

生成模型概览

变分自编码器 (Variational Auto Encoder)

详情可见[从狭义EM到变分自编码器]

生成式对抗网络 (Generative Adversarial Nets)

动机

深度学习在判别式模型 (Discriminative Models) 上已经取得了巨大成功 (比如图像分类、语音识别)，但在生成式模型 (Generative Models) 领域却一直步履维艰。

在 GAN 之前，主流的生成模型 (如受限玻尔兹曼机 RBM、深度信念网络 DBN) 主要依赖于显式密度估计 (Explicit Density Estimation)。这意味着模型试图直接写出并最大化真实数据的概率密度函数 $p(x)$ 。

- 1、计算上的灾难：对于像高分辨率图像这样极其复杂、高维的数据，其真实的概率分布极其复杂。要计算这个概率分布的归一化常数 (配分函数, Partition Function) 在计算上是不可解的 (Intractable)。
- 2、近似方法的低效：为了绕过这个计算障碍，研究者们不得不使用极其复杂的数学近似方法，最典型的就是马尔可夫链蒙特卡洛采样 (MCMC)。MCMC 在高维空间中极其缓慢，且很难判断是否已经收敛，导致这些生成模型训练极其困难、耗时，且生成的图像往往非常模糊。

既然直接计算概率密度 $p(x)$ 这么困难，能不能绕过这个数学计算，直接建立一个“黑盒”，只要这个黑盒能源源不断地生成看起来像真实数据的东西就可以了？这就是隐式生成建模 (Implicit Generative Modeling) 的思想。

原理

为了学习生成器在数据 \mathbf{x} 上的分布 p_g ，定义输入噪声变量的先验分布为 $p_z(\mathbf{z})$ ，然后将映射到数据空间的函数表示为 $G(\mathbf{z}; \theta_g)$ ，其中 G 是由参数为 θ_g 的多层感知机表示的可微函数。还定义了第二个多层感知机 $D(\mathbf{x}; \theta_d)$ ，它输出一个单一的标量。代表输入数据 \mathbf{x} 来自真实数据 (而非生成器分布 p_g) 的概率。

训练 D 的目标是最大化其为真实的训练样本和来自 G 的生成样本分配正确标签的概率；同时训练 G 来最小化 $\log(1 - D(G(\mathbf{z})))$ ，以欺骗 D 。换言之， D 和 G 正在进行以下带价值函数 $V(G, D)$ 的双人极大极小博弈

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

在实践中使用迭代的数值方法来实现这个博弈。

注意：在训练初期，生成器 G 产生的都是明显的假图 (噪声)，而判别器 D 很容易就能分辨出真伪。因此， $D(G(\mathbf{z}))$ 的输出会非常接近于 0。函数 $f(x) = \log(1 - x)$ 在 $x \rightarrow 0$ 时导数是 $-\frac{1}{1-x}$ ，当 $x \approx 0$ 时，导数约为 -1 。这意味着，当生成器表现最差的时候，它获得的反馈信号 (梯度) 却非常平缓。梯度太小，导致生成器很难快速学习和改进，因此建议 G 最大化 $\log(D(G(\mathbf{z})))$ 。此时优化方向与原损失函数一致，纳什均衡唯一解也没有改变。

算法1:

for 训练迭代次数 do

for k 步 do

从噪声先验分布 $p_g(\mathbf{z})$ 中采样包含 m 个噪声样本的小批量数据 $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$ 。

从数据生成分布 $p_{\text{data}}(\mathbf{x})$ 中采样包含 m 个真实样本的小批量数据 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 。

通过沿随机梯度的上升方向来更新判别器（最大化收益）：

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

end for

从噪声先验分布 $p_g(\mathbf{z})$ 中采样包含 m 个噪声样本的小批量数据 $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$

通过沿随机梯度的下降方向来更新生成器（最小化损失）：

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

end for

命题 1:

对于一个固定的生成器 G （即生成的假数据分布 p_g 是固定的），最优的判别器 $D_G^*(\mathbf{x})$ 的解析解为

$$D_G^*(\mathbf{x}) = \frac{p_{\text{data}}(\mathbf{x})}{p_{\text{data}}(\mathbf{x}) + p_g(\mathbf{x})}$$

