GAN

Binary Classification

给定数据分布 $p_{data}(x,y), y \in \{0,1\}$，我们希望 fit Binary Classifier $p_\phi(y \mid x)$ 到 $p_{data}(y \mid x)$。通过 MLE 可以获得优化目标:

$$\phi^* = \underset{\phi}{\operatorname{argmax}}\, \mathbb{E}_{p_{data}(x|y)}[\log p_\phi(\, y \mid x\,)]$$

假设数据集是平衡的，即 $p_{data}(y) = Bern(0.5)$，则上述期望可以描述为:

$$\begin{aligned}\phi^* &= \underset{\phi}{\operatorname{argmax}}\{\mathbb{E}_{p_{data}(x|y=1)}[\log p_\phi(\, y=1 \mid x\,)] + \mathbb{E}_{p_{data}(x|y=0)}[\log p_\phi(\, y=0 \mid x\,)]\} \\ &= \underset{\phi}{\operatorname{argmax}}\{\mathbb{E}_{p_{data}(x|y=1)}[\log p_\phi(\, y=1 \mid x\,)] + \mathbb{E}_{p_{data}(x|y=0)}[1-\log p_\phi(\, y=1 \mid x\,)]\}\end{aligned}$$

上述任务也被称为 Binary Cross Entropy（BCE）Loss。

GAN

我们定义 Generator 为p_θ，期望其能生成和p_{data}近似的数据。Discriminator 为$p_\phi(\, y \mid x\,) = D_\phi(x)$，用于分辨数据。因此我们可以构建一个 Binary Classification 问题:

$$\tilde{p}(x,y) = \tilde{p}(x \mid y)\tilde{p}(y), \quad \tilde{p}(y) = Bern(0.5), \quad \tilde{p}(x \mid y) = \begin{cases} p_{data}(x \mid y) & y=1 \\ p_\theta(x \mid y) & y=0 \end{cases}$$

Discriminator 为$p_\phi(\, y \mid x\,) = D_\phi(x)$，其优化目标为:

$$\phi^* = \underset{\phi}{\operatorname{argmax}}\, \underbrace{\{\mathbb{E}_{p_{data}(x)}[\log D_\phi(x)] + \mathbb{E}_{p_\theta(x)}[1-\log D_\phi(x)]\}}_{\mathcal{L}(\theta,\phi)}$$

而 Generator $p_\theta(x) = G_\theta$

$$\begin{aligned}\theta^* &= \underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{p_\theta(x)}[1-\log D_\phi(x)] \\ &= \underset{\theta}{\operatorname{argmin}}\, \underbrace{\{\mathbb{E}_{p_{data}(x)}[\log D_\phi(x)] + \mathbb{E}_{p_\theta(x)}[1-\log D_\phi(x)]\}}_{\mathcal{L}(\theta,\phi)}\end{aligned}$$

因此优化目标为

$$\theta^*, \phi^* = \min_\theta \max_\phi \mathcal{L}(\theta,\phi)$$

使用 MC 近似:

$$\mathbb{E}_{p_\theta(x)}[1-\log D_\phi(x)] \approx 1-\log D_\phi(x)\,, x \sim p_\theta(x)$$

我们可以用 $p(z)$ 隐式定义$p_\theta(x)$:

$$x \sim p_\theta(x) \iff z \sim p(z), x = G_\theta(z)$$

通常来说，我们认为 z 是一个随机噪声，即 $z = \epsilon, p(z) = \mathcal{N}(z; 0, \mathbf{I})$

Equivalence to JSD (Jensen-Shannon divergence) Minimisation

$$\begin{aligned}
\mathcal{L}(\theta,\phi) &= \mathbb{E}_{p_{data}(x)}[\log D_\phi(x)] + \mathbb{E}_{p_\theta(x)}[1-\log D_\phi(x)] \\
&= \int_x p_{data}(x)\log D_\phi(x)\ \ dx + \int_x p_\theta(x)\,[1-\log D_\phi(x)]dx \\
&= \int_x \{p_{data}(x)\log D_\phi(x) + p_\theta(x)[1-\log D_\phi(x)]\}\ \ dx \\
&= \int_x \{p_{data}(x)\log D_\phi(x) - p_\theta(x)\log D_\phi(x) + p_\theta(x)\}\ \ dx \\
\frac{\partial \mathcal{L}}{\partial \phi} &= \frac{\partial \mathcal{L}}{\partial D_\phi(x)}\frac{\partial D_\phi(x)}{\partial \phi} \\
\nabla_\phi \mathcal{L}(\theta,\phi) &= \int_x \left(\frac{p_{data}(x)}{D_\phi(x)} - \frac{p_\theta(x)}{1-D_\phi(x)}\right)\nabla_\phi D_\phi(x)dx
\end{aligned}$$

直接优化！（假设 Discriminator 有无限 capacity！）

$$\begin{aligned}
\nabla_\phi \mathcal{L}(\theta,\phi) &= 0 \\
\int_x \left(\frac{p_{data}(x)}{D_\phi(x)} - \frac{p_\theta(x)}{1-D_\phi(x)}\right)\nabla_\phi D_\phi(x)dx &= 0 \\
\frac{p_{data}(x)}{D_\phi(x)} - \frac{p_\theta(x)}{1-D_\phi(x)} &= 0 \\
D_\phi(x) &= \frac{p_{data}(x)}{p_\theta(x)+p_{data}(x)}
\end{aligned}$$

代入原式，则 $\mathcal{L}(\theta,\phi)$ 变为$\mathcal{L}(\theta,\phi^*(\theta))$

$$\begin{aligned}
\mathcal{L}(\theta,\phi) &= \mathbb{E}_{p_{data}(x)}[\log D_\phi(x)] + \mathbb{E}_{p_\theta(x)}[1-\log D_\phi(x)] \\
\mathcal{L}\big(\theta,\phi^*(\theta)\big) &= \mathbb{E}_{p_{data}(x)}\left[\log\frac{p_{data}(x)}{p_\theta(x)+p_{data}(x)}\right] + \mathbb{E}_{p_\theta(x)}\left[1-\log\frac{p_{data}(x)}{p_\theta(x)+p_{data}(x)}\right] \\
&= \mathbb{E}_{p_{data}(x)}\left[\log\frac{p_{data}(x)}{p_\theta(x)+p_{data}(x)}\right] + \mathbb{E}_{p_\theta(x)}\left[\cancel{1+}\log\frac{p_\theta(x)}{p_\theta(x)+p_{data}(x)}\right]
\end{aligned}$$

$$\begin{aligned}
&= \mathbb{E}_{p_{data}(x)}\left[\log\frac{p_{data}(x)}{\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)}+\log\frac{1}{2}\right] + \mathbb{E}_{p_\theta(x)}\left[\log\frac{p_\theta(x)}{\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)}+\log\frac{1}{2}\right] \\
&= \mathbb{E}_{p_{data}(x)}\left[\log\frac{p_{data}(x)}{\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)}\right] + \mathbb{E}_{p_\theta(x)}\left[\log\frac{p_\theta(x)}{\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)}\right] - 2\log 2 \\
&= KL\left[p_{data}(x)\|\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)\right] \ + KL\left[p_\theta(x)\|\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)\right] - 2\log 2 \\
&= 2\underbrace{\left\{\frac{1}{2}KL\left[p_{data}(x)\|\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)\right] + \frac{1}{2}KL\left[p_\theta(x)\|\frac{1}{2}\big(p_\theta(x)+p_{data}(x)\big)\right]\right\}}_{\mathrm{JSD}[p_{data}(x)\|p_\theta(x)]} - 2\log 2 \\
&= 2\mathrm{JSD}[p_{data}(x)\|p_\theta(x)] - 2\log 2
\end{aligned}$$

