAN的扩展

条件生成

- 根据条件针对性的生成数据

$$\begin{bmatrix}0.3\\-0.1\\\vdots\\-0.7\end{bmatrix}\begin{bmatrix}0.1\\-0.1\\\vdots\\0.7\end{bmatrix}\begin{bmatrix}-0.3\\0.1\\\vdots\\0.9\end{bmatrix}$$

In a specific range



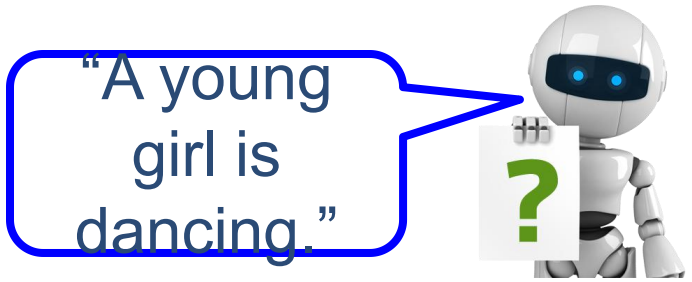
“Girl with red hair and red eyes”

“Girl with yellow ribbon”

条件生成

Caption Generation

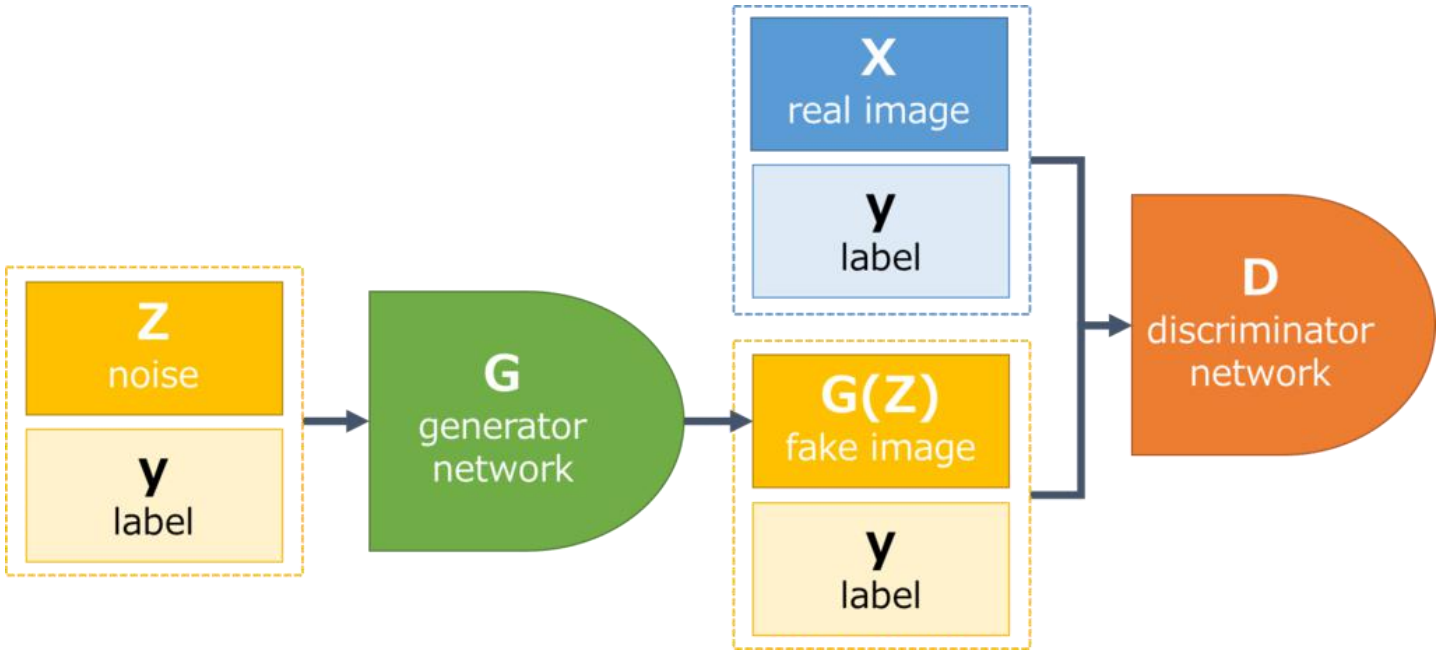
Given condition:

