

日期： /

Score based model

生成式模型是如何建模 $p(x)$ 的？

Autoregressive models: $P_\theta(x) = \prod_{t=1}^d P_\theta(x_t | X_{\leq t})$

Flow models: $P_\theta(x) = p(z) / \det(J_{f_\theta}(x))$, $z = f_\theta(x)$

VAEs: $P_\theta(x) \approx \int p(z) p_\theta(x|z) dz$

EBMs: $P_\theta(x) = e^{f_\theta(x)} / Z(\theta)$

GANs: 统过 $P_\theta(x)$ 直接建模采样 $x = g_\theta(z)$, $z \sim \pi(z)$

向量场

SBMs 以 分数 描述概率分布，只适用于连续变量
分数函数的定义是：

$$S(x) = \nabla_x \log p(x)$$

使用在 EBMs 中提及的方式定义训练目标：

$$\max \log P_\theta(x) = \max_\theta \log \frac{e^{f_\theta(x)}}{Z(\theta)} = \max f_\theta(x) - \log Z(\theta)$$

$$\nabla_\theta f_\theta(x_{\text{train}}) - \nabla_\theta \log Z(\theta) \approx \nabla_\theta f_\theta(x_{\text{train}}) - \nabla_\theta f_\theta(x_{\text{sample}})$$

日期： /

使用 Fisher 散度作为训练目标以规避采样：

$$\begin{aligned} & \min_{\theta} \frac{1}{2} \mathbb{E}_{x \sim P_{\text{data}}} [\|\nabla_x \log P_{\text{data}}(x) - \nabla_x \log P_{\theta}(x)\|_2^2] \\ &= \min_{\theta} \mathbb{E}_{x \sim P_{\text{data}}} [\frac{1}{2} \|\nabla_x \log P_{\theta}(x)\|_2^2 + \text{tr}(\nabla_x^2 \log P_{\theta}(x))] + \text{const.} \end{aligned}$$

所以与 EBM 不同的是，SBM 直接通过建模向量场描述概率分布。

对于 SBMs 有以下总结：

- Given: i.i.d. samples $\{x_1, x_2 \dots x_n\} \sim P_{\text{data}}(x)$
- Task: 建模 $\nabla_x \log P_{\text{data}}(x)$
- Score Model: $S_{\theta}(x): \mathbb{R}^d \rightarrow \mathbb{R}^d$
- Goal: $S_{\theta}(x) \approx \nabla_x \log P_{\text{data}}(x)$

Q1: 如何比较 $S_{\theta}(x)$ 与 $\nabla_x \log P_{\text{data}}(x)$?

A1: 通过 Fisher 散度

$$D_F = \frac{1}{2} \mathbb{E}_{x \sim P_{\text{data}}} [\|\nabla_x \log P_{\text{data}}(x) - S_{\theta}(x)\|_2^2]$$

$$\text{及近似 } \mathbb{E}_{x \sim P_{\text{data}}} [\frac{1}{2} \|S_{\theta}(x)\|_2^2 + \text{tr}(\underline{\nabla_x S_{\theta}(x)})]$$

如何保证这一部分
可以高效计算？

日期： /

对于一般的神经网络， $\text{tr}(\nabla_x S_0(x))$ 的计算在高维下是很昂贵的。而在EBMs中甚至更昂贵！

[Denoising Score Matching]