将价值函数展开

$$V(G, D) = \int_{\mathbf{x}} p_{\text{data}}(\mathbf{x}) \log(D(\mathbf{x})) d\mathbf{x} + \int_{\mathbf{z}} p_{\mathbf{z}}(\mathbf{z}) \log(1 - D(g(\mathbf{z}))) d\mathbf{z}$$

噪声 \mathbf{z} 通过生成器 $g(\mathbf{z})$ 映射后，产生的就是假样本 \mathbf{x} 。而这些假样本 \mathbf{x} 服从的分布正是 $p_g(\mathbf{x})$ 。因此，“在隐空间 \mathbf{z} 上积分”与“在生成空间 \mathbf{x} 上积分”是完全等价的。可以把第二项直接替换为关于 \mathbf{x} 的积分：

$$\int_{\mathbf{x}} p_g(\mathbf{x}) \log(1 - D(\mathbf{x})) d\mathbf{x}$$

合并为

$$V(G, D) = \int_{\mathbf{x}} [p_{\text{data}}(\mathbf{x}) \log(D(\mathbf{x})) + p_g(\mathbf{x}) \log(1 - D(\mathbf{x}))] d\mathbf{x}$$

积分号里面的表达式就可以简写为一个关于 $y = D(\mathbf{x})$ 的函数

$$f(y) = a \log(y) + b \log(1 - y)$$

$D^*(\mathbf{x}) = \frac{p_{\text{data}}(\mathbf{x})}{p_{\text{data}}(\mathbf{x}) + p_g(\mathbf{x})}$ 对于任意 \mathbf{x} 都达到最大，即为最优判别器。

当判别器达到最优时，生成器 G 面临的损失：

$$\begin{aligned} C(G) &= \max_D V(G, D) = V(G, D_G^*) \\ &= \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} \left[\log \frac{p_{\text{data}}(\mathbf{x})}{p_{\text{data}}(\mathbf{x}) + p_g(\mathbf{x})} \right] + \mathbb{E}_{\mathbf{x} \sim p_g} \left[\log \frac{p_g(\mathbf{x})}{p_{\text{data}}(\mathbf{x}) + p_g(\mathbf{x})} \right] \end{aligned}$$

定理1:

目标函数 $C(G)$ 的全局最小值是 $-\log 4$ 。当且仅当生成的数据分布与真实分布完全一致 (即 $p_g = p_{\text{data}}$) 时, 才能达到这个最小值。

易知 $p_g = p_{\text{data}}$ 时取得 $C(G) = -\log 4$, 将其从式中提出

$$\begin{aligned} C(G) &= -\log 4 + KL\left(p_{\text{data}} \parallel \frac{p_{\text{data}} + p_g}{2}\right) + KL\left(p_g \parallel \frac{p_{\text{data}} + p_g}{2}\right) \\ &= -\log 4 + 2 \cdot JSD(p_{\text{data}} \parallel p_g) \end{aligned}$$

任何 JS 散度都是非负的 ($JSD \geq 0$), 并且当且仅当两个分布完全相等时, JS 散度才等于 0, 定理1得证。

Jensen-Shannon divergence: 给定两个概率分布 P 和 Q , 定义中间分布 $M = \frac{1}{2}(P + Q)$, 则

$$JSD(P \parallel Q) = \frac{1}{2}KL(P \parallel M) + \frac{1}{2}KL(Q \parallel M)$$

核心性质:

- 对称性: $JSD(P \parallel Q) = JSD(Q \parallel P)$
- 有界: $0 \leq JSD(P \parallel Q) \leq 1$, 不会像 KL 散度会发散到无穷大

命题2:

如果生成器 G 和判别器 D 具有足够的能力, 并且在算法 1 的每一步中, 判别器都被允许在给定生成器的情况下达到其最优, 同时 p_g 也进行更新以改善目标函数 $\mathbb{E}_{\mathbf{x} \sim p_{\text{data}}}[\log D_G^*(\mathbf{x})] + \mathbb{E}_{\mathbf{x} \sim p_g}[\log(1 - D_G^*(\mathbf{x}))]$, 则 p_g 能收敛到 p_{data} 。(详情见[2014 Generative Adversarial Nets])