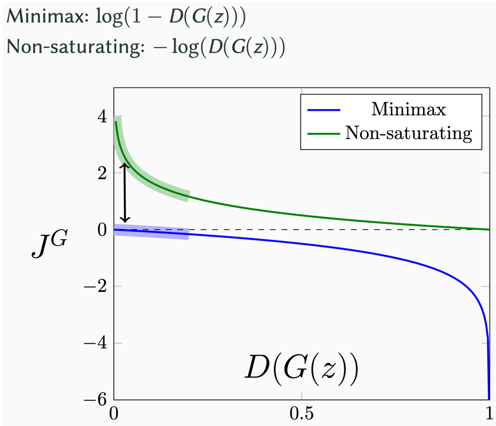
考虑 JSD 是一个有效的 Divergence，那么最优情况当且仅当 $p_{data} = p_\theta$。即 $D_\phi(x) = \frac{1}{2} - 0.5$，训练停止时刻为辨别器返回 0.5。

Alternative Loss for the Generator

优化 Generator 本质上在优化：

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{p_\theta(x)}[1 - \log D_\phi(x)]$$
$$\Leftrightarrow \underset{\theta}{\operatorname{argmax}}\, \mathbb{E}_{p_\theta(x)}[\log D_\phi(x)]$$

第二个式子是替代（Alternative）的优化目标。



D_ϕ 在 GAN 训练初期往往有接近完美的分类性能（因为在这个阶段"假"数据的质量很差）。
在这种情况下，有 $x \sim p_\theta(x), D_\phi(x) \approx 0$。同时假设生成模型由 $z \sim p(z), x = G_\theta(z)$ 隐式定义。

$D_\phi(x)$ 通常在最后一层使用 sigmoid 激活函数 $\sigma(t) = (1 + \exp[-t])^{-1}$ 来定义，即 $D_\phi(x) = \sigma(d_\phi(x))$。这意味着当$d_\phi(x) \to -\infty$时，$D_\phi(x) \approx 0$ 。
所以在 GAN 训练开始时，对于 $x \sim p_\theta(x)$，有$d_\phi(x) \to -\infty$。因此，这两个目标相对于 θ 的梯度是

$$\begin{aligned}
&\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p_{\boldsymbol{\theta}}(\boldsymbol{x})}[\log(1-D_{\boldsymbol{\phi}}(\boldsymbol{x}))] = -\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p(\boldsymbol{z})}[\log(1+\exp[d_{\boldsymbol{\phi}}(G_{\boldsymbol{\theta}}(\boldsymbol{z}))])] = -\mathbb{E}_{p(\boldsymbol{z})}[\underbrace{D_{\boldsymbol{\phi}}(G_{\boldsymbol{\theta}}(\boldsymbol{z}))}_{\approx 0}\nabla_{\boldsymbol{\theta}} d_{\boldsymbol{\phi}}(G_{\boldsymbol{\theta}}(\boldsymbol{z}))], \\
&\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p_{\boldsymbol{\theta}}(\boldsymbol{x})}[\log D_{\boldsymbol{\phi}}(\boldsymbol{x})] = -\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p(\boldsymbol{z})}[\log(1+\exp[-d_{\boldsymbol{\phi}}(G_{\boldsymbol{\theta}}(\boldsymbol{z}))])] = \mathbb{E}_{p(\boldsymbol{z})}[\underbrace{(1-D_{\boldsymbol{\phi}}(G_{\boldsymbol{\theta}}(\boldsymbol{z})))}_{\approx 1}\nabla_{\boldsymbol{\theta}} d_{\boldsymbol{\phi}}(G_{\boldsymbol{\theta}}(\boldsymbol{z}))].
\end{aligned} \tag{13}$$