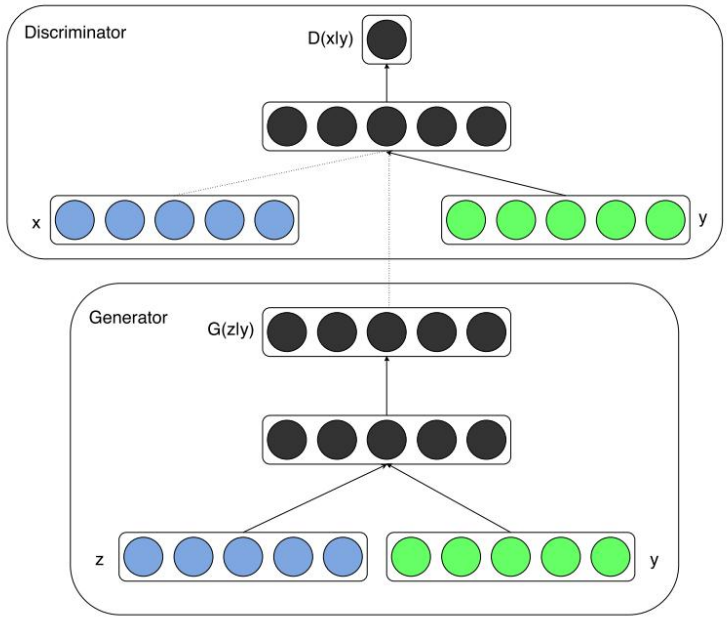


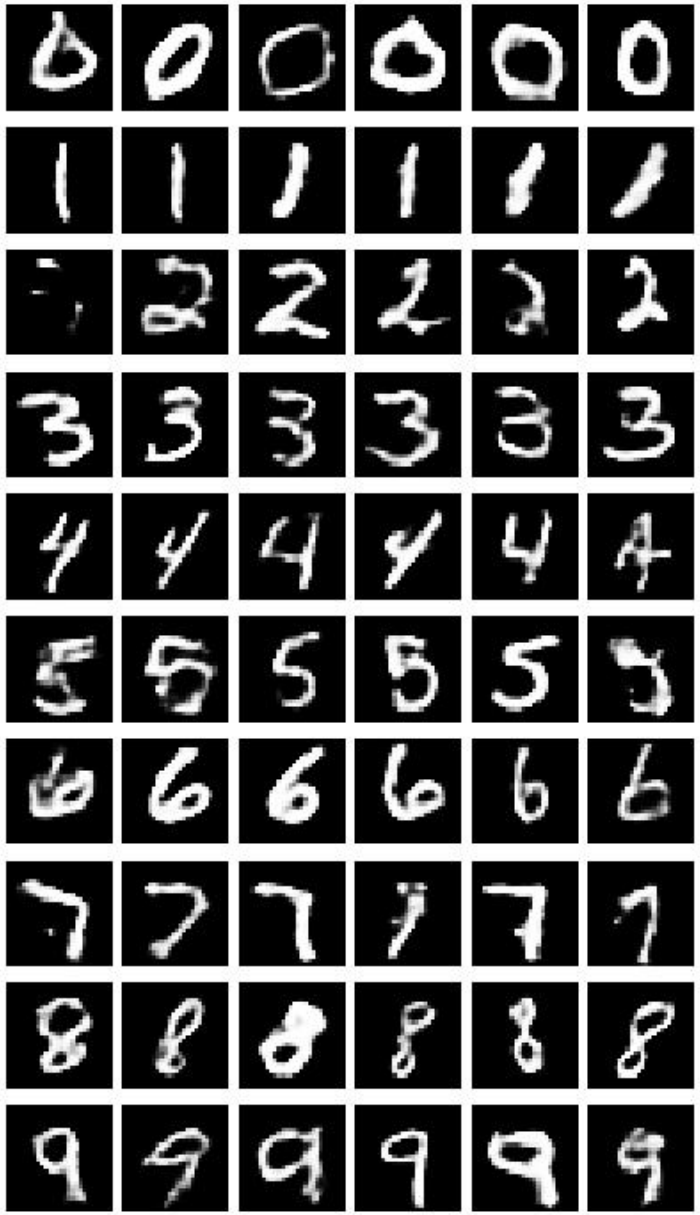
Chat-bot

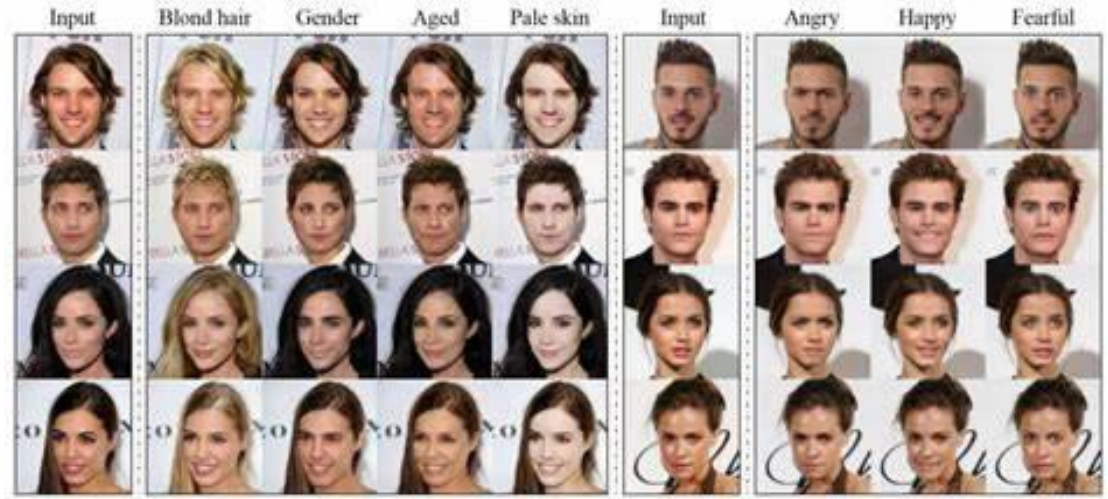
Given condition:

Conditional GAN

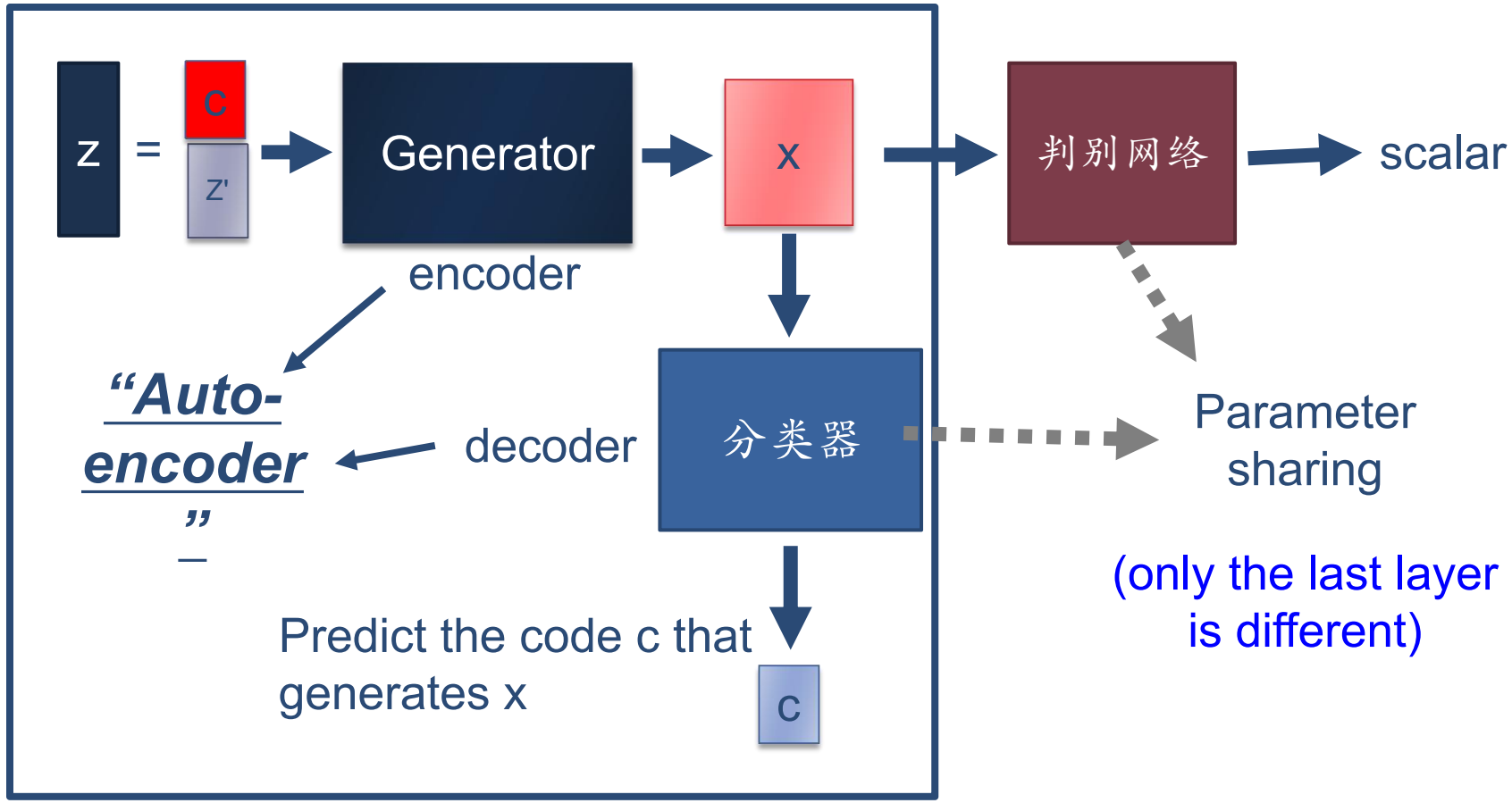


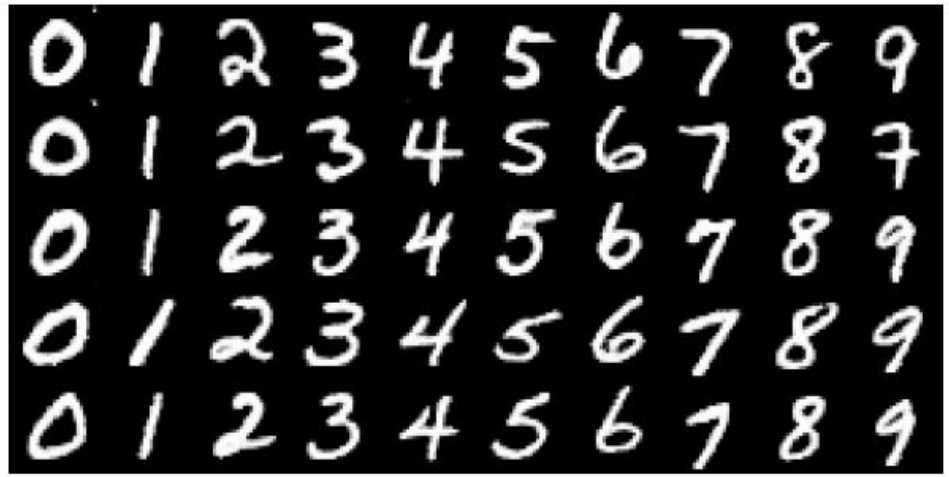




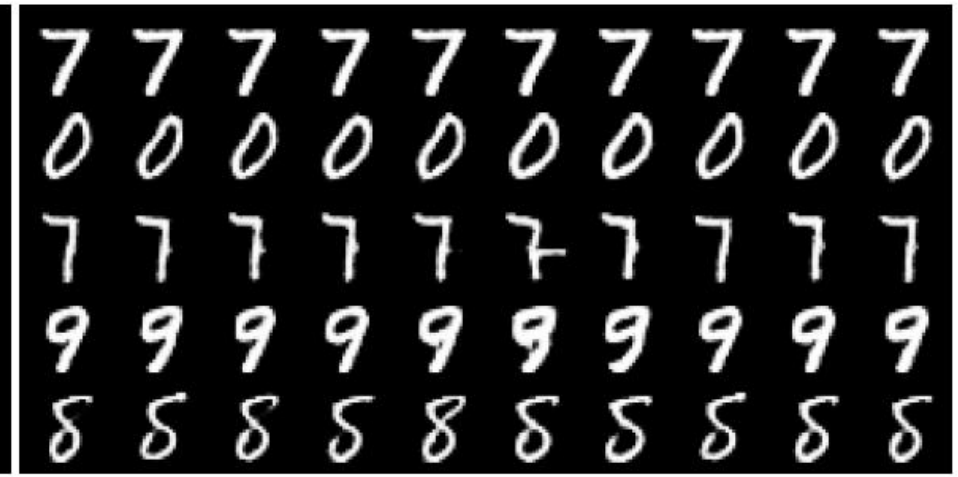

Input
Blond hair
Gender
Aged
Pale skin
Input
Angry
Happy
Fearful

InfoGAN

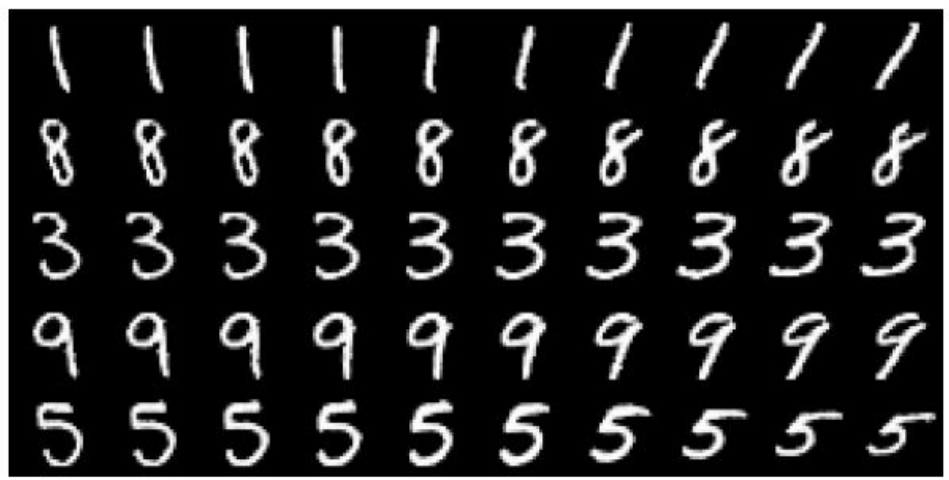

z
=
c
z'
Generator
x
判别网络
scalar
encoder
"Auto-encoder"
decoder
分类器
Parameter sharing
(only the last layer is different)
Predict the code c that generates x
c



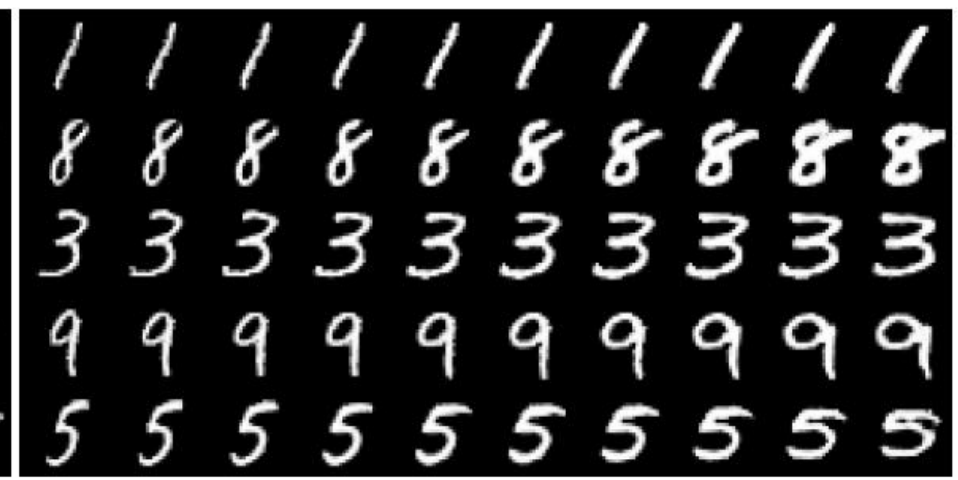
(a) Varying c_1 on InfoGAN (Digit type)



(b) Varying c_1 on regular GAN (No clear meaning)



(c) Varying c_2 from -2 to 2 on InfoGAN (Rotation)



(d) Varying c_3 from -2 to 2 on InfoGAN (Width)

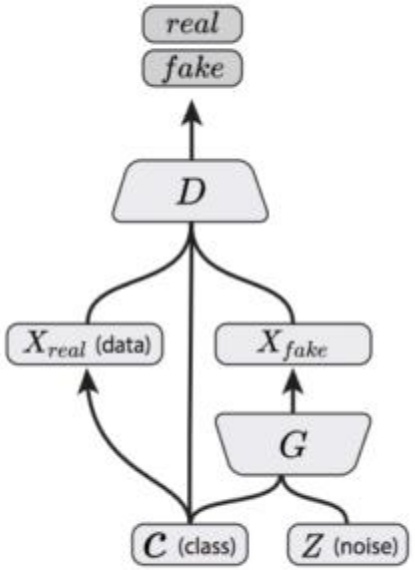
https://arxiv.org/abs/1606.03657

(a) Rotation (b) Width

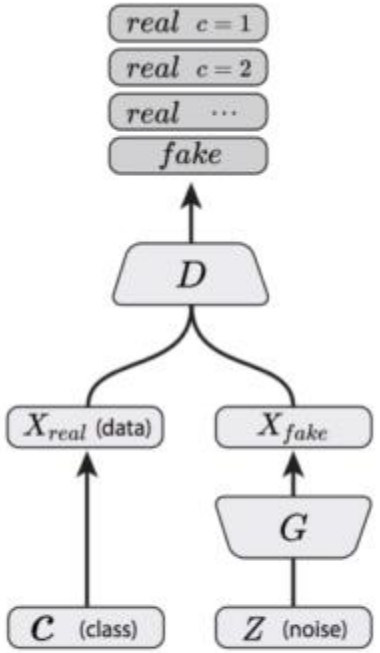
(c) Lighting (d) Wide or Narrow

https://arxiv.org/abs/1606.03657

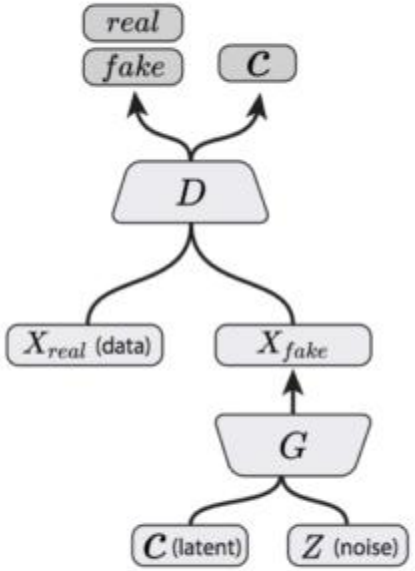
AC-GAN



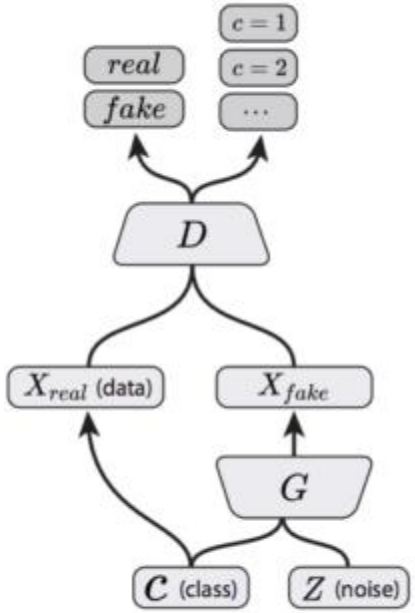
Conditional GAN
(Mirza & Osindero, 2014)