DSM 涉及几个成分：

向原始图像中添加高斯噪声：

$$q_\sigma(\tilde{x}|x) = \mathcal{N}(x; \underbrace{\sigma I^3}_{\sigma \text{通常取一个较小的值}})$$

$$q_\sigma(\tilde{x}) = \int p(x) q_\sigma(\tilde{x}|x) dx$$

把以下的 Fisher 散度作为原始目标的近似

$$\begin{aligned} (\text{min}) \quad & \frac{1}{2} \mathbb{E}_{\tilde{x} \sim q_\sigma} [\|\nabla_{\tilde{x}} \log q_\sigma(\tilde{x}) - S_0(\tilde{x})\|_2^2] \\ &= \frac{1}{2} \underbrace{\int q_\sigma(\tilde{x}) \|S_0(\tilde{x})\|_2^2 d\tilde{x}}_{\text{易算}} - \underbrace{\int q_\sigma(\tilde{x}) \nabla_{\tilde{x}} \log q_\sigma(\tilde{x})^\top S_0(\tilde{x}) d\tilde{x} + \text{const.}}_{\text{难算}} \\ &= - \int q_\sigma(\tilde{x}) \frac{1}{q_\sigma(\tilde{x})} \nabla_{\tilde{x}} q_\sigma(\tilde{x})^\top S_0(\tilde{x}) d\tilde{x} \\ &= - \int \nabla_{\tilde{x}} \left(\int P_{\text{data}}(x) q_\sigma(\tilde{x}|x) dx \right)^\top S_0(\tilde{x}) d\tilde{x} \\ &= - \int \left(\int P_{\text{data}}(x) \nabla_{\tilde{x}} q_\sigma(\tilde{x}|x) dx \right)^\top S_0(\tilde{x}) d\tilde{x} \\ &= - \int \left(\int P_{\text{data}}(x) q_\sigma(\tilde{x}|x) \nabla_{\tilde{x}} \log q_\sigma(\tilde{x}|x) dx \right)^\top S_0(\tilde{x}) d\tilde{x} \\ (\star) \quad &= - \iint P_{\text{data}}(x) q_\sigma(\tilde{x}|x) \nabla_{\tilde{x}} \log q_\sigma(\tilde{x}|x)^\top S_0(\tilde{x}) dx d\tilde{x} \\ &= - \mathbb{E}_{x \sim P_{\text{data}}(x), \tilde{x} \sim q_\sigma(\tilde{x}|x)} [\nabla_{\tilde{x}} \log q_\sigma(\tilde{x}|x)^\top S_0(\tilde{x})] \end{aligned}$$

日期： /

$$\text{因此 } \frac{1}{2} \mathbb{E}_{\tilde{x} \sim q_\phi} [\|\nabla_{\tilde{x}} \log q_\phi(\tilde{x}) - S_\phi(\tilde{x})\|_2^2]$$

$$= \frac{1}{2} \mathbb{E}_{\tilde{x} \sim q_\phi} [\|S_\phi(\tilde{x})\|_2^2] - \mathbb{E}_{x \sim P_{\text{data}}(x), \tilde{x} \sim q_\phi(\tilde{x}|x)} [\nabla_{\tilde{x}} \log q_\phi(\tilde{x}|x)^T S_\phi(x)] \\ + \text{const.}$$

$$\begin{aligned} (\star) &= \frac{1}{2} \mathbb{E}_{x \sim P_{\text{data}}(x), \tilde{x} \sim q_\phi(\tilde{x}|x)} [\|S_\phi(\tilde{x}) - \nabla_{\tilde{x}} \log q_\phi(\tilde{x}|x)\|_2^2] \\ &\quad - \frac{1}{2} \mathbb{E}_{x \sim P_{\text{data}}(x), \tilde{x} \sim q_\phi(\tilde{x}|x)} [\|\nabla_{\tilde{x}} \log q_\phi(\tilde{x}|x)\|_2^2] + \text{const.} \\ &= \frac{1}{2} \mathbb{E}_{x \sim P_{\text{data}}(x), \tilde{x} \sim q_\phi(\tilde{x}|x)} [\|S_\phi(\tilde{x}) - \underline{\nabla_{\tilde{x}} \log q_\phi(\tilde{x}|x)}\|_2^2] + \text{const.} \end{aligned}$$

单步

$$q_\phi(\tilde{x}|x) \sim N(\tilde{x}; x, \sigma^2 I) \quad \nabla_{\tilde{x}} \log q_\phi(\tilde{x}|x) = -\frac{\tilde{x}-x}{\sigma^2}$$

