深度学习（2）——生成对抗网络

2018年08月21日 10:39:40 越溪 阅读数 4052

深度学习（2）——生成对抗网络

> 译文，如有错误请与笔者联系

摘要

本文提出一个通过对抗过程来预测生成模型的新框架，其中我们同时训练两个模型：一个用来捕捉数据分布的生成模型G和预测样本来自训练数据而不是G的概率的判别模型D。G的训练目的是最大化D的判别出错概率。该框架实现了“二元极小极大博弈（minimax two-player game）”。在G和D的任意函数空间中，存在唯一解，即G恢复了训练数据的概率分布并且D的判别概率始终是1/2。G和D由多层感知器定义的情况下，整个系统可以使用反向传播的方式（BP）进行训练。在整个训练或生成样例过程中不需要任何马尔科夫链或展开近似推理网络。实验通过对生成样本进行定性和定量的评估来证明该框架的潜力。

1 引言

深度学习的前景是发现丰富的层次模型，它们表示人工智能应用中遇到的各种数据的概率分布，例如自然图像，包含语音的音频波形和自然语言语料库中的符号。到目前为止，深度学习中最有影响力的深度学习应用涉及到判别模型，通常都是将高维度、丰富的感知输入（如图像、音频、视频等）映射到类别标签。这些成功应用主要基于反向传播和Dropout算法，使用具有梯度效果良好的分段线性单元（piecewise linear units）。深度生成模型的成功案例较少，因为一些最大似然估计的逼近和相关策略中均出现了很多难以处理的概率计算，而且很难借助分段线性单元的优势。本文提出了一种可以规避这些困难的新的生成模型。

在提出的对抗模型中，生成模型的对手是判别模型，即用来学习判断样本来自模型分布还是数据分布。生成模型可以看作是造假团队，试图制造假币并且没有检验假币是否合格就使用它，而判别模型类似警察，视图检测假币。在双方博弈中促进双方改进模式，直到判别模型难以分辨出真假。

该框架可以使用多种模型和优化算法进行训练。在本文中，我们讨论生成模型通过多层感知机传递随机噪声生成样本的特殊情况，判别模型也是多层感知机。我们将这种特殊的情况称为对抗网络。在这种情况下，我们仅使用非常成功的反向传播和Dropout算法来训练两模型，并且在生成模型中仅使用前向传播来生成样本。不需要使用近似推理和马尔科夫链。

2 相关工作

直到最近，生成模型的大多数工作仍集中在提供概率分布模型的参数化规范的模型上。通过最大化对数似然来训练模型。在该系列模型中，最成功的是deep Boltzmann machine。这些模型通常具有很难处理的似然函数，因此需要对似然梯度进行多次逼近。这些困难推动了“生成机器”的发展，这些模型没有明确地表示可能性，但能够从所需的分布中生成样本。生成随机网络是生成机器的一个例子，它可以用精确的反向传播进行训练，而不是Boltzmann机器所需要的大量近似值。这项工作通过消除生成随机网络中使用的马尔科夫链来拓展生成机器的概念。

$$\lim_{\sigma \to 0} \nabla_x E_{\epsilon \sim N(0,\sigma^2 I)} f(x+\epsilon) = \nabla_x f(x)$$

本文通过公式1来观察生成过程反向传播导数。当我们开始这项工作时，没有意识到Kingma 、Welling 和Rezende等人已经开发出更一般随机反向传播规则，通过具有有限方差的高斯分布进行反向传播，并反向传播到协方差参数及均值，我们在这项工作中将其视为超参数。Kingma等人使用随机反向传播来训练变分自动编码器（VAE）。和生成对抗网络相同的是，VAE将一个可微分生成器与第二个神经网络配对；不同的是，VAE第二个网络是使用近似推理的识别模型。GAN通过可见单元进行区分，因此不能对离散数据建模，而VAE需要通过隐藏单元进行区分，因此不能具有离散的潜在变量。其他类似VAE的方法与本文提到的方法没有太密切的关系。

