本期我们来聊聊GANs（Generative adversarial networks，对抗式生成网络，也有人译为生成式对抗网络）。GAN最早由Ian Goodfellow于2014年提出，以其优越的性能，在不到两年时间里，迅速成为一大研究热点。

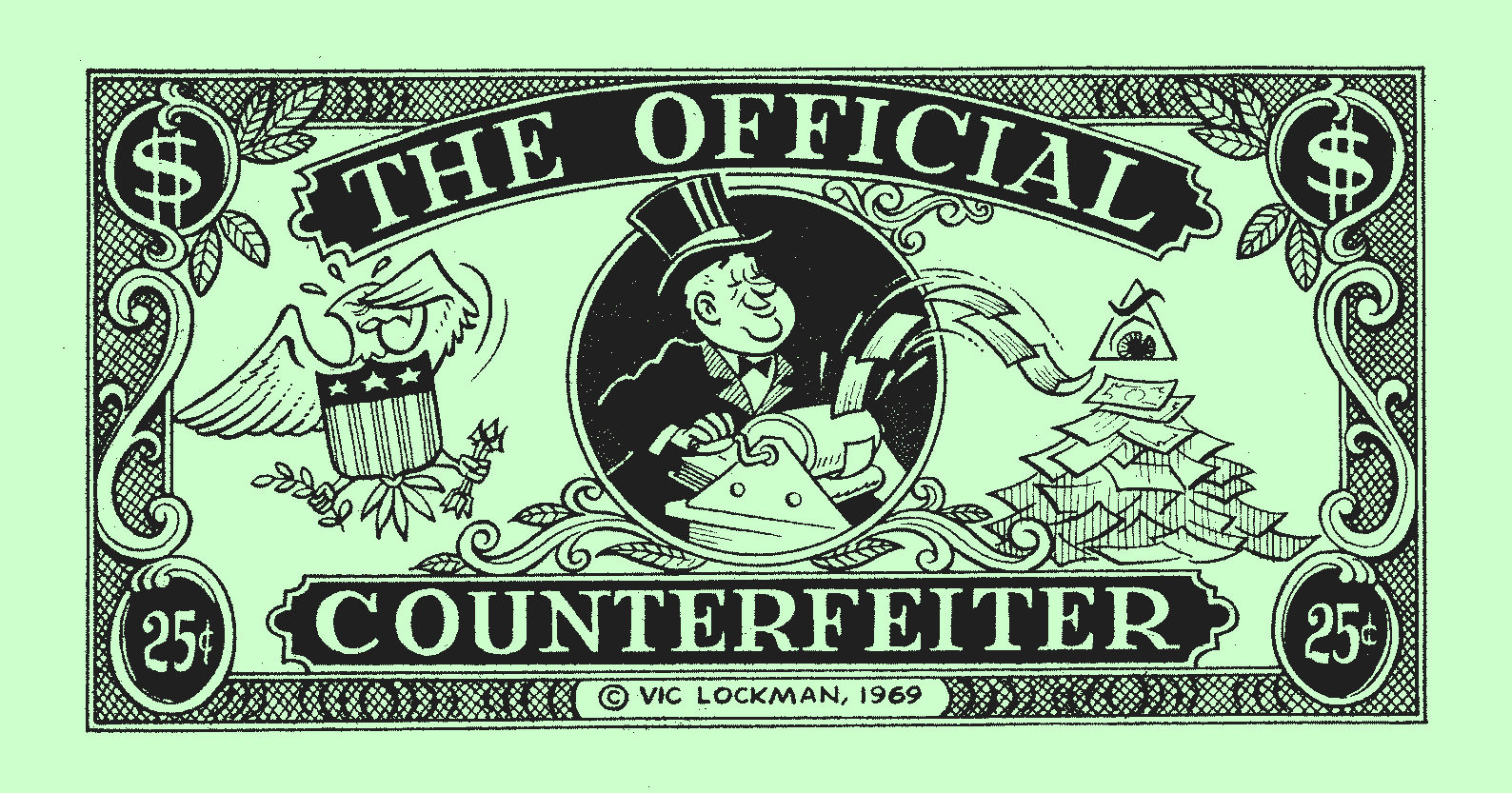
**GANs与博弈论**

GANs是一类生成模型，从字面意思不难猜到它会涉及两个“对手”，一个称为Generator（生成者），一个称为Discriminator（判别者）。Goodfellow最近arxiv上挂出的GAN tutorial文章中将它们分别比喻为伪造者（Generator）和警察（Discriminator）。伪造者总想着制造出能够以假乱真的钞票，而警察则试图用更先进的技术甄别真假钞票。两者在博弈过程中不断升级自己的技术。



从博弈论的角度来看，如果是零和博弈（zero-sum game），两者最终会达到纳什均衡（Nash equilibrium），即存在一组策略(g, d)，如果Generator不选择策略g，那么对于Discriminator来说，总存在一种策略使得Generator输得更惨；同样地，将Generator换成Discriminator也成立。囚徒的困境就是一个典型的零和博弈。

如果GANs定义的loss function满足零和博弈，并且有足够多的样本，双方都有充足的学习能力情况，在这种情况下，Generator和Discriminator的最优策略即为纳什均衡点，也即：Generator产生的都是“真钞”（材料、工艺技术与真钞一样，只是没有得到授权...），Discriminator会把任何一张钞票以1/2的概率判定为真钞。



讲到这里，今天的推送基本就讲完了...

额，至少思想和框架都介绍完了，具体怎么操作呢，大家可以先想一想，然后再往下看。