Semi-Supervised GAN
(Odena, 2016; Salimans, et al., 2016)



InfoGAN
(Chen, et al., 2016)

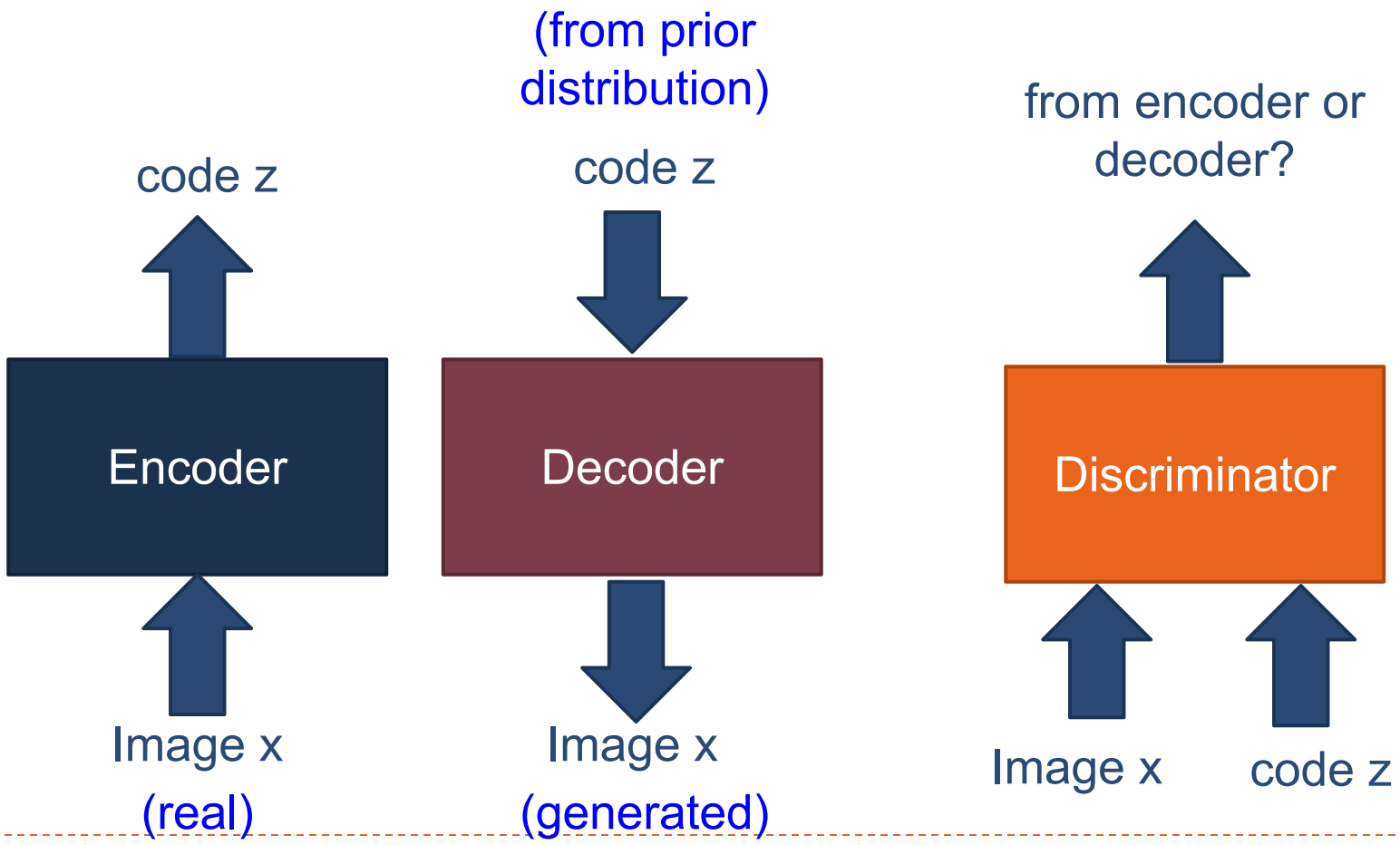


AC-GAN
(Present Work)

BiGAN

Jeff Donahue, Philipp Krähenbühl, Trevor Darrell, "Adversarial Feature Learning", ICLR, 2017

Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, Aaron Courville, "Adversarially Learned Inference", ICLR, 2017



序列生成

序列数据的潜在规律

- 以自然语言为例，

面包上涂黄油

面包上涂袜子

- 后一个句子在人脑的语义整合时需要更多的处理时间，更不符合自然语言规则。
- 规则是什么？

语言模型

- 自然语言理解 → 一个句子的可能性/合理性
  - ！ 在报那猫告做只
  - 那只猫在作报告!
  - 那个人在作报告!



- 一切都是概率!

序列概率模型

- 给定一个序列样本，其概率为

$$\mathrm{p}(x_{1:\mathrm{T}}) = \mathrm{p}(x_1, x_2, \cdots, x_{\mathrm{T}})$$

- 和一般的概率模型类似，序列概率模型有两个基本问题：
  - （1）学习问题：给定一组序列数据，估计这些数据背后的概率分布；
  - （2）生成问题：从已知的序列分布中生成新的序列样本。

序列概率模型

- 给定一个序列样本，其概率为

$$\mathrm{p}(x_{1:\mathrm{T}}) = \mathrm{p}(x_1, x_2, \cdots, x_{\mathrm{T}})$$

- 序列数据有两个特点：
  - （1）样本是变长的；
  - （2）样本空间为非常大。
- 对于一个长度为T的序列，其样本空间为$|\mathrm{V}|^T$。因此，我们很难用已知的概率模型来直接建模整个序列的概率。

序列概率模型

- 序列概率

$$\mathrm{p}(x_{1:\mathrm{T}}) = \prod_{\mathrm{t}} \mathrm{p}(x_{\mathrm{t}}|x_{1:\mathrm{t}-1})$$
$$\approx \prod_{\mathrm{t}} \mathrm{p}(x_{\mathrm{t}}|x_{\mathrm{t}-1}, \cdots, x_{\mathrm{t}-n+1}) = \prod_{\mathrm{t}} g(h_t)$$

- 因此，序列数据的概率密度估计问题可以转换为单变量的条件概率估计问题，即给定$x_{1:\mathrm{t}-1}$时x_{t}的条件概率$\mathrm{p}(x_{\mathrm{t}}|x_{1:\mathrm{t}-1})$。

自回归生成模型

给定N个序列数据$\{\mathbf{x}_{1:T_n}^{(n)}\}_{n=1}^{N}$，序列概率模型需要学习一个模型$p_\theta(x|\mathbf{x}_{1:(t-1)})$来最大化整个数据集的对数似然函数。

$$\max_\theta \sum_{n=1}^{N} \log p_\theta(\mathbf{x}_{1:T_n}^{(n)}) = \max_\theta \sum_{n=1}^{N} \sum_{t=1}^{T_n} \log p_\theta(x_t^{(n)}|\mathbf{x}_{1:(t-1)}^{(n)}). \quad (15.5)$$

- 在这种序列模型方式中，每一步都需要将前面的输出作为当前步的输入，是一种自回归（autoregressive）的方式。
  - 自回归生成模型（Autoregressive Generative Model）