以前的工作也使用了判别标准来训练生成模型。这些方法使用了深度生成模型难以处理的标准。这些方法甚至难以近似深度模型，因为他们涉及到概率的比率，这些概率不能使用降低概率的变分近似来近似。噪声对比估计（NCE）涉及通过学习权重来训练生成模型，使得该模型可用于区分固定噪声分布的数据。使用先前训练的模型作为噪声分布允许训练一系列质量提高的模型。这可以被视为一种非正式的竞争机制，其精神与对抗网络中的机制类似。NCE的关键限制在于其鉴别器由噪声分布的概率密度和模型分布的比率来定义，因此需要能够通过两种密度评估和反向传播。

以前的一些工作使用了两个神经网络竞争的一般概念。最相关的工作是可预测性最小化。在可预测性最小化中，第一个神经网络中的每个隐藏单元被训练为与第二网络的输出不同，第二网络在给定所有其他隐藏单元的值的情况下预测该隐藏单元的值。这项工作在三个重要方面与可预测性最小化不同：1）在这项工作中，网络之间的竞争是唯一的训练标准，并且自身足以训练网络。可预测性最小化只是一个正则化器，它鼓励神经网络的隐藏单元在完成其他任务时在统计上独立;它不是主要的训练标准。 2）比赛的性质不同。在可预测性最小化中，比较两个网络的输出，一个网络试图使输出相似，另一个网络试图使输出不同。有问题的输出是单个标量。在GAN中，一个网络产生一个丰富的高维向量，用作另一个网络的输入，并尝试选择另一个网络不知道如何处理的输入。 3）学习过程的规范是不同的。可预测性最小化被描述为优化问题，其中目标函数被最小化，并且学习接近目标函数的最小值。 GAN基于二元极小极大博弈而不是优化问题，并且具有一个网络寻求最大化而另一个寻求最小化的价值函数。游戏终止于一个鞍点，该鞍点相对于一个玩家的策略是最小的，而对于另一个玩家的策略是最大值。

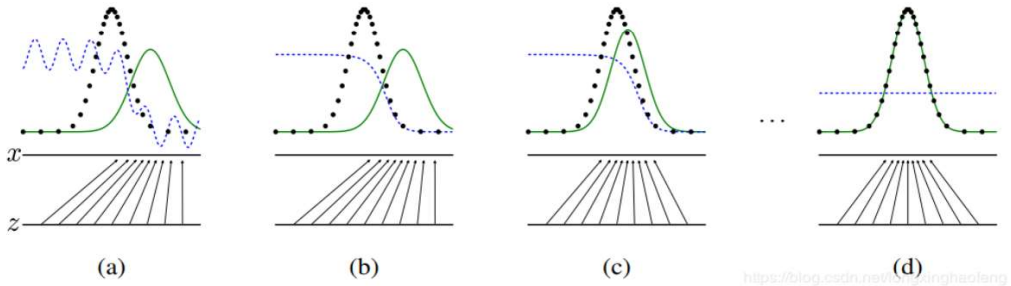
生成对抗网络有时与“对抗样本”的相关概念混淆。对抗样本是通过在分类网络的输入上直接使用基于梯度的优化而找到的样本，以便找到类似于数据但错误分类的样本。这与目前的工作不同，因为对抗样本不是训练生成模型的机制。相反，对抗样本主要是一种分析工具，用于表明神经网络以有趣的方式表现，通常以高置信度对两个图像进行不同的分类，即便它们之间的差异对于人类观察者来说是不可察觉的。这种对抗样本的存在确实表明，生成性对抗网络训练可能效率低下，因为它们表明现代判别网络可以在不模仿该类的任何人类可感知属性的情况下容易地识别一个类。