下面的内容，我们分为三个部分：

1. 为什么要研究生成模型（Generative model）？只是出于好玩吗？
2. GANs——Ian Goodfellow在2014年提出来的版本，以及它的一些演化（GANs有很多种，这里主要介绍最初的版本，其他版本将在以后介绍）
3. GANs有什么优缺点？

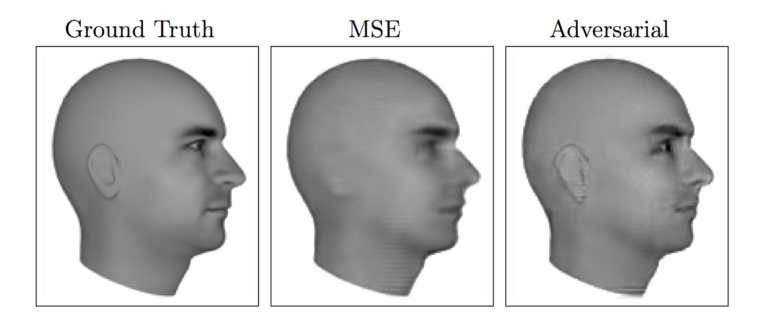
以下内容根据Goodfellow 2014年发表的GAN论文和代码（arxiv 1406.2661）以及近期arxiv上挂出的NIPS tutorial（arxiv 1701.00160 ）整理。

**为什么要研究生成模型**

先来讲讲为什么要研究生成模型，首先，生成模型真的很好玩......

还是来看看Ian Goodfellow怎么说吧。

* 高维概率分布在实际应用中非常常见。训练生成模型，并用生成模型采样可以检验我们对高维概率分布的表示和操纵能力。
* 生成模型能够嵌入到增强学习（reinforcement learning）的框架中。例如用增强学习求解规划问题，可以用生成模型学习一个条件概率分布，agent可以根据生成模型对不同actions的响应，选择尽可能好的action。
* 生成模型一般是非监督或者半监督模型，能够处理数据缺失问题，并且还可以用于预测缺失的数据。
* 生成模型，特别是GANs，能处理多模态输出（multi-modal）的问题。多模态输出问题（对于离散输出来说，就是多类标问题【multi-label】）是很多机器学习算法没办法直接处理的问题，很多机器学习算法的loss定义为均方误差，它实际上将多种可能的输出做了平均。下图给出了一个预测视频的下一帧的例子，使用MSE训练的结果，图像模糊了（它对所有可能的结果做了平均），而使用GANs训练的结果，则不存在这种问题。