因此原目标函数在最早期很容易出现 Gradient Vanishing 的问题。因此被称为“non-saturated objective”。非饱和目标。

另一个理由是在给出最优判别器的情况下，推导出生成器的最优解。定义 $f(t) = \log(1 + t^{-1}) - \log 2$. $f(t)$ 是 convex 且 $f(1) = 0$。因此可以定义 f-divergence:

$$\begin{aligned}
\mathrm{D}_f[p_{\boldsymbol{\theta}}(\boldsymbol{x})||p_{\text{data}}(\boldsymbol{x})] &:= \int p_{\boldsymbol{\theta}}(\boldsymbol{x}) f\left(\frac{p_{\text{data}}(\boldsymbol{x})}{p_{\boldsymbol{\theta}}(\boldsymbol{x})}\right) d\boldsymbol{x} \\
&= \int p_{\boldsymbol{\theta}}(\boldsymbol{x}) \log\left(1 + \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x})}{p_{\text{data}}(\boldsymbol{x})}\right) d\boldsymbol{x} - \log 2 \\
&= -\mathbb{E}_{p_{\boldsymbol{\theta}}(\boldsymbol{x})}[\log D_{\boldsymbol{\phi}^*(\boldsymbol{\theta})}(\boldsymbol{x})] - \log 2.
\end{aligned}$$

这表示最大化“non-saturated objective”等价于最小化 f-divergence。

Conditional GAN

我们的目标是使用 LVM 拟合 $p_{data}(x \mid y)$:

$$p_\theta(\, x \mid y \,) = \int_z p_\theta(\, x \mid y, z \,) p(z) \, dz$$

令 $p(z) = \mathcal{N}(z; 0, \mathbf{I})$, GAN 并没有和 VAE 类似，显式定义 $p(x \mid y, z)$，GAN 使用如下逻辑:

$$x \sim p_\theta(\, x \mid y, z \,) \iff z \sim p(z), x = G_\theta(z, y)$$

类似的，其优化目标为

$$\min_\theta \max_\phi \{ \mathbb{E}_{p_{data}(x)}[\log D_\phi(x)] + \mathbb{E}_{p_\theta(x)}[1 - \log D_\phi(x)] \}$$

$$\Downarrow$$

$$\min_\theta \max_\phi \{ \mathbb{E}_{p_{data}(x,y)}[\log D_\phi(x, y)] + \mathbb{E}_{p_\theta(x|y)p_{data}(y)}[1 - \log D_\phi(x, y)] \}$$

类似的，使用 MC 近似后:

$$\mathbb{E}_{p_\theta(x|y)p_{data}(y)}[1 - \log D_\phi(x, y)] \approx \log[1 - D_\phi(G_\theta(z, y), y)], \qquad z \sim p(z), y \sim p_{data}(y)$$

类似的，optimal discriminator:

$$D_\phi(x, y) = \frac{p_{data}(x, y)}{p_\theta(x \mid y) p_{data}(y) + p_{data}(x, y)}$$

优化 θ 等价于优化 $\mathrm{JSD}[p_{data}(x, y) \parallel p_\theta(x \mid y) p_{data}(y)]$

Wasserstein Distance

是最优传输理论中的一个重要概念，**其目标是寻找将一个 distribution 转换为另一个 distribution 的最低成本方法**。Wasserstein 距离的对偶形式是通过从集合$\mathcal{F} = \{f : \|f\|_L \leq 1\}$ (1-Lipschitz 函数集合)中取最优测试函数来定义的：

$$W_2[p, q] = \sup_{\|f\|_L \leq 1} \mathbb{E}_p[f(x)] - \mathbb{E}_q[f(x)]$$

函数 $f : \mathbb{R}^d \to \mathbb{R}$ 被称为 l-Lipschitz（被标注为 $\|f\|_L \leq l$），如果:

$$|f(x_1) - f(x_2)| \leq l \|x_1 - x_2\|_2, \qquad \forall x_1, x_2 \in \mathbb{R}^d$$

