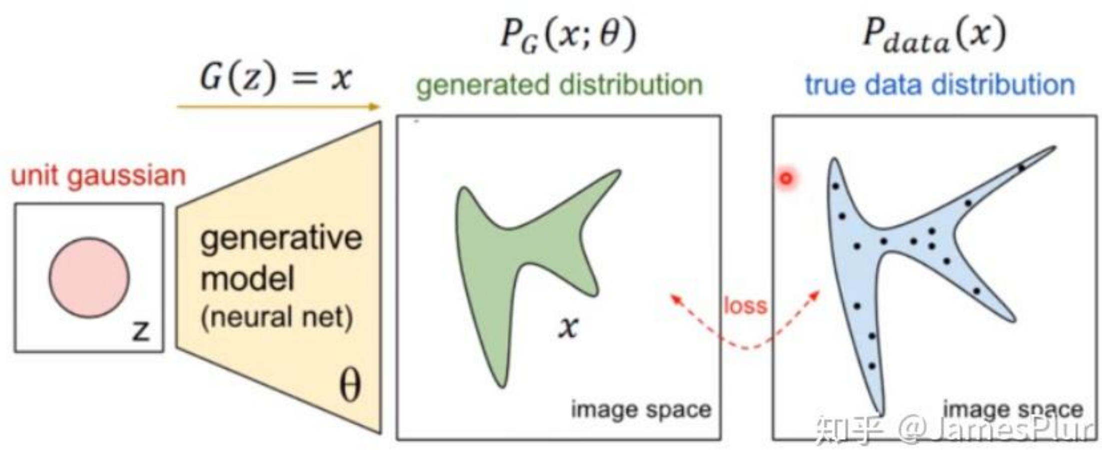
GAN的损失函数

理解生成对抗网络的关键在于理解GAN的损失函数

JS散度

GAN实际是通过对先验分布施加一个运算G, 来拟合一个新的分布



如果从传统的判别式网络的思路出发，只要选定合适的loss，就可以使生成分布和真实分布之间的距离尽可能逼近

KL散度经常用来衡量分布之间距离

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)}$$

但KL散度是不对称的。不对称意味着，对于同一个距离，观察方式不同，获取的loss也不同，那么整体loss下降的方向就会趋向于某个特定方向。这在GAN中非常容易造成模式崩塌，即生成数据的多样性不足

JS散度在KL散度的基础上进行了修正，保证了距离的对称性：

$$JS(P||Q) = \frac{1}{2} KL(P||\frac{P+Q}{2}) + \frac{1}{2} KL(Q||\frac{P+Q}{2})$$

实际上，无论KL散度还是JS散度，在直接用作loss时，都是难以训练的：由于分布只能通过取样计算，这个loss在每次迭代时都几乎为零

GAN loss的推导

GAN的训练方法，能够巧妙的解决这个问题：

先训练D，再训练G，二者相互对抗，直到收敛

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**
  for k steps **do**
    • Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
    • Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
    • Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

  end for
  • Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  • Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

end for
The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

在原始的GAN中，提出的loss是：

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

当G固定且运算可逆时（实际上这一点一般不成立，但不影响了解GAN的思想）：

$$E_{z\sim p_z(z)}[\log(1 - D(G(z)))] = E_{x\sim p_G(x)}[\log(1 - D(x))]$$

代入loss公式，进而有：

$$\begin{aligned} &\max_D V(D,G) \\ &= \max_D E_{x\sim p_{data}(x)}[\log D(x)] + E_{x\sim p_G(x)}[\log(1 - D(x)] \\ &= \max_D \int_x p_{data}(x)\log D(x) + p_g(x)\log(1 - D(x))dx \end{aligned}$$