优势

- 1、抛弃了 MCMC 采样: GAN 的生成器 G 只需要接收一个简单的随机噪声 z , 通过神经网络前向传播一次, 就能生成样本 $G(z)$ 。这不需要任何复杂的马尔可夫链迭代, 采样速度极快。
- 2、纯粹基于反向传播训练: 在 GAN 的框架里, 我们不需要去近似复杂的下界 (如 VAE), 也不需要配分函数。判别器 D 提供了一个灵活的、可学习的目标函数。只要 G 和 D 都是可微的神经网络, 整个系统就可以完全依赖标准的反向传播算法进行端到端的优化。
- 3、生成质量的飞跃: 传统的基于最大似然估计的模型, 往往会因为试图覆盖数据分布的所有模式 (Mode) 而生成平均化、模糊的图像。而 GAN 的对抗机制强制生成器必须产生能够“以假乱真”的清晰细节, 否则就会被判别器识破。

劣势

- 1、训练极不稳定: GAN 的本质是寻找一个纳什均衡, 但在高维度的神经网络中, 使用基于梯度的优化算法极难真正达到这个均衡点。生成器和判别器经常陷入无休止的震荡中, 参数更新失去方向, 导致模型无法收敛; 维持生成器和判别器的平稳更新也需要人为调整。
- 2、模式崩溃: 生成器在博弈中只生成某一种或几种特定的样本 (比如只生成某一种角度、同一种颜色的猫), 就能稳定地欺骗判别器。结果生成器停止探索数据的多样性, 导致生成的样本高度同质化。

扩散模型 (Diffusion Model)

动机

DDPM 的提出旨在解决当时生成模型（如 GAN 训练不稳定、VAE 样本质量受限且易过度平滑）在稳定性与生成质量之间难以兼顾的问题，通过将数据生成过程建模为一个逐步加噪再逐步去噪的马尔可夫链，并以变分推断为理论基础进行优化，从而在保证训练稳定性的同时实现高保真样本生成。

原理

扩散模型是一类隐变量模型，形式为 $p_\theta(\mathbf{x}_0) := \int p_\theta(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T}$ ，其中 $\mathbf{x}_1, \dots, \mathbf{x}_T$ 是与数据 $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ 具有相同维度的隐变量。联合分布 $p_\theta(\mathbf{x}_{0:T})$ 被称为逆向过程 (reverse process)，它被定义为一个具有可学习高斯转移概率的马尔可夫链，起始于 $p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$ 。

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T) \prod_{t=1}^T p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t), \quad p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t))$$

扩散模型区别于其他类型隐变量模型的地方在于，其近似后验 $q(\mathbf{x}_{1:T} | \mathbf{x}_0)$ （被称为前向过程或扩散过程）被固定为一个马尔可夫链，该链根据方差表 (variance schedule) β_1, \dots, β_T 逐渐向数据中添加高斯噪声，即该过程为固定程序而非可学习对象，在 VAE 中推断网络（近似后验 q ）是需要用神经网络去学习的

$$q(\mathbf{x}_{1:T} | \mathbf{x}_0) := \prod_{t=1}^T q(\mathbf{x}_t | \mathbf{x}_{t-1}), \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t} \mathbf{x}_{t-1}, \beta_t \mathbf{I})$$

训练是通过优化负对数似然的常规变分界（Variational Bound）来进行的（见[从狭义EM到变分自编码器]式 147）：

$$\mathbb{E}[-\log p_\theta(\mathbf{x}_0)] \leq \mathbb{E}_q \left[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T} | \mathbf{x}_0)} \right] = \mathbb{E}_q \left[-\log p(\mathbf{x}_T) - \sum_{t \geq 1} \log \frac{p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)}{q(\mathbf{x}_t | \mathbf{x}_{t-1})} \right] =: \mathcal{L}$$

前向过程的方差 β_t 可以通过重参数化来学习，也可以作为超参数保持恒定。逆向过程的表达能力部分由 $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ 中选择的高斯条件概率来保证，因为当 β_t 很小时，这两个过程具有相同的函数形式（见[2015 Deep unsupervised learning using nonequilibrium thermodynamics]，如果一个微小的前向扩散步骤是高斯分布，那么只要步长足够小，其时间反演（逆向过程）在数学上严格证明也是一个高斯分布。）。

前向过程的一个显著特性是，它允许在任意时间步 t 以解析形式采样 \mathbf{x}_t ：使用符号 $\alpha_t := 1 - \beta_t$ 和 $\bar{\alpha}_t := \prod_{s=1}^t \alpha_s$ ，得到下述公式：