3 对抗网络

当生成模型和判别模型都是多层感知器时，可直接应用生成模型框架。 为了在输入数据x上学习生成器的分布p_g，我们在输入噪声变量$p_z(z)$上定义先验，然后将数据空间的映射表示为$G(z;\theta_g)$，其中G是由多层感知器表示的可微函数，参数是θ_g。我们还定义了输出单个标量的第二个多层感知器$D(x;\theta_d)$。 $D(x)$表示x来自数据而不是p_g的概率。我们训练D来最大化为训练样本和来自G的样本分配正确标签的概率(即最大化$\log D(x)$)。我们同时训练G来最小化 $\log(1-D(G(z)))$。 换句话说，D和G使用值函数$V(G，D)$进行以下二元极小极大博弈：

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p_z(z)}[\log(1-D(G(z)))] \tag{1}$$

在下一节中，我们将对对抗网络进行理论分析，本质上表明，当G和D被赋予足够的容量时，即在非参数极限下，训练标准允许我们恢复数据生成分布。
有关该方法的不太正式，但更具教学意义的解释，请参见图1。



> 训练生成对抗网络的同时，更新判别分布（D，蓝色虚线）使D能区分数据生成分布（黑色虚线）的样本和生成分布p_g（G，绿色实线）的样本。
> 下面的水平线为均匀采样z的区间，上面的水平线为x的分布区间。向上箭头表示映射$x=G(z)$如何在变换后的样本上施加非均匀分布p_g。G在p_g高密度区域收缩，且在p_g的低密度区域扩散。
> （a）考虑靠近收敛的对抗对：p_g类似于p_{data}，D是部分准确的分类器。
> （b）保持G不动，优化D，在算法D的内循环中训练以区分样本和数据，收敛到$D*(x)=\frac{p_{data}(x)}{p_{data}(x)+p_g(x)}$，使得分类准确率最高。
> （c）在每次更新G之后，D的梯度引导G（z）流向更可能被分类为数据的区域。即保持D不动，优化G，直到混淆程度最高。
> （d）经过多次迭代后，如果G和D的性能足够，它们将达到两个都无法改善的点，因为$p_g=p_{data}$。判别器不能区分两个分布，即$D(x)=\frac{1}{2}$。

在实践中，我们必须使用迭代的数值方法来实现游戏。在训练的内循环中单独优化D到最优在计算上是不允许的，并且如果数据集有限，这将导致过拟合。相反，我们在优化D的k个步骤和优化G的一个步骤之间交替。只要G变化足够慢，这导致D保持接近其最优解。该过程在算法1中正式呈现。

> 算法1

算法1 用Mini-batch随机梯度下降训练生成对抗网络。应用于判别器的梯度下降次数k是超参数。 我们在实验中使用了k=1，这是代价最低的选择。

---

for 训练迭代次数 **do**
  for 优化 k 次 **do**
    从$p_g(z)$选取的m个batch$\{z^{(1)},\cdots,z^{(m)}\}$
    从$p_{data}(x)$选取的m个batch$\{x^{(1)},\cdots,x^{(m)}\}$
    通过加上随机梯度来更新判别器
  end for
  从$p_g(z)$选取的m个batch$\{z^{(1)},\cdots,z^{(m)}\}$
  通过减去其随机梯度来更新生成器
end for
基于梯度的更新可以使用任何标准的基于梯度的学习规则。本文在实验中使用了动量(momentum)梯度下降。

在实践中，公式1可能无法为G学习提供足够的梯度。在学习初期，当G很差时，D可以高置信度拒绝G生成的样本，因为它们与训练数据明显不同。在这种情况下，$\log(1-D(G(z)))$饱和。我们可以训练G以最大化$logD(G(z))$，而不是训练G以最小化$log(1-D(G(z)))$。该目标函数导致G和D动力学的相同固定点，但在学习早期提供更强的梯度。

4 理论成果

生成器G隐含地将概率分布p_g定义为当$z \sim p_z$时获得的样本$G(z)$的分布。因此，如果有足够的性能和训练时间,我们希望算法1收敛到一个好的估计器p_{data}。 该部分的结果是在非参数设置中完成的，例如，我们通过研究收敛性来表示具有无限容量的模型概率密度函数的空间。

我们将在4.1节中说明这个最大最小游戏（使对方得点减到最低以使自己得最高分的战略）对于$p_g = p_{data}$具有全局最优。然后我们将在4.2节中展示算法1优化等式1，从而获得所需的结果。

4.1 $p_g = p_{data}$具有全局最优性

任何给定的生成器G,我们首先考虑它的最优判别器D.