* 某些任务在本质上要求根据某些分布产生样本。例如：用低分辨率图像产生高分辨率图像；用草图生成真实图像；根据卫星图像产生地图等。如下图所示。



**最初版本的GANs模型**

GANs在很多应用问题上取得非常好的效果，让生成模型进入人们的视野。前面已经介绍了GANs的思想，接下来，我们就来进一步介绍GANs是怎么做的。

首先，我们需要定义博弈的双方：Generator和Discriminator。

以训练生成图像为例，Generator的任务就是根据输入产生图像。而Discriminator则是判断图像是真实图像还是Generator生成的图像。

Generator和Discriminator具体怎么做可以自己定义。

这里根据我的理解，举个例子：我们知道卷积神经网络能够实现图像抽象特征的提取，如果我们定义Generator的输入为“图像的特征”（训练时没有“图像的特征”，输入可以是随机的噪声向量），那么可以用含有反卷积层或者上采样层的网络实现图像的重构，也就是图像的生成。

Discriminator是一个二元分类器，输入是图像，输出是两类：“自然”图像/Generator产生的图像。这里说的“自然”图像并不一定是自然图像，可以是合成的图像，人眼看上去图像是自然的。二元分类器有很多种，卷积神经网络是一个不错的选择。

其次，要定义loss function才能训练。前面说了，GANs可以看成一个博弈，那么博弈双方都会有cost（代价），如果是零和博弈，那么双方的cost之和为0。Discriminator是一个分类器，它的loss可以定义用交叉熵来定义：



如果是零和博弈，那么Generator的loss就定义为：



整个优化问题就是一个minmax博弈：



Goodfellow在2014年的论文中证明了在理想情况下（博弈双方的学习能够足够强、拥有足够的数据、每次迭代过程中，固定Generator，Discriminator总能达到最优），这样构造的GANs会收敛到纳什均衡解。基于此，在2014年的论文里面，作者提出的算法是，每次迭代过程包括两个步骤：更新k次Discriminator（k>=1）；更新1次Generator。也就是，应该让Discriminator学得更充分一些。PS：2016 NIPS tutorial中，作者指出，每次迭代对博弈双方同时进行（随机）梯度下降，在实际操作中效果最好。

然而，这样定义的零和博弈在实际中效果并不好，实际训练的时候，Discriminator仍然采用前面介绍的loss，但Generator采用下面的loss：



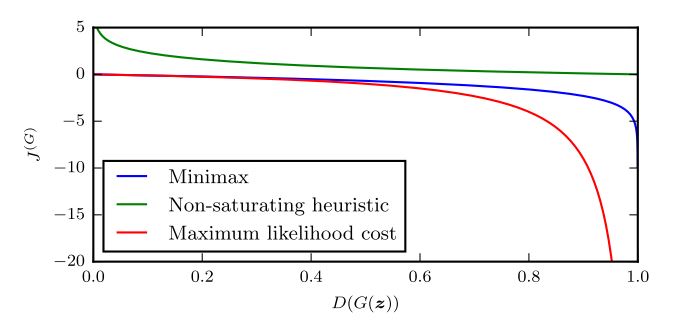
也就是说，Generator的loss只考虑它自身的交叉熵，而不用全局的交叉熵。这样做能够确保任一博弈失利的一方仍有较强的梯度以扭转不利局面。作者称采用这个loss的博弈为Non-saturating heuristic game。采用这个版本的loss，整个博弈不再是零和博弈，没有理论保证它会收敛达到纳什均衡。

**最大似然博弈GANs**

原始版本的GANs模型并不是通过最大化似然函数（也即最小化KL散度）来训练的。Goodfellow在14年的另一篇论文（https://arxiv.org/abs/1412.6515）中证明了，若每次迭代Discriminator都达到最优，Generator采用下面的loss，则GANs等价于最大化似然函数：