$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I})$$

因此，通过随机梯度下降优化 \mathcal{L} 的随机项，可以实现高效的训练。进一步的改进来自于通过重写 \mathcal{L} 来降低方差。

$$\begin{aligned}
& \mathcal{L} \\
&= \mathbb{E}_q \left[-\log p(\mathbf{x}_T) + \sum_{t=1}^T \log \frac{q(\mathbf{x}_t | \mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)} \right] \\
&\quad \because q(\mathbf{x}_t | \mathbf{x}_{t-1}) = q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_0) = \frac{q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)q(\mathbf{x}_t | \mathbf{x}_0)}{q(\mathbf{x}_{t-1} | \mathbf{x}_0)} \\
&= \mathbb{E}_q \left[-\log p(\mathbf{x}_T) + \sum_{t=1}^T \log \frac{q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)q(\mathbf{x}_t | \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)q(\mathbf{x}_{t-1} | \mathbf{x}_0)} \right] \\
&= \mathbb{E}_q \left[-\log p(\mathbf{x}_T) + \sum_{t=1}^T \log \frac{q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)} + \sum_{t=1}^T \log \frac{q(\mathbf{x}_t | \mathbf{x}_0)}{q(\mathbf{x}_{t-1} | \mathbf{x}_0)} \right] \\
&\quad \because \sum_{t=1}^T \log \frac{q(\mathbf{x}_t | \mathbf{x}_0)}{q(\mathbf{x}_{t-1} | \mathbf{x}_0)} = \left(\log \frac{q(\mathbf{x}_1 | \mathbf{x}_0)}{q(\mathbf{x}_0 | \mathbf{x}_0)} \right) + \left(\log \frac{q(\mathbf{x}_2 | \mathbf{x}_0)}{q(\mathbf{x}_1 | \mathbf{x}_0)} \right) + \dots + \left(\log \frac{q(\mathbf{x}_T | \mathbf{x}_0)}{q(\mathbf{x}_{T-1} | \mathbf{x}_0)} \right) \\
&= \mathbb{E}_q \left[-\log p(\mathbf{x}_T) + \sum_{t=1}^T \log \frac{q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)} + \log q(\mathbf{x}_T | \mathbf{x}_0) \right] \\
&= \mathbb{E}_q \left[D_{KL}(q(\mathbf{x}_T | \mathbf{x}_0) || p(\mathbf{x}_T)) + \sum_{t=1}^T \log \frac{q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)} \right] \\
&= \mathbb{E}_q \left[\underbrace{D_{KL}(q(\mathbf{x}_T | \mathbf{x}_0) || p(\mathbf{x}_T))}_{\mathcal{L}_T} + \sum_{t>1} \underbrace{D_{KL}(q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) || p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t))}_{\mathcal{L}_{t-1}} - \log p_\theta(\mathbf{x}_0 | \mathbf{x}_1) \right]
\end{aligned}$$

上式使用 KL 散度直接将 $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ 与前向过程的后验概率进行比较，当以 \mathbf{x}_0 为条件时，这些后验概率是解析可求的（因为高斯分布的乘积和商必然也是一个高斯分布，通过配方可求其参数）

$$q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) = \frac{q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_0)q(\mathbf{x}_{t-1} | \mathbf{x}_0)}{q(\mathbf{x}_t | \mathbf{x}_0)} = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I})$$

其中 $\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\bar{\alpha}_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t$, $\tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$ 。因此，重写 \mathcal{L} 中的所有 KL 散度均为高斯分布间的比较，故可通过 Rao-Blackwell 化方法以解析形式进行计算，从而替代高方差的蒙特卡洛估计。

DDPM 原文忽略了前向过程方差 β_t 可以通过重参数化进行学习的事实，而是将它们固定为常数，因此，近似后验 q 没有任何可学习的参数，所以 \mathcal{L}_T 在训练期间是一个常数，可以被忽略。

对 $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$ 的选择时为了简化，将方差 $\boldsymbol{\Sigma}_\theta$ 设置为未经训练的、依赖于时间的常数 $\sigma_t^2 \mathbf{I}$ （与 VAE 的 decoder 类似）。实验上， $\sigma_t^2 = \beta_t$ 和 $\sigma_t^2 = \tilde{\beta}_t$ 取得了相似的结果。第一个选择对于数据分布为 $\mathbf{x}_0 \sim \mathcal{N}(0, \mathbf{I})$ 时是最优的，第二个选择对于数据确定为单个点时是最优的。其次，为了表示均值 $\boldsymbol{\mu}_\theta$ ，作者基于 $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2 \mathbf{I})$ 提出了一种特定的参数化方法

$$\mathcal{L}_{t-1} = \mathbb{E}_q \left[\frac{1}{2\sigma_t^2} \|\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|^2 \right] + C$$