序列生成

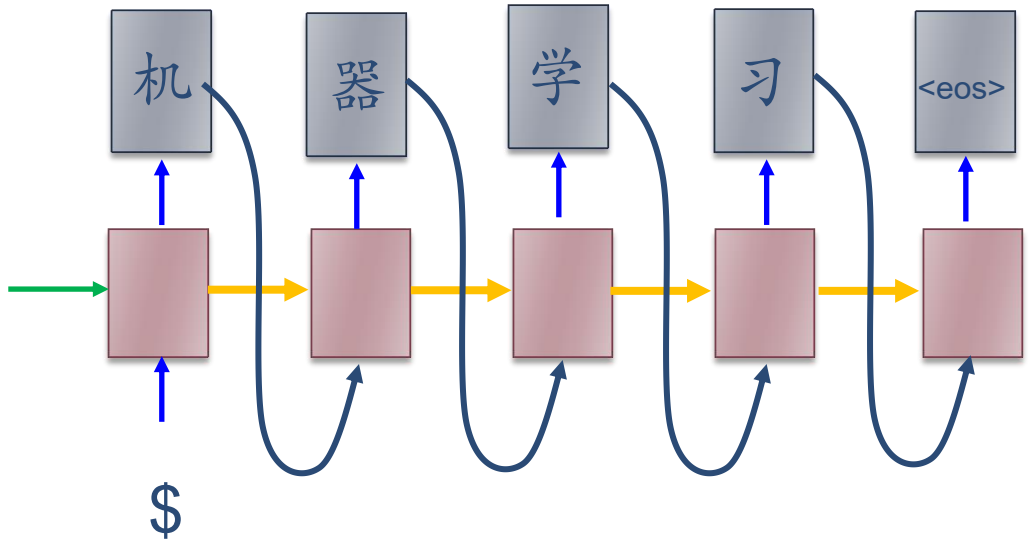
一旦通过最大似然估计训练了模型$p_\theta(x|\mathbf{x}_{1:(t-1)})$，就可以通过时间顺序来生成一个完整的序列样本。令$\hat{x}_t$为在第$t$时根据分布$p_\theta(x|\hat{\mathbf{x}}_{1:(t-1)})$生成的词，

$$\hat{x}_t \sim p_\theta(x|\hat{\mathbf{x}}_{1:(t-1)}), \tag{15.6}$$

其中$\hat{\mathbf{x}}_{1:(t-1)} = \hat{x}_1, \cdots, \hat{x}_{t-1}$为前面$t-1$步中生成的前缀序列。

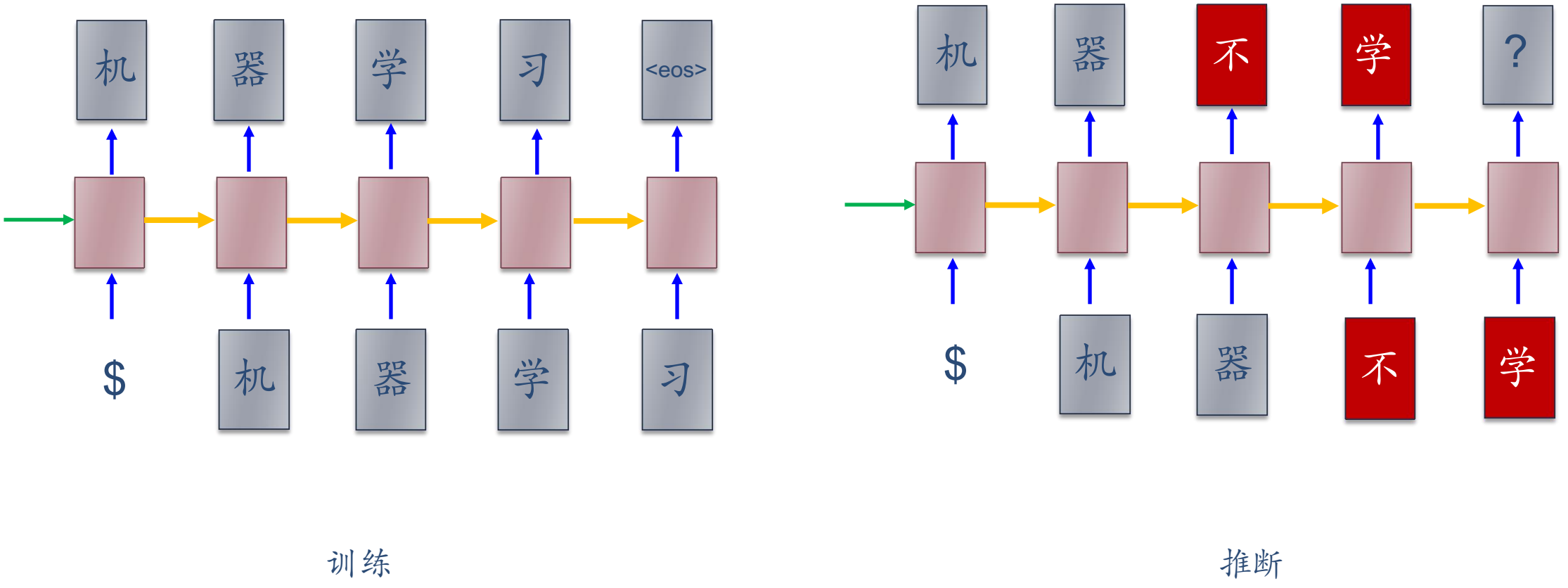
- 自回归的方式可以生成一个无限长度的序列。为了避免这种情况，通常会设置一个特殊的符号“<eos>”来表示序列的结束。在训练时，每个序列样本的结尾都加上符号“<eos>”。在测试时，一旦生成了符号“<eos>”，就中止生成过程。

序列生成



Teacher Forcing

曝光偏差（Exposure Bias）



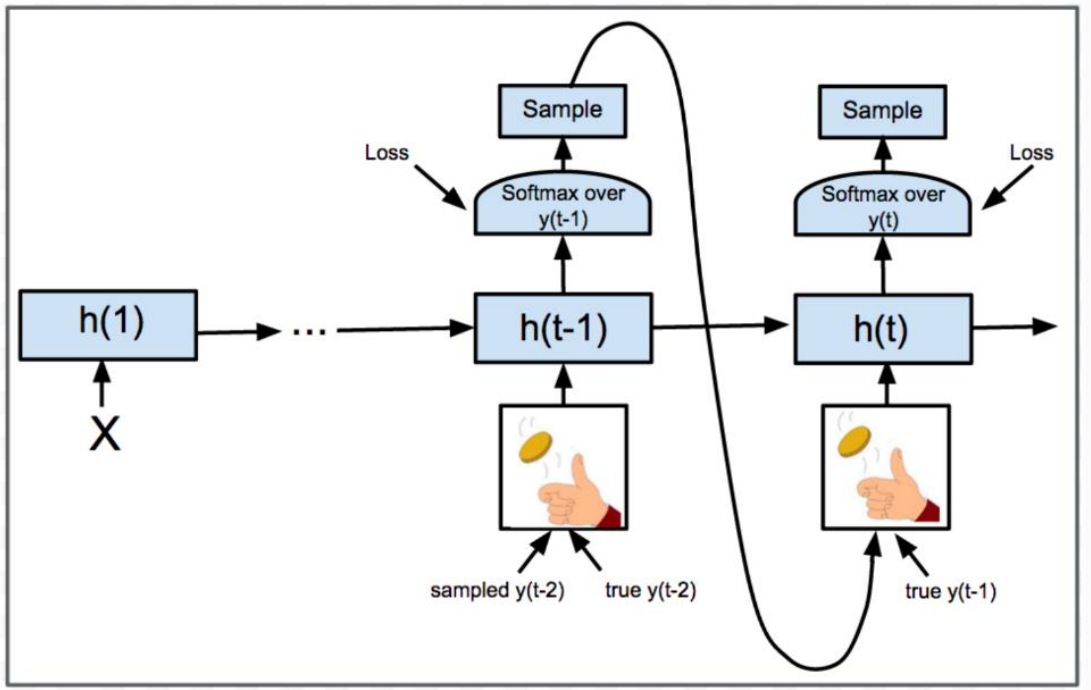
训练　　推断

如何改进?