这种处理使得我们无需计算 Jacobian 矩阵的逆

使用 DSM 的训练过程大致如下：

- 获得数据点 $\{x_1, x_2, \dots, x_n\} \sim P_{\text{data}}(x)$
- 获得添加噪声的数据点 $\{\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n\} \sim q_\phi(\tilde{x})$
- 评估损失： $\frac{1}{2n} \sum_{i=1}^n [\|S_\phi(\tilde{x}_i) - \underline{\nabla_{\tilde{x}} \log q_\phi(\tilde{x}_i|x_i)}\|_2^2]$

$$\text{在 Gaussian 噪声下: } \frac{1}{2n} \sum_{i=1}^n [\|S_\phi(\tilde{x}_i) + \frac{\tilde{x}_i - x_i}{\sigma^2}\|_2^2]$$

只要噪声分布可以
计算的话，梯度可以封闭求出。
降噪损失就可取得

日期:

$$\begin{aligned} & \frac{1}{2} E_{q_\theta(\tilde{x})} [\|\nabla_{\tilde{x}} \log q_\theta(\tilde{x}) - s_\theta(\tilde{x})\|_2^2] \\ &= \frac{1}{2} E_{p(x)} E_{q_\theta(\tilde{x}|x)} [\|\frac{1}{\sigma^2}(x-\tilde{x}) - s_\theta(\tilde{x})\|_2^2] + \text{const.} \end{aligned}$$

[Tweedie's formula]

最佳的去噪方式是沿噪声的对数似然方向。

$$\begin{aligned} E_{x \sim p(x|\tilde{x})}[X] &= \tilde{x} + \sigma^2 \nabla_{\tilde{x}} \log q_\theta(\tilde{x}) \\ &\approx \tilde{x} + \sigma^2 s_\theta(\tilde{x}) \end{aligned}$$

[Sliced score match]

这是有别于 MSM 的另一种简化方式。其基本思想是当 $s_\theta(x)$ 与真实样本空间的向量场一致时，其沿某个方向的投影也应当一致。

SSM 可以直接匹配真实概率分布。

但仍需求导，所以进
度慢于 DSM

SSM 的优化目标为：

$$\frac{1}{2} \operatorname{E}_{v \sim P_v} \operatorname{E}_{x \sim P_{\text{data}}} [(\underbrace{V^\top \nabla_x \log p_{\text{data}}(x)}_{\text{自先验分布采样的投影向量}} - V^\top s_\theta(x))^2]$$

通过类似的简化流程：

$$\begin{aligned} & \operatorname{E}_{v \sim P_v} \operatorname{E}_{x \sim P_{\text{data}}} [V^\top \nabla_x s_\theta(x) V + \frac{1}{2} (V^\top s_\theta(x))^2] \\ &= \operatorname{E}_{v \sim P_v} \operatorname{E}_{x \sim P_{\text{data}}} [V^\top \nabla_x (\underbrace{V^\top s_\theta(x)}_{\text{这是一个梯度而非 Jacobian 矩阵}}) + \frac{1}{2} (V^\top s_\theta(x))^2] \end{aligned}$$

日期： /

Q₂：如何采样？

A₂：单纯使用Langevin MCMC不会有很好的效果。

这是由于：对于图像，这意味着像素点
人间并不独立

① 真实数据倾向于位于低维流形上，意味着分数不总是具有意义

② Langevin MCMC在低密度区相当低效，分数估计在这些区域并不准确

③ 分数函数难以对混合概率分布的权重进行建模

对于DSM这意味着口不应过大

通过向原始数据中加入Gauss噪声可以明显改善采样过程。但问题在于建模也会基于带有噪声的数据。

低噪声：建模准确，采样困难

中噪声：建模有偏，采样可行

高噪声：过度平滑，采样简单

[Annealed Langevin Dynamics: Joint Scores for Sampling]