可以通过 $\mathbf{x}_t(\mathbf{x}_0, \epsilon) = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ for $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ 重参数化等式

$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$ 并应用前向后验公式 $\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0)$ 来进一步展开上式得到

$$\begin{aligned}
\mathcal{L}_{t-1} - C &= \mathbb{E}_{\mathbf{x}_0, \epsilon} \left[\frac{1}{2\sigma_t^2} \left\| \tilde{\mu}_t \left(\mathbf{x}_t(\mathbf{x}_0, \epsilon), \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{x}_t(\mathbf{x}_0, \epsilon) - \sqrt{1-\bar{\alpha}_t}\epsilon \right) \right) - \boldsymbol{\mu}_\theta(\mathbf{x}_t(\mathbf{x}_0, \epsilon), t) \right\|^2 \right] \\
&= \mathbb{E}_{\mathbf{x}_0, \epsilon} \left[\frac{1}{2\sigma_t^2} \left\| \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{x}_t(\mathbf{x}_0, \epsilon) - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon \right) - \boldsymbol{\mu}_\theta(\mathbf{x}_t(\mathbf{x}_0, \epsilon), t) \right\|^2 \right]
\end{aligned}$$

μ_θ 必须在给定 \mathbf{x}_t 的情况下预测上述表达式。因为 \mathbf{x}_t 本身就是模型的输入，可以选择下列这种参数化形式，其中 ϵ_θ 是一个旨在从 \mathbf{x}_t 中预测噪声 ϵ 的函数近似器（即神经网络）

$$\mu_\theta(\mathbf{x}_t, t) = \tilde{\mu}_t\left(\mathbf{x}_t, \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_\theta(\mathbf{x}_t, t)\right)\right) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t)\right)$$

利用上式可化简 $\mathcal{L}_{t-1} - C$ 为

$$\mathbb{E}_{\mathbf{x}_0, \epsilon} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1 - \bar{\alpha}_t)} \left\| \epsilon - \epsilon_\theta\left(\sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, t\right) \right\|^2 \right]$$

算法 1 训练

重复

$\mathbf{x}_0 \sim q(\mathbf{x}_0)(p_{\text{data}}(\mathbf{x}))$
 $t \sim \text{Uniform}(\{1, \dots, T\})$
 $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

梯度下降 $\nabla_\theta \left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, t) \right\|^2$

直到收敛

算法 2 采样

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

for $t = T, \dots, 1$ do

$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$

$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$

end for

返回 \mathbf{x}_0

优势

- 1、训练极度稳定：DDPM 将目标函数完美简化为了预测噪声的均方误差 (MSE)，彻底摆脱了 GAN 中极难调参的极小极大博弈和模式崩溃问题。
- 2、生成质量极高：在图像合成上达到了前所未有的保真度。

劣势

- 1、采样极其缓慢：在生成阶段，它需要通过反向马尔可夫链一步一步进行去噪。生成一张图片往往需要进行成百上千次的神经网络前向传播，推理速度远远慢于只需一次前向传播的 GAN 或 VAE。
- 2、隐空间缺乏压缩与直观语义：DDPM 的隐变量维度与原始数据完全一致，它缺乏像 VAE 那样紧凑、结构化且易于直接进行算术语义插值的隐空间表示。
- 3、对数似然指标（无损压缩率）非最优：虽然它的生成图像质量极佳，但 DDPM 在计算负对数似然时，依然无法与当时顶级的自回归模型 (Autoregressive Models) 或流模型 (Flows) 相抗衡。

流模型 (Normalizing Flow)

动机

变分推断要求用一类已知的概率分布来近似难以处理的后验分布，所使用的近似类别通常有限，例如平均场近似，这意味着任何解都无法完全逼近真实后验分布。理想的变分分布族 $q_\phi(\mathbf{z} \mid \mathbf{x})$ 应当具备高度灵活性，最好能灵活到包含真实后验分布作为其解之一。

原理

相关概念

归一化流描述了概率密度通过一系列可逆映射的变换过程。通过反复应用变量变换法则，初始密度“流经”这一系列可逆映射。在此序列的末端，能够得到一个有效的概率分布，因此这类流被称为归一化流。