命题1.

固定G，最优判别器D是
证明. 给定任意生成器G，判别器D的训练标准为最大化目标函数$V(G,D)$:

$$\begin{aligned} V(G,D) &= \int_x p_{data}(x)\log(D(x))dx + \int_z p_z(z)\log(1-D(g(z)))dz \\ &= \int_x p_{data}(x)\log(D(x)) + p_g(x)\log(1-D(x))dx \end{aligned} \tag{3}$$

对于与任意$(a,b) \in R^2 \backslash \{0,0\}$，当$y \in [0,1]$时，函数$y \to a\log(y)+b\log(1-y)$在$\frac{a}{a+b}$处达到最大值。无需在$Supp(p_{data}) \bigcap Supp(p_g)$外定义判别器。证毕。

D 的训练目标可视为最大化估计条件概率$P(Y=y|x)$的对数似然。当$y=1$时，x来自$p-data$；当$y=0$时，x来自p_g。公式1的二元极小极大博弈可形式化为：

$$\begin{aligned} C(G) &= max_D V(G,D) \\ &= E_{x\sim p_{data}}[\log D_G^*(x)] + E_{z\sim p_z(z)}[\log(1-D_G^*(G(x)))] \\ &= E_{x\sim p_{data}}[\log D_G^*(x)] + E_{x\sim p_g}[\log(1-D_G^*(x))] \\ &= E_{x\sim p_{data}}[\log \frac{p_{data}(x)}{p_{data}(x)+p_g(x)}] + E_{x\sim p_g}[\log \frac{p_g(x)}{p_{data}(x)+p_g(x)}] \end{aligned} \tag{4}$$

> 定理1.当且仅当$p_g = p_{data}$时，训练标准$C(G) = \max V(G,D)$可以达到全局最优解。此时，$C(G) = -log4$。

证明.$p_g = p_{data}$时，$D_G^*(x) = \frac{1}{2}$。由公式（4）得$C(G) = \log\frac{1}{2} + \log\frac{1}{2} = -\log 4$。
可以看到，仅当$p_g = p_{data}$时，这是$C(G)$的最优值：

$$E_{x\sim p_{data}}[-\log 2] + E_{x\sim p_g}[-\log 2] = -\log 4$$

从$C(G) = V(G, D_G^*)$中减去上式，得：

$$C(G) = -\log 4 + KL(p_{data} \,||\, \frac{p_{data}+p_g}{2}) + KL(p_g \,||\, \frac{p_{data}+p_g}{2}) \tag{5}$$

其中，KL为 Kullback-Leibler 散度 。我们在前面的表达式中认识到模型分布和数据生成过程之间的Jensen-Shannon 散度：

$$C(G) = -\log 4 + 2\cdot JSD(P_{data} \,||\, p_g) \tag{6}$$

由于两个分布间的 Jensen-Shannon散度总为非负的，当且仅当两者相等时JSD为0，此时$C(G)$的全局最小值为$C* = -log4$，且唯一解为$pg_{=}p_{data}$，即生成模型完美复制数据的分布。

4.2 算法1的收敛性

命题2.

如果G和D性能足够，并且在算法1的每一步，对于给定的G，允许判别器达到最优，并且更新p_g以便提高标准

$$E_{x\sim p_{data}}[\log D_G^*(x)] + E_{x\sim p_g}[\log(1-D_G^*(x))]$$

那么p_g就会收敛到p_{data}。

证明. 如在上述标准中所做的那样，考虑$V(G,D) = U(p_g, D)$作为p_g的函数。 注意，$U(p_g, D)$在pg中是凸的(即凸函数)。 凸函数的上确界的次导数可以找到函数在最大值处的导数。换句话说，如果$f(x) = sup_{\alpha\in A} f_\alpha(x)$