对于积分区间内的每一个x，设被积函数为f 为：

$$f(D) = p_{data}(D)\log y + p_g(x)\log(1 - D)$$

注意这里x是固定的，变量是D。对f求导，得到当
$f(D) = p_{data}(D)\log y + p_g(x)\log(1-D)$时，f存在最大值。

由于被积函数的最大值对于任意x都成立，所以当 $D = \frac{p_{data}}{p_{data}+p_G}$时， V(D, G)有最大值

代入loss公式，有:

$$
\begin{aligned}
&\min_G \max_D V(D,G) \\
&= \min_G \int_x p_{data}(x)\log\frac{p_{data}}{p_{data}+p_G} + p_g(x)\log(\frac{p_G}{p_{data}+p_G})dx \\
&= -2log2 + \min_G \int_x p_{data}(x)\log\frac{p_{data}}{(p_{data}+p_G)/2} + p_g(x)\log(\frac{p_G}{(p_{data}+p_G)/
\end{aligned}
$$
$$
= -2log2 + \min_G [2JSD(P_{data}||P_G)]
$$

所以原始GAN的loss实际等价于JS散度

Wasserstein Loss

JS散度存在一个严重的问题：两个分布没有重叠时，JS散度为零，而在训练初期，JS散度是有非常大的可能为零的。所以如果D被训练的过于强，loss会经常收敛到-2log2而没有梯度

对于这个问题，WGAN提出了一个新的loss，Wasserstein loss， 也称作地球移动距离：

$$W(P_r, P_g) = \inf_{r\sim\prod(P_r,P_g)} E_{(x,y)\sim r}||x-y||$$

这个距离的直观含义是，将分布r移动到分布g所需要的距离，所以即使是两个分布没有重叠，这个loss也是有值的

可以证明，该距离可以转化为如下形式：

$$W(P_r, P_g) = \sup_{||f||_{L\leq 1}} E_{x\sim P_r}[f(x)] - E_{y\sim P_g}[f(y)]$$

其中f必须满足1-Lipschitz连续，即： $||f(x)-f(y)|| \leq ||x-y||$可以看到，符合1-Lipschitz连续的函数的梯度是受限的，可以有效的防止梯度的爆炸，使训练更加稳定

Spectral Normalization

对于GAN来说，f其实就是指的D或G，也就是神经网络。对于神经网络来说，一般是由一系列矩阵乘法复合而成的。可以证明，如果矩阵乘法这个运算满足1-Lipschitz连续，那么其复合运算也会满足1-Lipschitz连续，神经网络也就满足1-Lipschitz连续

对于矩阵变换A来说，它满足K-Lipschitz连续的充要条件是：$||Ax|| \le K||x||$ 对其等价变换有：

$$
\begin{aligned}
&||Ax|| \le K||x|| \\
&\langle Ax, Ax \rangle \le K^2 \langle x, x \rangle \\
&(Ax)^T Ax \le K^2 x^T x \\
&x^T A^T Ax - K^2 x^T x \le 0 \\
&x^T (A^T A - K^2 I)x \le 0
\end{aligned}
$$

假设 $A^T A$的特征向量构成的基底为 $v1, v2, ...$对应的特征值为 $\lambda 1, \lambda 2, ...$,则x可由特征向量表示：$\lambda 1, \lambda 2, ...$

那么有：

$$
\begin{aligned}
&x^T (A^T A - K^2 I)x \le 0 \\
&[\sum_i a_i v_i^T][\sum_j (\lambda_j - K^2) a_j v_j] \le 0
\end{aligned}
$$

只有当i 不等于j时，式子不为零, 且 $v_i^T v_i = 1$

所以有：$\sum_i (\lambda_j - K^2) a_j^2 \le 0$

矩阵 $A^T A$是半正定矩阵，所有特征值都为非负，所以只要矩阵除以它最大的奇异值的开方，就可以满足1-Lipschitz连续。power iteration 是求奇异值的一种简便算法，

称这种除以最大奇异值的操作为spectral norm

Hinge loss

Hinge loss 是对地球移动距离的一种拓展

Hinge loss 最初是SVM中的概念，其基本思想是让正例和负例之间的距离尽量大，后来在Geometric GAN中，被迁移到GAN:

$$
\begin{aligned}
&L_D = E(max(0, 1 - D(x))) + E(max(0, 1 + D(G(z)))) \\
&L_G = -E(D(G(z)))
\end{aligned}
$$

对于D来说，只有当$D(x) < 1$ 的正向样本，以及$D(G(z)) > -1$的负样本才会对结果产生影响

也就是说，只有一些没有被合理区分的样本，才会对梯度产生影响

这种方法可以使训练更加稳定