如果处处可微，则

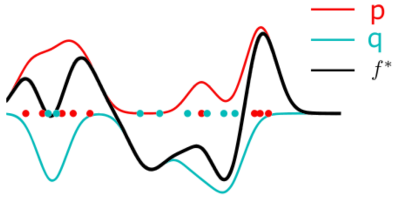
$$\|f\|_L \leq 1 \iff \|\nabla_x f(x)\|_2 \leq 1, \qquad \forall x \in \mathbb{R}^d$$

为什么要用 Wasserstein Distance

你有两杯不同的果汁(代表两个分布 p 和 q)
test function 就像是各种测试手段：可能是测甜度的、测酸度的、测浓度的等
你用这些测试手段分别测两杯果汁，看看最大的差异能有多大
但是测试手段不能太极端（这就是利普希茨条件的作用），必须是"合理"的测试方法

利普希茨条件限制了函数变化的"剧烈程度"。
如果一个函数满足利普希茨条件，那么对于函数上的任意两个点，它们的纵向距离（函数值之差）不能超过它们横向距离（自变量之差）的某个固定倍数。这个倍数就是利普希茨常数。

Integral probability metrics (IPMs)



Wasserstein Distance 是 distribution distance IPM 家族的一部分。

Definition 1. *(Integral probability metric (IPM)) Given a set of test functions $\mathcal{F}$, consider the following quantity:*

$$\mathrm{D}[p,q] = \sup_{f \in \mathcal{F}} |\mathbb{E}_p[f(\boldsymbol{x})] - \mathbb{E}_q[f(\boldsymbol{x})]|, \tag{28}$$

where $|\cdot|$ denotes a norm in the output space of f. If $\mathcal{F}$ is sufficiently large such that $\mathrm{D}[p,q] = 0$ iff. $p = q$, then $\mathrm{D}[p,q]$ is said to be an integral probability metric defined by the test functions in $\mathcal{F}$.

考虑一种通过比较分布的 moments（如均值、方差、峰度 kurtosis 等）来比较分布的策略。简单来说，如果两个分布 p 和 q 在所有阶数上都具有相同的 moments，那么 p 和 q 应该是相同的。因此，要检验 p 和 q 是否相同，可以找到最佳的 moment，或者更广义地说，找到最佳的测试函数 f，使其能最大程度地区分 p 和 q。如果这样的最优测试函数仍然无法区分 p 和 q，那么这两个分布就是相同的。

这种直观理解可以在上图中进行可视化。我们从可视化中看到，最优测试函数 f^* 在 $p(x) > q(x)$ 的区域取正值，反之亦然。换句话说，最优测试函数不仅告诉我们 p 是否等于 q，还提供了关于 p 和 q 如何相互不同的信息。这对于 IPM 在对抗性学习中的应用是一个有用的性质：由于 f^* 详细描述了 p 和 q 之间的差异，我们可以以一种有指导的方式优化 q 分布，使其逼近目标分布 p。实际上，IPM 的各种版本已被用作 GAN 文献中的优化目标。

在 W-GANs，可以把 Discriminator 可以看作是一个 parameterised 的 test function $f := D_\phi$

考虑 $\|f\|_L \leq 1 \iff \|\nabla_x f(x)\|_2 \leq 1,\ \forall x \in \mathbb{R}^d$

$$W_2[p,q] = \sup_{\|f\|_L \leq 1} \mathbb{E}_p[f(x)] - \mathbb{E}_q[f(x)]$$

因此可以改写为

$$\begin{aligned} d_W &= \mathbb{E}_{p_{data}(x)}[f(x)] - \mathbb{E}_{p_\theta(x)}[f(x)] \\ &= \mathbb{E}_{p_{data}(x)}[D_\phi(x)] - \mathbb{E}_{p_\theta(x)}[D_\phi(x)] \\ s.t. &\left\|\nabla_x D_\phi(x)\right\|_2 \leq 1, \qquad \forall x \in \mathbb{R}^d \end{aligned}$$