- word-level supervision -> sentence-level supervision
  - GAN
  - 强化学习
- 减小真实数据分布和生成数据分布的差异
  - Schedule Sampling
  - DAD
  - Professor Forcing
- 非自回归模型

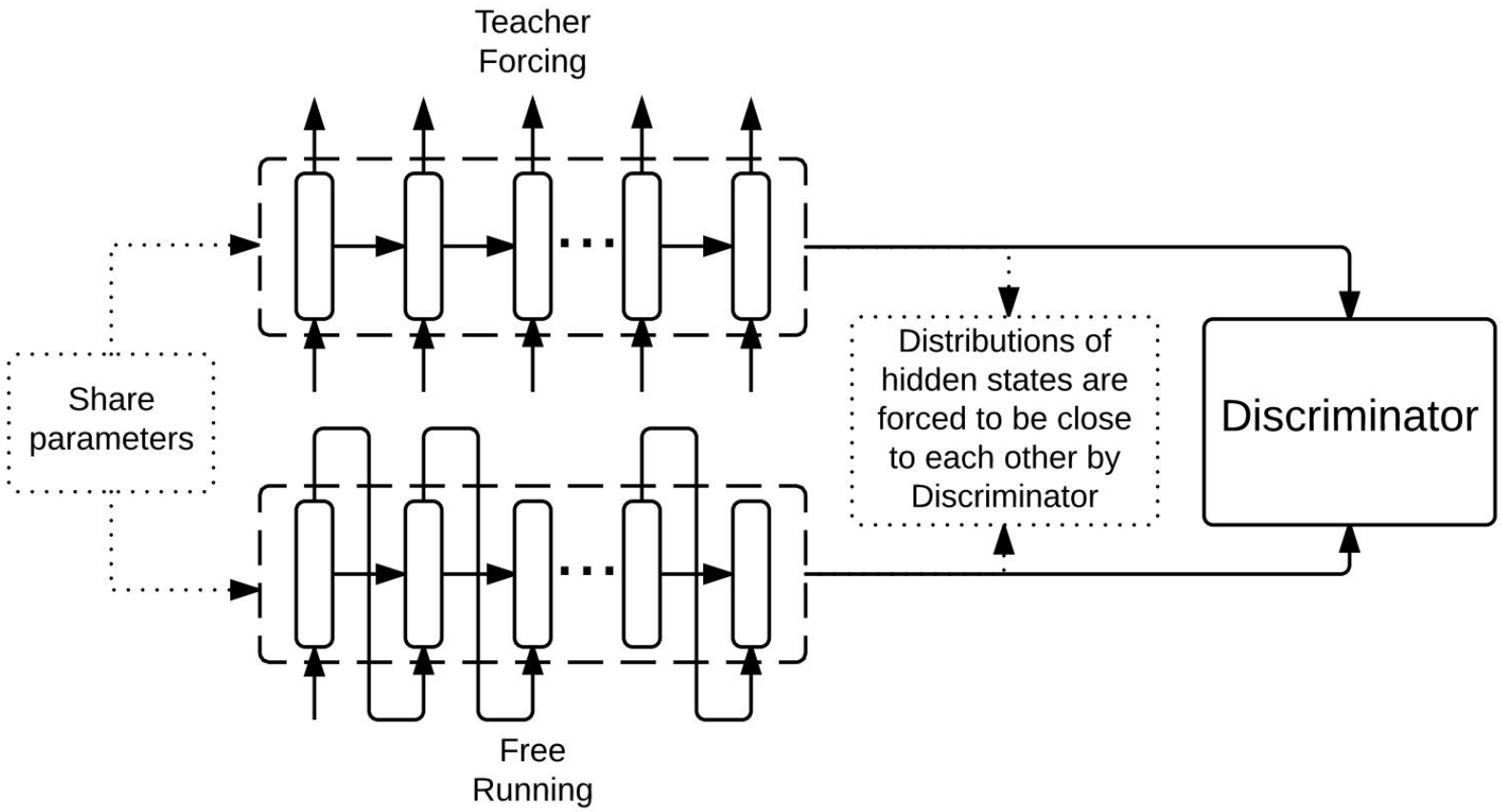
Schedule Sampling

- Gently Change the training from using the true previous token to the generated token