,且对于每一个α，$f_\alpha(x)$关于x时凸的，那么，如果$\beta = argsup_{\alpha \in A} f_\alpha(x)$,则$\partial f_\beta(x) \in \partial f$。等价于黑顶对应的G和最优判别D，梯度下降更新$p_g$。在定理1 中证明$sup_D U(p_g, D)$关于$p_g$是凸的且具有唯一的全局最优解，因此，$p_g$更新足够小时，$p_g$收敛到$p_x$，证毕。

实际上，对抗网络通过函数$G(z;\theta_g)$表示有限的p_g分布的簇，且优化θ_g而不是p_g本身，所以证明不适用。然而，实际中多层感知机的优异性能表明尽管缺少理论保证，它们仍为可合理使用的模型。

5 实验

我们训练了对抗网络的一系列数据集，包括MNIST， the Toronto Face Database（TFD）和CIFAR-10。 生成网络（G）使用ReLU和Sigmoid激活函数的混合，而判别网络（D）使用maxout 激活函数。训练判别网络时使用Dropout。虽然我们的理论框架允许在G的中间层使用Dropout和其他噪声，但我们仅在G的最底层使用噪声输入。

本文用G生成的样本来拟合高斯Parzen窗，并计算该分布下的对数似然来估计测试集数据在p_g下的概率。通过验证集上的交叉验证获得高斯的σ参数。该程序在Breuleux等人的文章中有介绍,并用于确切似然性不可控的各种生成模型上。实验结果报告在表1中。这种估计可能性的方法具有稍高的方差，并且在高维空间中表现不佳，但这是我们知道的最好方法。生成模型的进展是可以取样，但不能估计可能性，这直接促使进一步研究如何评价这些模型。在图2和图3中，我们显示了训练后从G中抽取的样本。虽然我们没有声称这些样本比现有方法生成的样本更好，但我们认为这些样本至少可以与文献中更好的生成模型竞争，并突出了对抗框架的潜力。