- 这是一种逐渐去噪的过程。

- 这种方法要求对含有不同水平噪声的数据建模，并依次使用这些模型进行去噪采样。 把噪声水平作为模型输入即可实现 (NOSN)

- 使用DSM进行训练是自然的。

- 使用加权的 DSM loss，权重为 $\lambda(\sigma_i)$

$$\text{loss} = \frac{1}{L} \sum_{i=1}^L \lambda(\sigma_i) E_{q_{\sigma_i}(x)} [\|\nabla_x \log q_{\sigma_i}(x) - s_i(x; \sigma_i)\|_2^2]$$

$$= \frac{1}{L} \sum_{i=1}^L \lambda(\sigma_i) E_{x \sim p_{\text{data}}, z \sim N(0, I)} [\|\nabla_{\tilde{x}} \log q_{\sigma_i}(\tilde{x}; \sigma_i) - s_i(\tilde{x}, \sigma_i)\|_2^2] + \text{const.}$$

$$= \frac{1}{L} \sum_{i=1}^L \lambda(\sigma_i) E_{x \sim p_{\text{data}}, z \sim N(0, I)} [\|s_i(x + \sigma_i z; \sigma_i) + \frac{\partial}{\partial x} s_i\|_2^2] + \text{const.}$$

- 最大噪声尺度: $\sigma_1 = \text{maximum pairwise distance between datapoints}$

最小噪声尺度: $\sigma_L = \text{sufficiently small}$

如何插值: 使不同尺度噪声下的分布有明显重叠

权重选择: $\lambda(\sigma_i) = \sigma_i^{-2}$ 平衡不同噪声级别下的损失

日期： /

现在的问题是，能否把添加/去除噪声的过程表示为连续过程？即选择无限多的噪声水平。

(SDE)

我们可以用随机微分方程描述原始分布至噪声分布的连续过程：

$$dx_t = \underline{f(x_t, t) dt + g(t) d\omega_t} \quad (-\text{般结构})$$

Deterministic drift Infinitesimal noise

对于图像添加噪声的扩散过程：

$$dx_t = \underline{\sigma(t) d\omega_t}$$

而去噪过程可以描述为：

$$dx_t = -\underline{\sigma(t)^2 \nabla_x \log p(x_t)} dt + \underline{\sigma(t) d\bar{w}_t}$$

Score function
麦克斯拉根
MCMC
噪声项

所以想法是建立一个模型：

$$s_\theta(x, t) \approx \nabla_x \log p_\theta(x)$$

训练过程为：

$$\mathbb{E}_{t \sim U[0, T]} \left[\lambda(t) \mathbb{E}_{p_\theta(x)} \left[\| \nabla_x \log p_\theta(x) - s_\theta(x, t) \|_2^2 \right] \right]$$

日期: /

$$\text{反向 SDE: } dx = -\sigma^2(t) s_0(x, t) dt + \sigma(t) d\bar{w}$$

这个 SDE 有许多解法, 例如:

[Euler - Maruyama method]

$$x \leftarrow x - \sigma(t)^2 s_0(x, t) \Delta t + \sigma(t) z \quad (z \sim N(0, \Delta t))$$

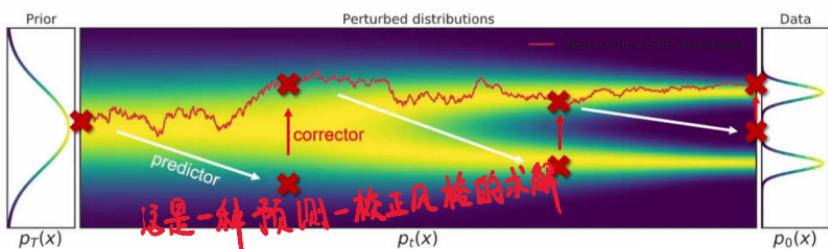
(一种高数化时间的方式)

$$t \leftarrow t + \Delta t$$

采样:

Predictor-Corrector sampling methods

- Predictor-Corrector sampling.
 - Predictor: Numerical SDE solver
 - Corrector: Score-based MCMC



以一个简单的程序描述扩散过程

$$q(x_t | x_{t-1}) = N(x_t; \sqrt{1-p_t} x_{t-1}, p_t I)$$

这定义了一个联合概率分布:

$$q(\underline{x}_1: \underline{x}_T | x_0) = \prod_{t=1}^T q(x_t | x_{t-1})$$

扩散过程就类似于编码过程, 但编码方式只是加入噪声
编码过程不同于 VAE, 是非参数化的

日期： /

或者： $q(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t)I)$ $\bar{\alpha}_t = \prod_{s=1}^t (1 - \beta_s)$

这是另一个视角的编码过程

这意味着可以直接对 t 时同步的扩散采样，而依赖此前的扩散结果

那么，只要我们可以取得 $q(x_{t-1} | x_t)$ ，就可以反转扩散过程，从采样自先验分布的噪音中恢复数据。

Sample x_1 from $p(x_1) = \mathcal{N}(x_1; 0, I) = \pi$

Denoising $p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma^2 I)$

这类似于解码过程

类似的联合概率分布： $p_\theta(x_{0:T}) = p(x_T) \prod p_\theta(x_{t-1} | x_t)$

有一些 MC 近似的味道

[Hierarchical VAE]

Normal VAE: $\textcircled{x} \rightarrow \textcircled{z} \rightarrow \hat{x}$

Hierarchical VAE: $\textcircled{x} \rightarrow \textcircled{z}_1 \rightarrow \textcircled{z}_2 \rightarrow \hat{x}$

H-VAE 的采样过程：①自 $N(0, I)$ 中采样 z_1

②自 $p(z_2 | z_1)$ 中采样 z_2

③自 $p(x | z_2)$ 中采样 x

日期： /

在 Normal VAE 中， $ELBO = E_{q_\phi(z|x)} [\log \frac{p(z, x; \theta)}{q_\phi(z|x)}]$

在 H-VAE 中， $ELBO = E_{q(z_1, z_2|x)} [\log \frac{p(x, z_1, z_2)}{q(z_1, z_2|x)}]$

那么对于扩散过程，可以类似地写出

$$ELBO = E_{q(x_0)q(x_{1:T}|x_0)} \log \frac{p(x_{0:T})}{q(x_{1:T}|x_0)}$$

$$L = -ELBO$$

参数化的去噪过程可以表示为：

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2 I)$$

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{1-\bar{\alpha}_t}} (x_t - \frac{\bar{\alpha}_t}{\sqrt{1-\bar{\alpha}_t}} \underline{E_\theta(x_t, t)})$$

如此做，则 ELBO 可以转化为一般的去噪分数匹配损失：

$$L = E_{x_0 \sim q(x_0), t \sim U[1, T], \epsilon \sim \mathcal{N}(0, 1)} [\lambda_t \underbrace{\|\epsilon - E_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t} \epsilon, t)\|_2^2}_{\text{缩放因子}} \underbrace{\lambda_t \|\epsilon\|_2^2}_{\text{这部分可以看作分数}}]$$

这两种视角决定了采样是通过解码进行还是
通过 Langevin MCMC 并行。

Algorithm 1 Training

```
1: repeat
2:    $x_0 \sim q(x_0)$ 
3:    $t \sim \text{Uniform}\{1, \dots, T\}$ 
4:    $\epsilon \sim \mathcal{N}(0, I)$ 
5:   Take gradient descent step on
       $\nabla_\theta \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t} \epsilon, t)\|^2$ 
6: until converged
```

Algorithm 2 Sampling

```
1:  $x_T \sim \mathcal{N}(0, I)$ 
2: for  $t = T, \dots, 1$  do
3:    $z \sim \mathcal{N}(0, I)$ 
4:    $x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z$ 
5: end for
6: return  $x_0$ 
```