以深度隐变量高斯模型（deep latent Gaussian models DLGM）为例，其由 L 层高斯隐变量 \mathbf{z}_l 构成层次结构，每层隐变量以非线性方式（由深度神经网络定义）依赖于上一层。联合概率为：

$$p(\mathbf{x}, \mathbf{z}_1, \dots, \mathbf{z}_L) = p(\mathbf{x} \mid f_0(\mathbf{z}_1)) \prod_{l=1}^{L-1} p(\mathbf{z}_l \mid f_l(\mathbf{z}_{l+1}))$$

隐变量的先验服从单位高斯分布 $p(\mathbf{z}_L) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ，观测似然 $p(\mathbf{x} \mid \mathbf{z})$ 可以是任何基于 \mathbf{z}_1 并通过深度神经网络参数化的适当分布。

此类模型具有高度通用性，将因子分析、主成分分析、非线性因子分析及非线性高斯信念网络等模型作为特例包含其中（见[2014 Stochastic backpropagation and approximate inference in deep generative models]）。

基础知识

考虑一个具有可逆的、平滑的映射 $f: \mathbb{R}^d \rightarrow \mathbb{R}^d$ ，其逆映射为 $f^{-1} = g$ ，即复合函数满足 $g \circ f(\mathbf{z}) = \mathbf{z}$ 。如果使用这个映射来变换一个具有分布 $q(\mathbf{z})$ 的随机变量 \mathbf{z} ，由此产生的随机变量 $\mathbf{z}' = f(\mathbf{z})$ 具有如下分布

$$q(\mathbf{z}') = q(\mathbf{z}) \left| \det \frac{\partial f^{-1}}{\partial \mathbf{z}'} \right| = q(\mathbf{z}) \left| \det \frac{\partial f}{\partial \mathbf{z}} \right|^{-1}$$

其中第一个等号是体积的拉伸与概率的守恒（微积分换元法）：

无论怎么对空间进行扭曲映射（从 \mathbf{z} 映射到 \mathbf{z}' ），空间里包含的总概率必须保持为 1。这意味着，在原空间中极小体积微元 $d\mathbf{z}$ 内的概率质量，必须等于映射后新空间极小体积微元 $d\mathbf{z}'$ 内的概率质量。数学表达为：

$$q(\mathbf{z})d\mathbf{z} = q(\mathbf{z}')d\mathbf{z}'$$

在多维空间中，用函数 $\mathbf{z} = f^{-1}(\mathbf{z}')$ 进行坐标变换时，新旧体积微元之间的关系并不是简单的线性缩放，而是由雅可比矩阵的行列式的绝对值来决定的。它衡量了空间在变换中的体积膨胀或收缩率。

$$d\mathbf{z} = \left| \det \frac{\partial f^{-1}}{\partial \mathbf{z}'} \right| d\mathbf{z}'$$

联立上述两式并约去微元 $d\mathbf{z}'$ 得到第一个等号。

第二个等号是链式法则（逆函数定理）：

考虑恒等映射 $f^{-1}(f(\mathbf{z})) = \mathbf{z}$ ，同时对两边关于 \mathbf{z} 求导，得到

$$\frac{\partial f^{-1}}{\partial \mathbf{z}'} \cdot \frac{\partial f}{\partial \mathbf{z}} = \mathbf{I}$$

由线性代数两个矩阵乘积的行列式，等于它们各自行列式的乘积： $\det(\mathbf{A} \cdot \mathbf{B}) = \det(\mathbf{A}) \cdot \det(\mathbf{B})$ 。对上

式两边同时取行列式：

$$\det \left(\frac{\partial f^{-1}}{\partial \mathbf{z}'} \right) \cdot \det \left(\frac{\partial f}{\partial \mathbf{z}} \right) = \det(\mathbf{I}) = 1$$

将等式移项得到第二个等号

$$\det \left(\frac{\partial f^{-1}}{\partial \mathbf{z}'} \right) = \frac{1}{\det \left(\frac{\partial f}{\partial \mathbf{z}} \right)} = \left(\det \frac{\partial f}{\partial \mathbf{z}} \right)^{-1}$$

可以通过组合几个简单的映射并连续应用上述公式，来构造任意复杂的概率密度。通过一条包含 K 个变换 f_k 的链，对具有分布 q_0 的随机变量 \mathbf{z}_0 进行连续变换，所得到的密度 $q_K(\mathbf{z})$ 及其生成过程为：