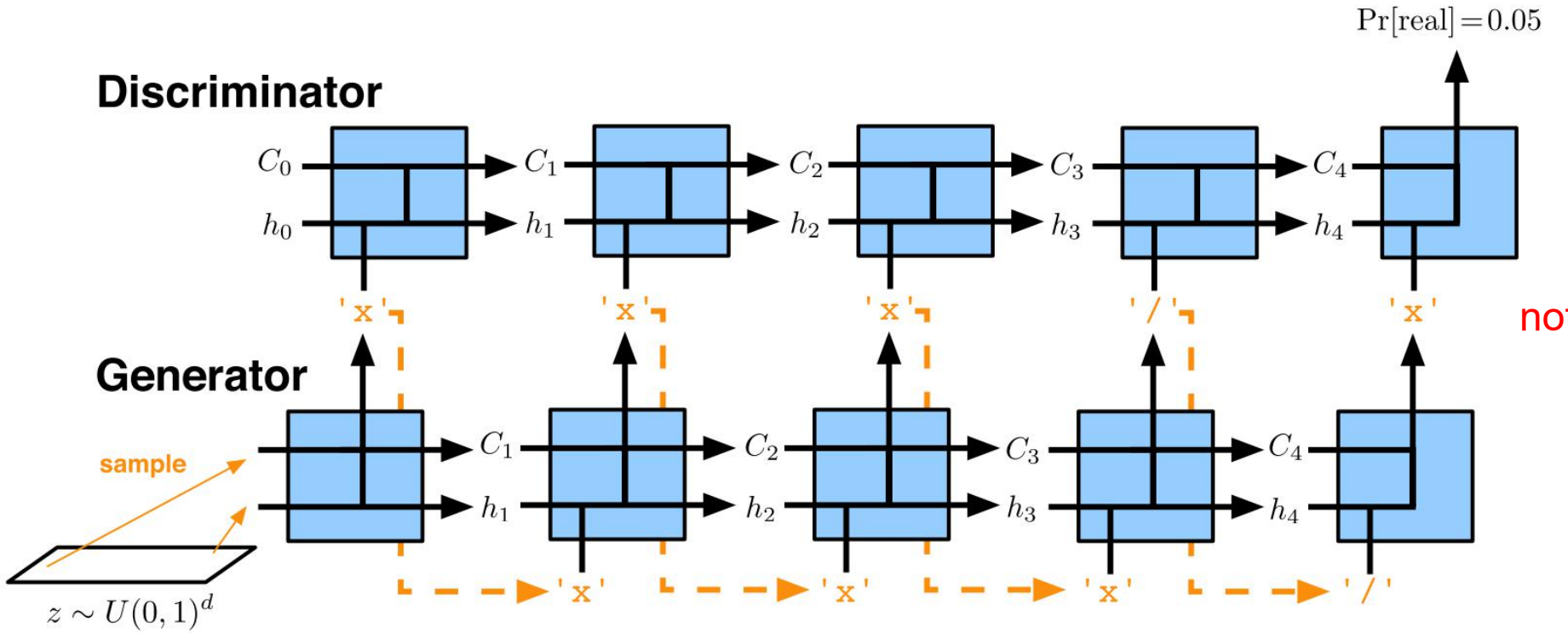


Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks, NIPS 2015

Professor Forcing



序列生成中的GAN

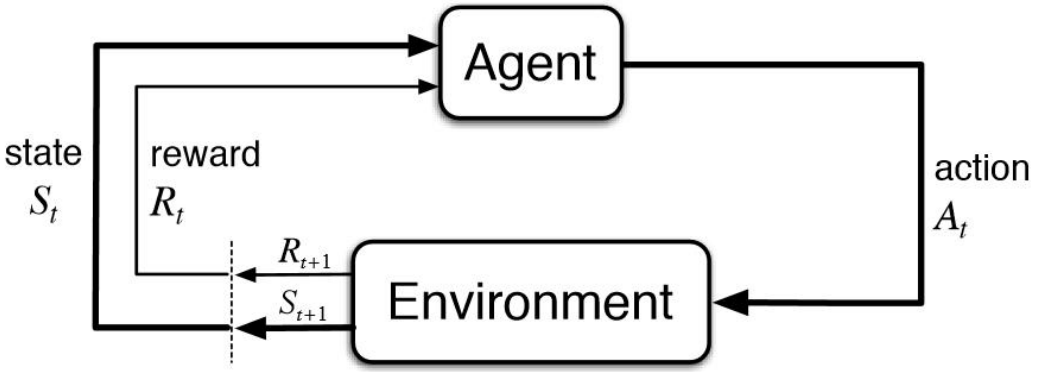


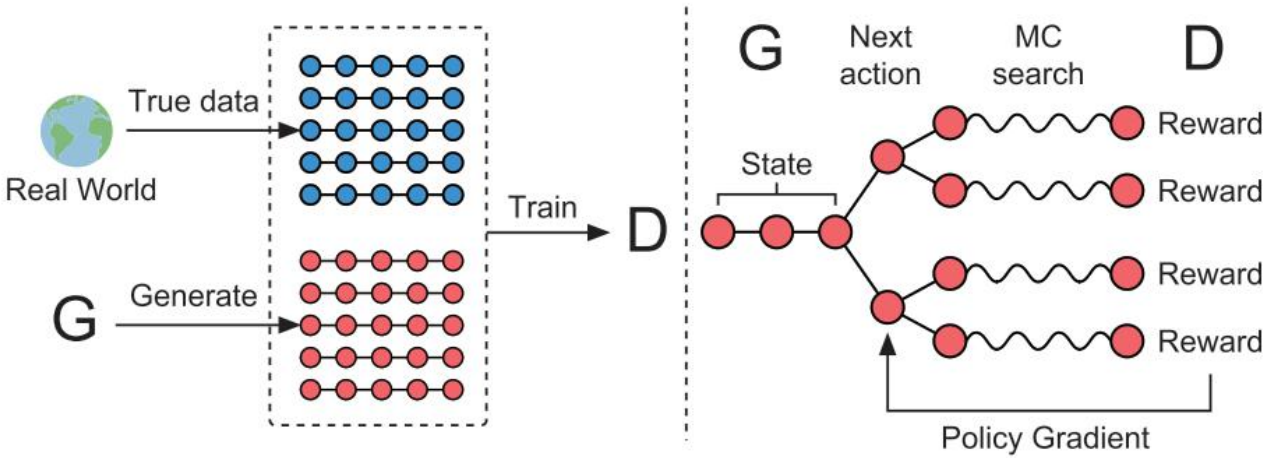
not differentiable

Gumbel-softmax distribution

SeqGAN

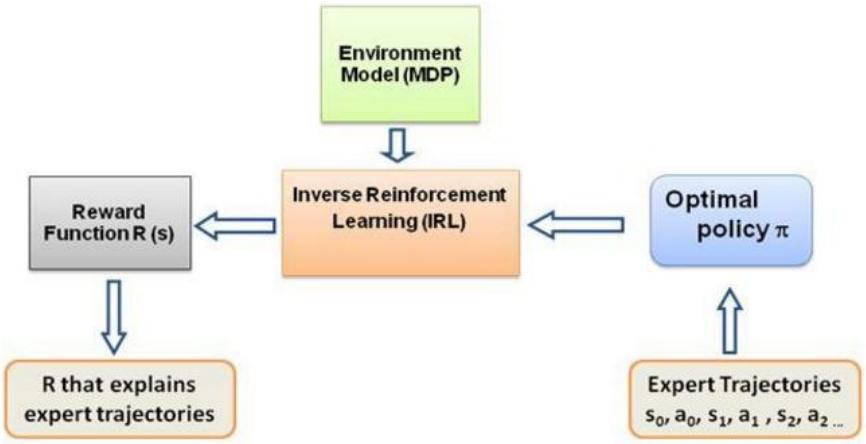
- 使用强化学习



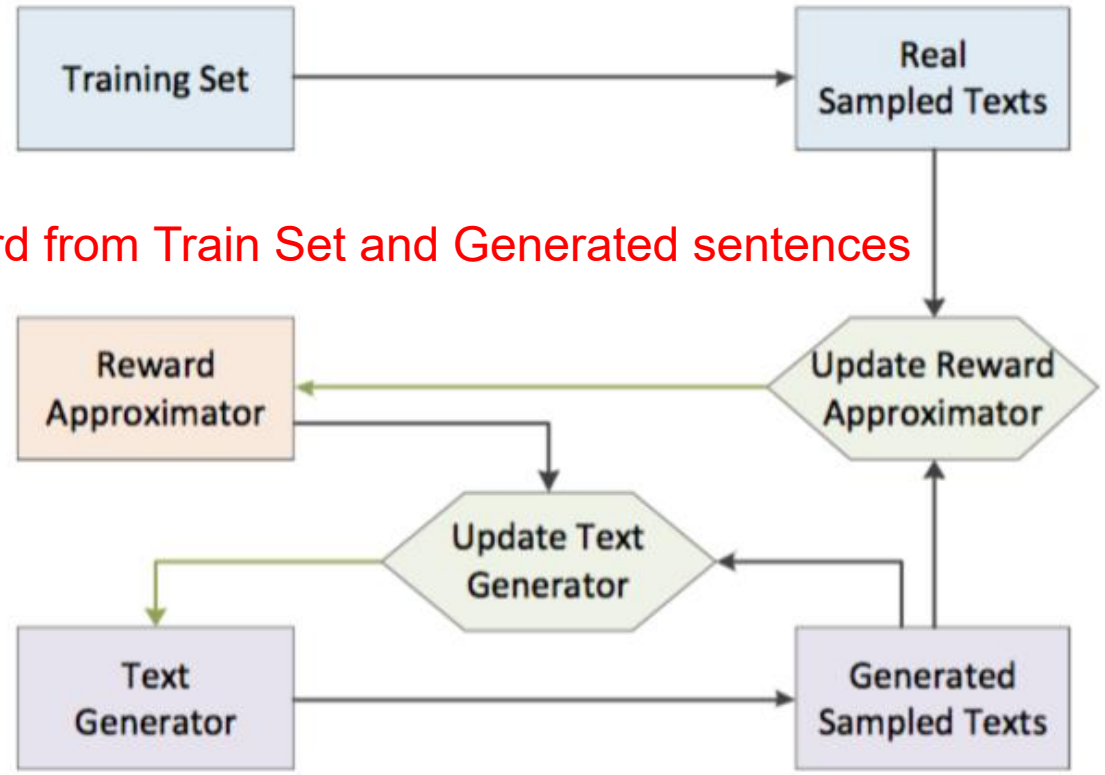


GAN VS IRL

- GAN在文本生成的不足
  - 奖励稀疏
  - 模型坍塌
- 逆向强化学习（Inverse Reinforcement Learning）



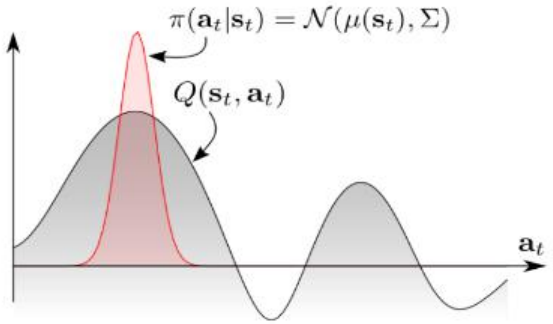
逆向强化学习

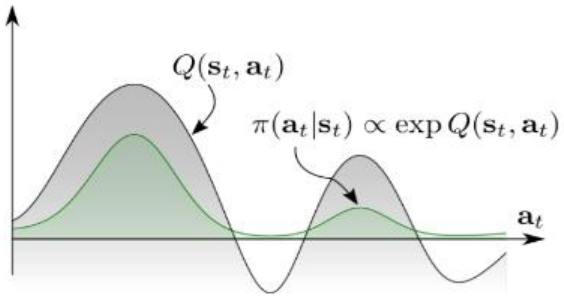


Toward Diverse Text Generation with Inverse Reinforcement Learning, IJCAI 2018

IRL减轻模型坍塌的原因

- A soft data distribution assumption
  - Max-entropy RL
  - It seeks for multimodal policy distribution.