加入噪声

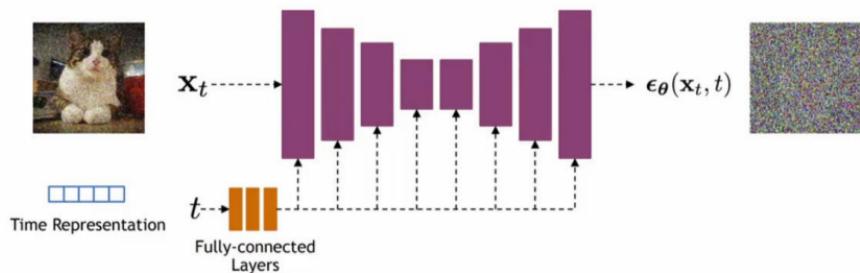
DDPM 中的视角接近 DSM

日期:

实践中使用U-net风格的解码器

Architecture for the denoiser

Unet architecture used in practice



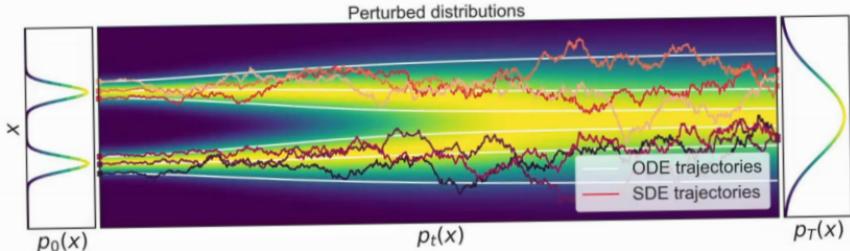
对于连续(无穷多时间步)的情况,已介绍使用预测-校正风格求解SDE,而DDPM使用一种不同的策略,即将SDE转换为ODE

SDE

$$dx_t = \sigma(t) dw_t \Leftrightarrow \frac{dx_t}{dt} = -\frac{1}{2} \sigma(t)^2 \underbrace{\nabla_x \log p_t(x_t)}_{\text{score function}} \approx s_\theta(x, t)$$

ODE

这提供一个确定性的可逆映射!!! (FBM)



日期： /

在此视角下，评估仍然是可行的：

$$\log p_0(x_0) = \log \pi(x_T) - \frac{1}{2} \int_0^T \sigma(t)^2 \text{trace}(\nabla_x s_\theta(x, t)) dt$$

多项式复杂度

然而 SDE 通常能取得更好的采样效果。

对高步数模型进行蒸馏可以缩减推理时间。

[Latent diffusion model]

在传统的扩散步骤前先进行一个隐空间编码，降低数据维度，从而加速推理。

此外，这将允许 Diffusion 模型用于更多的数据结构，如文本。

通常使用预训练 VAE，且无需关注隐空间分布是否接近 Gauss 分布。

日期： /

[Conditional generation]

令 (x, y) 表示 $(\text{image}, \text{caption})$ 对。

一个条件生成模型拟合 $p(x|y)$

可以使用以下的分数匹配：

$$E_{(x,y) \sim p_{\text{data}}(x,y)} E_{e \sim N(0,1)} E_{t \sim U[0,1]} \| E_{\theta}(x_t, t; y) - e \|_2^2$$

如果已有生成模型 $P_\theta(x)$ ，有分类器 $P_\theta(y|x)$

由 Bayes' rule : $p(x|y) = \frac{p(x)p(y|x)}{p(y)}$ 取对数计算

$$\text{然而 } \nabla_x \log p(x|y) = \nabla_x \log p(x) + \nabla_x \log p(y|x) - \nabla_x \log p(y)$$

$$= \nabla_x \log p(x) + \nabla_x \log p(y|x)$$

由 Langevin MCMC采样

日期: /