因此可以把优化看作最小化 Wasserstein Distance

$$\begin{aligned} &\min_\theta \max_\phi \{\mathbb{E}_{p_{data}(x)}[D_\phi(x)] - \mathbb{E}_{p_\theta(x)}[D_\phi(x)]\} \\ &\text{subject to } \left\|\nabla_x D_\phi(x)\right\|_2 \leq 1, \qquad \forall x \in \mathbb{R}^d \end{aligned}$$

但是对所有的 x 计算 constraint 是不现实的，因此 point-wise constraint 被下列 constraint 替代:

$$\mathbb{E}_{\hat{p}(x)}(\left\|\nabla_x D_\phi(x)\right\|_2 - 1)^2 = 0$$

辅助插值分布（auxiliary “interpolation” distribution） $\hat{p}(x)$ 使用如下生成过程生成:

$$\begin{aligned} &x \sim \hat{p}(x) \iff \\ &x_d \sim p_{data}(x), x_g \sim p_\theta(x), \alpha \sim \text{Uniform}([0,1]), x = \alpha x_d + (1-\alpha) x_g \end{aligned}$$

supp: 对于一个随机变量 X，其支撑集是指该随机变量可能取值的所有点的集合。

最初的约束只需要在 $p_{data}(x)$ 和 $p_\theta(x)$ 的支撑集上评估判别器，所以只需要在这些支撑集的并集上保证 $\left\|\nabla_x D_\phi(x)\right\|_2 \leq 1$ 这个约束成立。而且可以证明，目标函数的最优判别器在这个并集上会满足 $\left\|\nabla_x D_\phi(x)\right\|_2 = 1$。

现在，替代约束要求在 $\hat{p}(x)$ 的支撑集上满足 $\left\|\nabla_x D_\phi(x)\right\|_2 = 1$。根据构造方法，我们知道 $p_{data}(x)$ 和 $p_\theta(x)$ 的支撑集的并集是 $\hat{p}(x)$支撑集的子集。这就说明，如果满足了约束，那么原始约束也就自然满足了。

$$\min_\theta \max_\phi \left\{ \mathbb{E}_{p_{data}(x)}[D_\phi(x)] - \mathbb{E}_{p_\theta(x)}[D_\phi(x)] - \lambda \underbrace{\mathbb{E}_{\hat{p}(x)}(\left\|\nabla_x D_\phi(x)\right\|_2 - 1)^2}_{\text{GP}} \right\}$$

对于带有替代约束的目标函数，可以用拉格朗日乘数法求解（$\lambda \geq 0$），这就得到了 WGAN-GP（Wasserstein GAN with gradient penalty）的目标函数。

GP 的目的是为了确保判别器满足 1-Lipschitz 约束： $\left\|\nabla_x D_\phi(x)\right\|_2 \approx 1$。因此只在判别器时候会训练。

Algorithm 1 WGAN with gradient penalty. We use default values of $\lambda = 10$, $n_{\text{critic}} = 5$, $\alpha = 0.0001$, $\beta_1 = 0$, $\beta_2 = 0.9$.

Require: The gradient penalty coefficient λ, the number of critic iterations per generator iteration n_{critic}, the batch size m, Adam hyperparameters α, β_1, β_2.
Require: initial critic parameters w_0, initial generator parameters θ_0.

```
while θ has not converged do
    for t = 1, ..., n_critic do
        for i = 1, ..., m do
            Sample real data x ~ P_r, latent variable z ~ p(z), a random number ε ~ U[0, 1].
            x̃ ← G_θ(z)
            x̂ ← εx + (1 − ε)x̃
            L^(i) ← D_w(x̃) − D_w(x) + λ(‖∇_x̂ D_w(x̂)‖_2 − 1)^2
        end for
        w ← Adam(∇_w (1/m) Σ_{i=1}^m L^(i), w, α, β_1, β_2)
    end for
    Sample a batch of latent variables {z^(i)}_{i=1}^m ~ p(z).
    θ ← Adam(∇_θ (1/m) Σ_{i=1}^m −D_w(G_θ(z)), θ, α, β_1, β_2)
end while
```