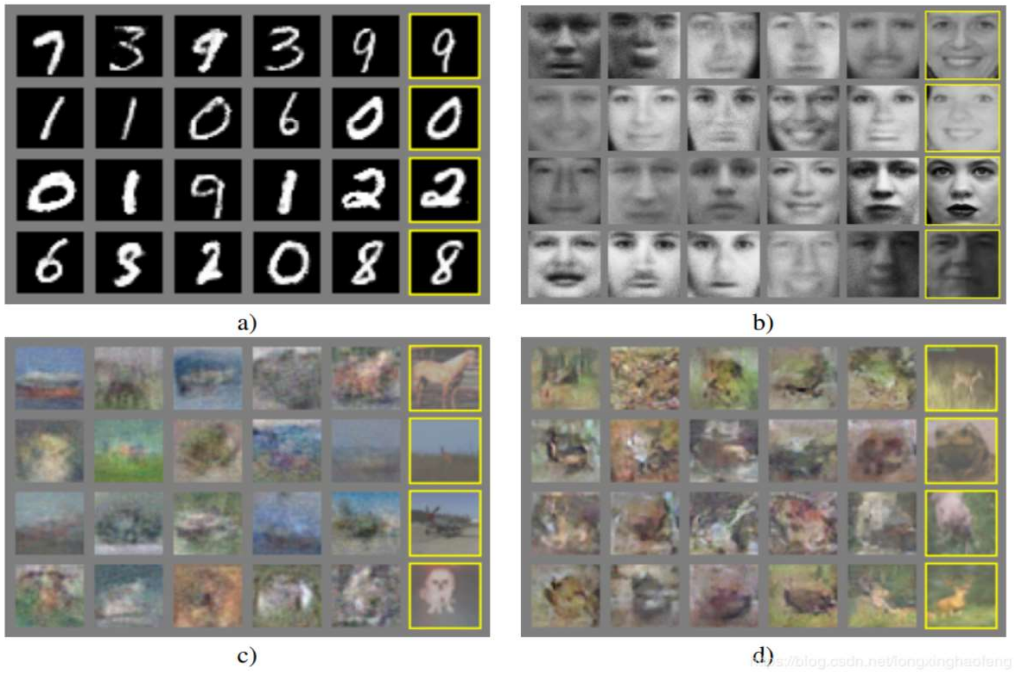


图2：来自模型部分样本的可视化。每张图最右边的列显示最近的相邻样本的训练示例，以证明模型没有记住训练集。 样品是随机抽取，而不是挑选。与其他深度生成模型的可视化不同，这些图像是来自模式分布的实际样本，而不是给定隐藏单元样本的条件均值。而且，这些样品是不相关的，因为取样过程不依赖于马尔可夫链混合。
a）MNIST
b）TFD
c）CIFAR-10（完全连接模型）
d）CIFAR-10（卷积鉴别器和“反卷积”发生器）

Figure 3: Digits obtained by linearly interpolating between coordinates in z space of the full model.

图3：通过在完整模型的z空间中的坐标之间线性插值获得的数字。

Model	MNIST	TFD

DBN	138±2	1909±66
Stacked CAE	121±1.6	2110±50
Deep GSN	214±1.1	1890±29
Adversarial nets	225±2	2057±26

表1：Parzen基于窗口的对数似然估计。
在MNIST上报告的数字是测试集上样本的平均对数似然，并且在示例中计算平均值的标准误差。在TFD上，我们计算了数据集折叠的标准误差，使用每个折叠的验证集选择不同的σ。在TFD上，σ在每个折叠上交叉验证，并计算每个折叠的平均对数似然。对于MNIST，我们将与其他数据集的实数（而不是二进制）版本进行比较。

#6 优点和缺点
与以前的建模框架相比，这个新框架具有优点和缺点。缺点主要在于没有$p_g(x)$的明确表示（就像一个黑盒子，不可解释），并且D在训练期间必须与G很好地同步（特别是，在不更新D的情况下，G不得过多训练，以避免“Helvetica场景”.否则，x的相同值时，G丢失太多的z值以至于模拟p_{data}多样性不足”），就像Boltzmann机器的负链必须在学习步骤之间保持最新一样。 优点是永远不需要马尔可夫链，仅用反向传播获得梯度，学习期间不需要推理，并且可以将多种函数融合到模型中。表2总结了生成式对抗网与其他生成建模方法的比较。

	Deep directed graphical models	Deep undirected graphical models	Generative autoencoders	Adversarial models
Training	Inference needed during training.	Inference needed during training. MCMC needed to approximate partition function gradient.	Enforced tradeoff between mixing and power of reconstruction generation	Synchronizing the discriminator with the generator. Helvetica.
Inference	Learned approximate inference	Variational inference	MCMC-based inference	Learned approximate inference
Sampling	No difficulties	Requires Markov chain	Requires Markov chain	No difficulties
Evaluating $p(x)$	Intractable, may be approximated with AIS	Intractable, may be approximated with AIS	Not explicitly represented, may be approximated with Parzen density estimation	Not explicitly represented, may be approximated with Parzen density estimation
Model design	Nearly all models incur extreme difficulty	Careful design needed to ensure multiple properties	Any differentiable function is theoretically permitted	Any differentiable function is theoretically permitted

表2：挑战生成模型：
对于涉及模型的每个主要操作，对于深度生成建模的不同方法都有不同的方法。

上述优点主要是针对计算的。对抗模型也可能仅通过流经判别器的梯度从G中获得一些统计优势，而不是直接用数据示例更新。 这意味着输入的部分不会直接复制到生成器的参数中。对抗性网络的另一个优点是它们可以代表非常尖锐，甚至简并的分布，而基于马尔可夫链的方法要求分布有些模糊，以便链能够在模式之间混合。