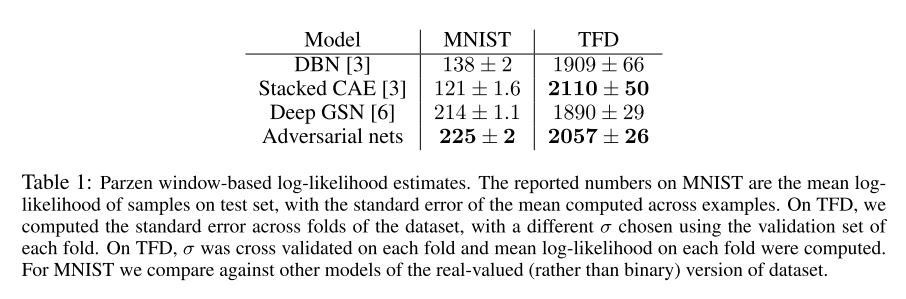


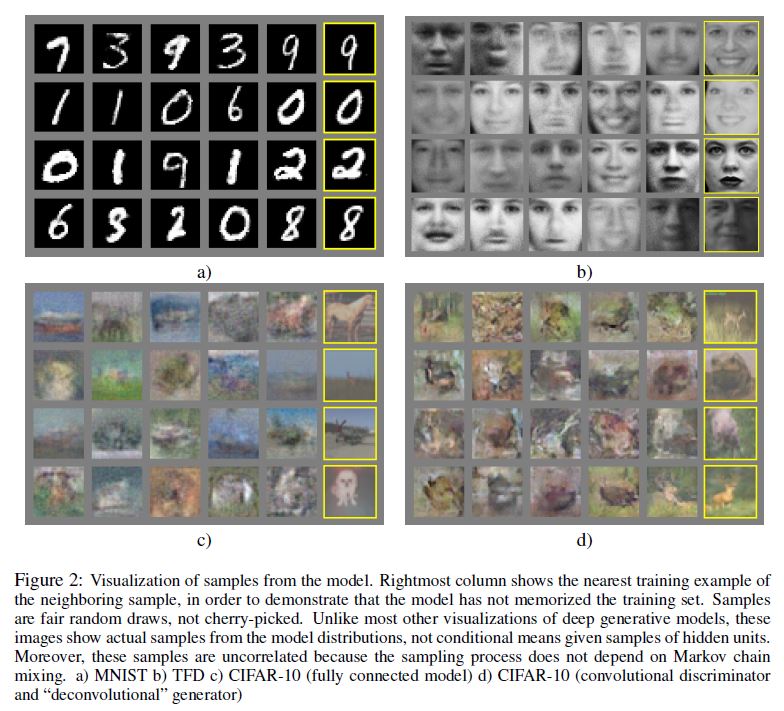
至此，本文总共出现了三种loss，它们的函数图像长这样：

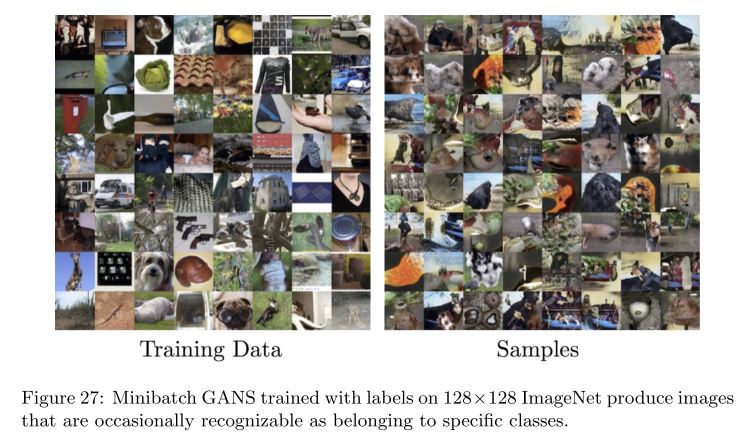


**GANs实验效果**

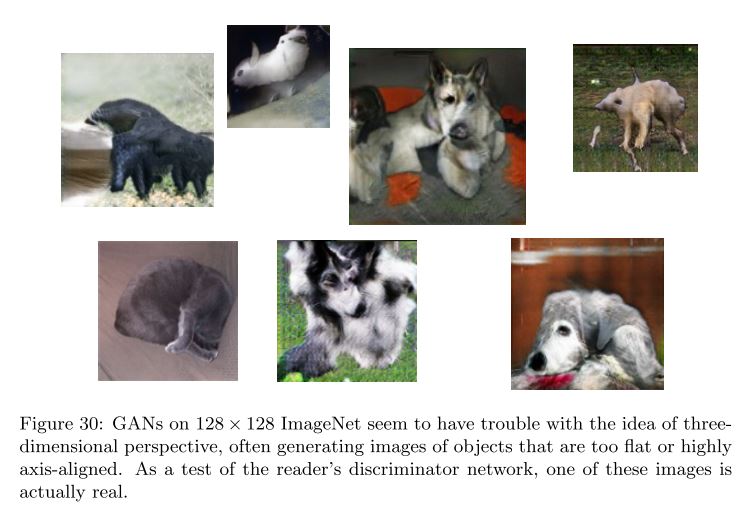
根据Goodfellow 2014年的论文，GANs在MNIST和TFD数据集上训练，产生的样本更接近于真实分布（参看下图中的Table 1）。然而，虽然效果更好，GANs也存在一些问题（如下面的Figure 29和30），话不多说，看图。











**GANs优缺点**

优点：

* 相比于FVBNs（全可见信念网络，Fully visible belief networks）模型，GANs能并行生成样本，不需要逐维产生。
* 相比于玻尔兹曼机（Boltzmann machines）、非线性ICA（non-linear ICA）等生成模型，GANs对Generator的设计的限制很少。
* 理论上保证了某些GANs能够收敛到纳什均衡。
* 最重要的是，相比于其他生成模型，GANs产生的样本更好。（这一条是主观判断...）

缺点：

* 训练GANs实际上是在找纳什均衡解，这比优化一个目标函数要困难。
* GANs生成的图像比较sharp，也就是说，它倾向于生成相似的图像。作者在2016 NIPS tutorial中指出，这个缺陷与采用何种KL散度作为loss无关，而可能是与训练过程有关。详细的讨论可以参看参考文献2的3.2.5和5.1.1章节。PS：这是一个非常重要的问题，如果想深入理解GANs，请务必阅读原文详细了解。

PS： 如果看完你还是没理解是怎么做的，那就看代码吧，网上的代码很多，点击阅读原文（Ian Goodfellow 2014年论文的代码）或者看文末给的代码链接（有simple版本的代码）。阅读源码对于理解文献的方法有很大的好处（前提是源码好理解...）；看完论文自己复现论文结果对于进一步理解论文，甚至改进方法都有很大的帮助。这也是我目前在做的事情。

**参考文献**

1. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in Neural Information Processing Systems. 2014: 2672-2680.
2. Goodfellow I. NIPS 2016 Tutorial: Generative Adversarial Networks[J]. arXiv preprint arXiv:1701.00160, 2016.
3. Goodfellow I J. On distinguishability criteria for estimating generative models[J]. arXiv preprint arXiv:1412.6515, 2014.

**GANs代码**

1. Theano+pylearn2版本（Goodfellow提供）： <https://github.com/goodfeli/adversarial>
2. Keras版本： <https://github.com/jhayes14/GAN>
3. Tensorflow版本（含GANs+VAEs+DRAW）： <https://github.com/ikostrikov/TensorFlow-VAE-GAN-DRAW>

**下期预告**

下期我们继续介绍生成模型。按照Goodfellow在2016 NIPS tutorial中的分类（仅考虑能使用最大化似然函数进行优化的模型），生成模型分为两大类：有显式密度函数、没有显式密度函数。显然GANs属于后者。

下期我们介绍GANs的后续发展：DCGAN（Deep Convolutional Generative Adversarial Networks ）。参考文献为：

Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. arXiv preprint arXiv:1511.06434, 2015.