- GAN: $KL(q_\theta(\tau)||P_{data})$
- IRL: $KL\left(q_\theta(\tau)||P_\phi(\tau)\right)$
- Since $P_\phi(\tau)$ never equals to zero due to its assumption, IRL can alleviate the model collapse problem in GANs

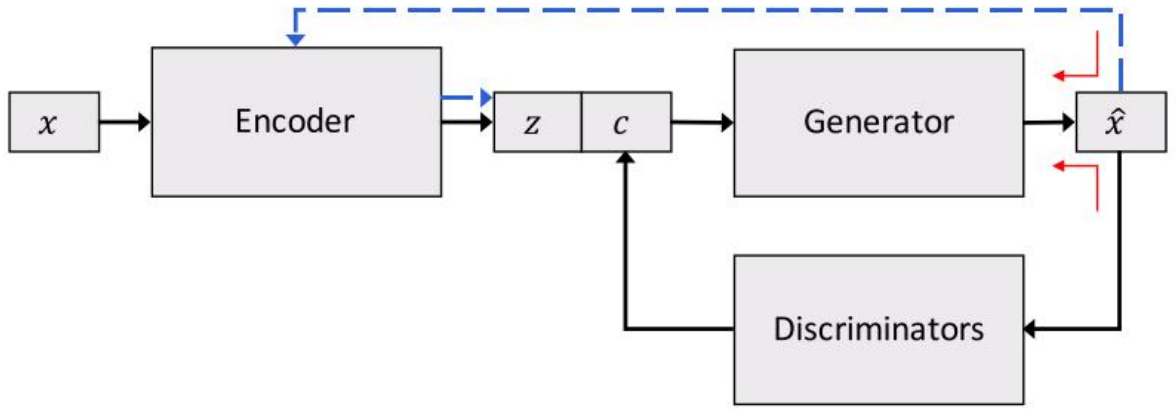
生成示例

Models	COCO	IMDB
MLE	(1) A girl sitting at a table in front of medical chair. (2) The person looks at a bus stop while talking on a phone.	(1) If somebody that goes into a films and all the film cuts throughout the movie. (2) Overall, it is what to expect to be she made the point where she came later.
SeqGAN	(1) A man holding a tennis racket on a tennis court. (2) A woman standing on a beach next to the ocean.	(1) The story is modeled after the old classic "B" science fiction movies we hate to love, but do. (2) This does not star Kurt Russell, but rather allows him what amounts to an extended cameo.
LeakGAN	(1) A bathroom with a toilet , window , and white sink. (2) A man in a cowboy hat is milking a black cow.	(1) I was surprised to hear that he put up his own money to make this movie for the first time. (2) It was nice to see a sci-fi movie with a story in which you didn't know what was going to happen next.
IRL (This work)	(1) A woman is standing underneath a kite on the sand. (2) A dog owner walks on the beach holding surfboards.	(1) Need for Speed is a great movie with a very enjoyable storyline and a very talented cast. (2) The effects are nothing spectacular, but are still above what you would expect, all things considered.

Text Style Transfer

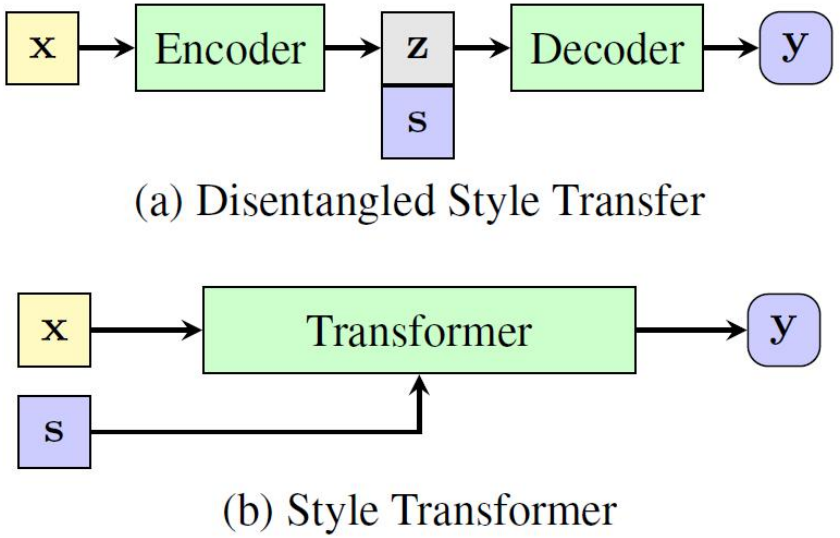
everything is fresh and so delicious

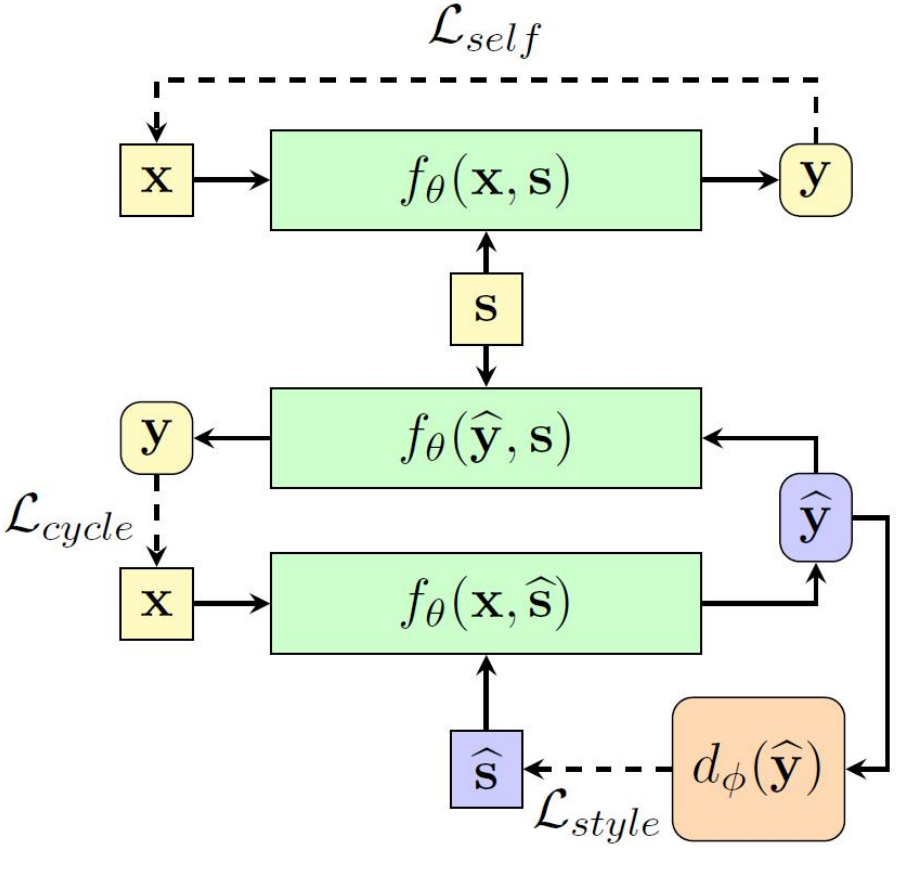
everything was so stale .



Toward Controlled Generation of Text, ICML 2017

Style Transformer





Style Transformer: Unpaired Text Style Transfer without Disentangled Latent Representation, ACL 2019

生成示例

	negative to positive
Input	the food 's ok , the service is among the worst i have encountered .
DAR	the food 's ok , the service is among great and service among .
CtrlGen	the food 's ok , the service is among the randy i have encountered .
Ours	the food 's delicious , the service is among the best i have encountered .
Human	the food is good , and the service is one of the best i 've ever encountered .
Input	this is the worst walmart neighborhood market out of any of them .
DAR	walmart market is one of my favorite places in any neighborhood out of them .
CtrlGen	fantastic is the randy go neighborhood market out of any of them .
Ours	this is the best walmart neighborhood market out of any of them .
Human	this is the best walmart out of all of them .
Input	always rude in their tone and always have shitty customer service !
DAR	i always enjoy going in always their kristen and always have shitty customer service !
CtrlGen	always good in their tone and always have shitty customer service !
Ours	always nice in their tone and always have provides customer service !
Human	such nice customer service , they listen to anyones concerns and assist them with it .

生成示例

	positive to negative
Input	everything is fresh and so delicious !
DAR	small impression was ok , but lacking i have piss stuffing night .
CtrlGen	everything is disgrace and so bland !
Ours	everything is overcooked and so cold !
Human	everything was so stale .
Input	these two women are professionals .
DAR	these two scam women are professionals .
CtrlGen	shame two women are unimpressive .
Ours	these two women are amateur .
Human	these two women are not professionals .
Input	fantastic place to see a show as every seat is a great seat !
DAR	there is no reason to see a show as every seat seat !
CtrlGen	unsafe place to embarrassing lazy run as every seat is lazy disappointment seat !
Ours	disgusting place to see a show as every seat is a terrible seat !
Human	terrible place to see a show as every seat is a horrible seat !

谢 谢