#7 结论和拓展
该框架承认了许多简单的扩展：

1. 添加c至G和D的输入，可获得条件的生成模型$p(x|c)$。
2. 给定x，为预测z，训练任意的网络可学习近似推理。类似于 wake-sleep 算法训练出的推理网络，但训练推理网络时可能要用到训练完成后的固定的生成网络。
3. 来近似建模所有的条件概率$P(x_S|x_S)$，其中，S为通过训练共享参数的条件模型簇的x的索引。本质上，对抗的网络可用于随机扩展 MP-DBM。
4. 半监督学习：当标签数据有限时，判别网络或推理网络的特征不会提高分类器效果。
5. 效率改善：为协调G和D设计更好的方法，或训练期间确定更好的分布来采样z，从而加速训练。

本文展示对抗的模型框架的可行性，证明了此研究方向有用。

补充

OpenAI Ian Goodfellow的Quora问答

与其他生成式模型相比较，生成式对抗网络有以下四个优势：

1、根据实际的结果，它们看上去可以比其它模型产生了更好的样本（图像更锐利、清晰）。
2、生成对抗式网络框架能训练任何一种生成器网络（理论上-实践中，用 REINFORCE 来训练带有离散输出的生成网络非常困难）。大部分其他的框架需要该生成器网络有一些特定的函数形式，比如输出层是高斯的。重要的是所有其他的框架需要生成器网络遍布非零质量（non-zero mass）。生成对抗式网络能学习可以仅在与数据接近的细流形（thin manifold）上生成点。
3、不需要设计遵循任何种类的因式分解的模型，任何生成器网络和任何鉴别器都会有用。
4、无需利用马尔科夫链反复采样，无需在学习过程中进行推断（Inference），回避了近似计算棘手的概率的难题。

与PixelRNN相比，生成一个样本的运行时间更小。GAN 每次能产生一个样本，而 PixelRNN 需要一次产生一个像素来生成样本。

与VAE 相比，它没有变化的下限。如果鉴别器网络能完美适合，那么这个生成器网络会完美地恢复训练分布。换句话说，各种对抗式生成网络会渐进一致（asymptotically consistent），而 VAE 有一定偏置。

与深度玻尔兹曼机相比，既没有一个变化的下限，也没有棘手的分区函数。它的样本可以一次性生成，而不是通过反复应用马尔可夫链运算器（Markov chain operator）。

与 GSN 相比，它的样本可以一次生成，而不是通过反复应用马尔可夫链运算器。

与NICE 和 Real NVE 相比，在 latent code 的大小上没有限制。

GAN目前存在的主要问题：

1、解决不收敛（non-convergence）的问题。

目前面临的基本问题是：所有的理论都认为 GAN 应该在纳什均衡（Nash equilibrium）上有卓越的表现，但梯度下降只有在凸函数的情况下才能保证实现纳什均衡。当博弈双方都由神经网络表示时，在没有实际达到均衡的情况下，让它们永远保持对自己策略的调整是可能的

【OpenAI Ian Goodfellow的Quora】。

2、难以训练：崩溃问题（collapse problem）

GAN模型被定义为极小极大问题，没有损失函数，在训练过程中很难区分是否正在取得进展。GAN的学习过程可能发生崩溃问题（collapse problem），生成器开始退化，总是生成同样的样本点，无法继续学习。当生成模型崩溃时，判别模型也会对相似的样本点指向相似的方向，训练无法继续。

【Improved Techniques for Training GANs】

3、无需预先建模，模型过于自由不可控。

与其他生成式模型相比，GAN这种竞争的方式不再要求一个假设的数据分布，即不需要formulate p(x)，而是使用一种分布直接进行采样sampling，从而真正达到理论上可以完全逼近真实数据，这也是GAN最大的优势。然而，这种不需要预先建模的方法缺点是太过自由了，对于较大的图片，较多的 pixel的情形，基于简单 GAN 的方式就不太可控了。在GAN[Goodfellow Ian, Pouget-Abadie J] 中，每次学习参数的更新过程，被设为D更新k回，G才更新1回，也是出于类似的考虑。

#参考：
Generative Adversarial Nets
生成式对抗网络GAN研究进展（二）——原始GAN
与判别网络对抗的生成网络 (Generative Adversarial Nets)
GAN公式原理推到