$\mathbf{z}_K = f_K \circ \dots \circ f_2 \circ f_1(\mathbf{z}_0)$ 。对应的对数密度为：

$$\ln q_K(\mathbf{z}_K) = \ln q_0(\mathbf{z}_0) - \sum_{k=1}^K \ln \left(\det \left| \frac{\partial f_k}{\partial \mathbf{z}_k} \right| \right)$$

此类变换的一个特性（通常被称为无意识统计学家法则，law of the unconscious statistician LOTUS）是，关于变换后密度 q_K 的数学期望，可以在不需要显式知道 q_K 表达式的情况下计算出来。任何期望 $\mathbb{E}_{q_K}[h(\mathbf{z})]$ 都可以被写成在 q_0 下的期望形式：

$$\mathbb{E}_{q_K}[h(\mathbf{z})] = \mathbb{E}_{q_0}[h(f_K \circ f_{K-1} \circ \dots \circ f_1(\mathbf{z}_0))]$$

当函数 $h(\mathbf{z})$ 不依赖于 q_K 本身时，这种计算不要求对数雅可比行列式项。通过恰当选择变换 f_K ，可以在最初使用简单的因子化分布（例如独立的高斯分布），并应用不同长度的归一化流，来获得日益复杂和多峰的分布。

可逆线性时间变换 (Invertible Linear-time Transformations)

平面流 (PLANAR FLOWS)

考虑一种形式如下的变换族：

$$f(\mathbf{z}) = \mathbf{z} + \mathbf{u}h(\mathbf{w}^\top \mathbf{z} + b)$$

其中 $\lambda = \{\mathbf{w} \in \mathbb{R}^D, \mathbf{u} \in \mathbb{R}^D, b \in \mathbb{R}\}$ 是自由参数，对于这种映射可以在 $O(D)$ 的时间内计算对数雅可比行列式项（由矩阵行列式引理 $\det(\mathbf{I} + \mathbf{u}\mathbf{v}^\top) = 1 + \mathbf{v}^\top \mathbf{u}$ ）：

$$\psi(\mathbf{z}) = h'(\mathbf{w}^\top \mathbf{z} + b)\mathbf{w}$$

$$\det \left| \frac{\partial f}{\partial \mathbf{z}} \right| = |\det(\mathbf{I} + \mathbf{u}\psi(\mathbf{z})^\top)| = |1 + \mathbf{u}^\top \psi(\mathbf{z})|$$

由此可得经此变换的对数概率密度

$$\ln q_K(\mathbf{z}_K) = \ln q_0(\mathbf{z}) - \sum_{k=1}^K \ln |1 + \mathbf{u}_k^\top \psi_k(\mathbf{z}_k)|$$

径向流 (RADIAL FLOWS)

作为另一种选择，可以考虑一族围绕参考点 \mathbf{z}_0 修改初始密度 q_0 的变换。该变换族为：

$$f(\mathbf{z}) = \mathbf{z} + \beta h(\alpha, r)(\mathbf{z} - \mathbf{z}_0)$$

其中 $r = |\mathbf{z} - \mathbf{z}_0|$ 是点到中心的距离, $h(\alpha, r) = 1/(\alpha + r)$, 映射的参数为 $\lambda = \{\mathbf{z}_0 \in \mathbb{R}^D, \alpha \in \mathbb{R}, \beta \in \mathbb{R}\}$ 。

注意并非所有形如上两式的函数都是可逆的。需要的可逆性的条件, 以及以数值稳定的方式满足这些条件可见 [2015 Variational Inference with Normalizing Flows]。

基于流的自由能下界

回到边际似然的下界

$$\begin{aligned} & \log p_{\theta}(\mathbf{x}) \\ &= \log \int p_{\theta}(\mathbf{x} | \mathbf{z}) p(\mathbf{z}) \, d\mathbf{z} \\ &= \log \int \frac{q_{\phi}(\mathbf{z} | \mathbf{x})}{q_{\phi}(\mathbf{z} | \mathbf{x})} p_{\theta}(\mathbf{x} | \mathbf{z}) p(\mathbf{z}) \, d\mathbf{z} \\ &\geq -D_{\text{KL}}(q_{\phi}(\mathbf{z} | \mathbf{x}) || p(\mathbf{z})) + \mathbb{E}_q[\log p_{\theta}(\mathbf{x} | \mathbf{z})] = -\mathcal{F}(\mathbf{x}). \end{aligned}$$

如果用长度为 K 的流来参数化近似后验分布, 即令 $q_{\phi}(\mathbf{z} | \mathbf{x}) := q_K(\mathbf{z}_K)$, 那么自由能 (即变分下界的负值, 见上式) 可以写成关于初始分布 $q_0(\mathbf{z})$ 的数学期望 (采用平面流函数形式) :

$$\begin{aligned} \mathcal{F}(\mathbf{x}) &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})}[\log q_{\phi}(\mathbf{z} | \mathbf{x}) - \log p(\mathbf{x}, \mathbf{z})] \\ &= \mathbb{E}_{q_0(\mathbf{z}_0)}[\ln q_K(\mathbf{z}_K) - \log p(\mathbf{x}, \mathbf{z}_K)] \\ &= \mathbb{E}_{q_0(\mathbf{z}_0)}[\ln q_0(\mathbf{z}_0)] - \mathbb{E}_{q_0(\mathbf{z}_0)}[\log p(\mathbf{x}, \mathbf{z}_K)] - \mathbb{E}_{q_0(\mathbf{z}_0)}\left[\sum_{k=1}^K \ln |1 + \mathbf{u}_k^{\top} \psi_k(\mathbf{z}_k)|\right] \\ &= \mathbb{E}_{q_0(\mathbf{z}_0)}[\ln q_0(\mathbf{z}_0)] - \mathbb{E}_{q_0}\left[\log p(\mathbf{x} | f_0(\mathbf{z}_1)) + \sum_{l=1}^{L-1} \log p(\mathbf{z}_l | f_l(\mathbf{z}_{l+1})) + \log p(\mathbf{z}_L)\right] - \mathbb{E}_{q_0(\mathbf{z}_0)}\left[\sum_{k=1}^K \ln |1 + \mathbf{u}_k^{\top} \psi_k(\mathbf{z}_k)|\right] \end{aligned}$$

- $\mathbb{E}[\ln q_0(\mathbf{z}_0)]$: 简单的高斯噪声的熵, $q_0(\mathbf{z}_0 | \mathbf{x}) = \mathcal{N}(\mathbf{z}_0 | \mu(\mathbf{x}), \text{diag}(\sigma^2(\mathbf{x})))$ 。
- $-\mathbb{E}[\log p(\mathbf{x}, \mathbf{z}_K)]$: 在模型生成假样本时的重构误差, $\mathbf{z}_K \equiv \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_L\}$ 。
- $-\mathbb{E}[\sum \ln |\dots|]$: 这是流在扭曲空间时产生的体积变化罚项。流扭曲得越剧烈, 这个罚项的反馈就越清晰。

优势

- 1、精确的对数似然评估: 与 VAE 只能近似优化似然下界 (ELBO) 或 GAN 这种隐式模型根本无法计算似然不同, Flow 模型能够直接、精确地计算并优化数据真实的概率密度。
- 2、完美的双向可逆性: 编码 (推断) 和解码 (生成) 过程是绝对对称且无损的。模型可以将复杂数据完美映射为潜空间噪声, 也能将该噪声一丝不差地还原为原始数据。
- 3、极强的分布拟合能力: 理论上, 只要串联的“流”操作足够多, 它可以将任何极其简单的初始分布 (如标准高斯噪声) 扭曲、塑造成极其复杂且多峰的真实数据分布。

劣势

- 1、严苛的网络架构限制: 为了保证前向传播“绝对可逆”且反向传播时“雅可比行列式极易计算 (线性时间复杂度)”, 不能随意使用普通的神经网络层, 必须认真地设计特殊的网络层 (如平面流、径向流), 极大地限制了模型的表达上限。
- 2、无法进行降维压缩: 因为数学上的双射要求, 潜变量空间的维度必须严格等于原始数据的维度。这意味着对于高分辨率图片, 其潜空间同样极其庞大, 导致计算成本和显存占用极高, 且无法像 VAE 那

样提取低维、高浓缩的核心语义特征。

3、生成质量的妥协：受限于上述网络结构限制，在实际应用中，Flow 模型在极高保真度的图像生成任务上，其细节清晰度往往逊色于 GAN，也比不上后来的扩散模型 DDPM。