Lecture 15 - Sampling and Diffusion Models

赵尉辰

南开大学 统计与数据科学学院

目录

采样问题

采样(Sampling)

设 μ 是一个概率分布，采样问题是指：如何获得随机样本X，使得X的分布为μ。

优化(Optimizing)

设$f(x)$为一个目标函数，优化问题是指：如何在$x \in \mathcal{X}$这个可行范围内，找到目标函数f的最小值。

目录

生成

生成任务

在机器学习中，生成任务 是指模型的目标为生成新的数据实例，这些实例与已有数据(训练数据)具有相似的特征或模式，常见的生成任务包括文本生成、图像生成、视频生成等。

处理生成任务通常包含两个部分：

(1) 学习数据分布

学习通常利用深度神经网络实现，神经网络是一个参数化的模型，需要通过优化更新模型参数。

(2) 根据数据分布生成实例

生成新的数据样本即就是从数据分布中采样，需要通过采样算法实现。

目录

推断

推断是统计学中的基本问题。

推断任务

在机器学习中，所有未知的量，无论是对未来的预测，系统的隐藏状态，还是模型的参数，都被视为随机变量，并被赋予概率分布。推断任务是指根据已知数据，计算这些随机变量的后验分布。

设ϕ是未知变量，$\mathcal{D}$是已知变量，给定先验$p(\phi)$和似然$p(\mathcal{D}|\phi)$，我们可以通过贝叶斯定理计算后验：

$$p(\phi|\mathcal{D}) = \frac{p(\phi)p(\mathcal{D}|\phi)}{p(\mathcal{D})}$$

计算的瓶颈在于归一化系数

$$p(\mathcal{D}) = \int p(\mathcal{D}|\phi)p(\phi)d\phi$$

尤其是在高维情形。

推断

处理后验分布的计算有两种常见的方式:

(1) Monte Carlo 方法

高维积分$\int f(x)p(x)dx$难以计算时，Monte Carlo 方法通过采样服从$p(x)$的样本$\{X_i\}_{i=1,\ldots,n}$，用

$$\frac{1}{n}\sum_{i=1}^{n} f(X_i)$$

近似计算$\int f(x)p(x)dx$.

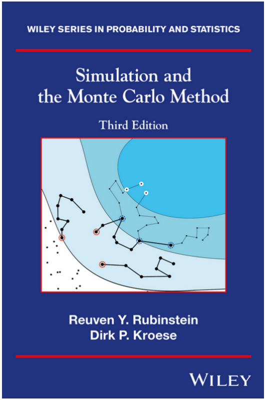
(2) 变分推断

由于$p(\phi|\mathcal{D})$不好直接计算，变分推断的思想是找一个分布$q(\phi)$来近似后验分布，如何近似则转化为一个优化问题:

$$q = argmin_{q\in\mathcal{Q}}\ D_{KL}(q(\phi)\|p(\phi|\mathcal{D}))$$

目录

随机模拟



- 随机数的模拟
- 基本随机变量的模拟
- 随机过程/随机向量的模拟

采样

定义 1 (Boltzmann-Gibbs分布)

*Boltzmann-Gibbs*分布是统计物理中描述系统在热平衡状态下粒子分布的一种概率分布。在一个由 N 个粒子组成的系统中，当系统处于温度 T 时，粒子在能量为 E_i 的状态下的概率 P_i 由以下公式给出：

$$P_i = \frac{e^{-\beta E_i}}{Z}$$

其中：$\beta = \frac{1}{k_B T}$，k_B 是玻尔兹曼常数，$Z \triangleq \sum_j e^{-\beta E_j}$称为配分函数，求和是对所有可能的状态 j 进行的。

问题 1 (Sampling)

给定光滑函数$V : \mathbb{R}^d \to \mathbb{R}$, 如何获得$\mathbb{R}^d$上的随机样本服从概率分布

$$\pi \propto e^{-V}$$

我们一般称V为位势函数*(potential function)*

Markov Chain Monte Carlo (MCMC)

定义 2 (Markov链)

一个*Markov*链是一个随机过程 $\{X_n\}_{n=0}^{\infty}$，满足：未来状态只依赖于当前状态，而与过去状态无关。即，对于任意的 n 和状态 i, j，有：

$$P(X_{n+1} = j \mid X_n = i, X_{n-1} = x_{n-1}, \ldots, X_0 = x_0) = P(X_{n+1} = j \mid X_n = i)$$

进一步，如果转移概率不随时间变化，即：

$$P(X_{n+1} = j \mid X_n = i) = P(X_1 = j \mid X_0 = i) \quad \forall n$$

那么我们称*Markov*链是时齐的。

Markov Chain Monte Carlo (MCMC)

定义 3 (平稳分布)

设 $\{X_n\}_{n=0}^{\infty}$ 是一个*Markov*链，其状态空间为 S。如果存在一个概率分布 π 满足以下条件：

$$\pi(j) = \sum_{i\in S} \pi(i)P(i,j) \quad \forall j \in S$$

其中 $P(i,j)$ 是从状态 i 转移到状态 j 的概率，则称 π 为该*Markov*链的平稳分布*(Stationary Distribution)*/不变测度*(Invariant Measure)*。

MCMC的思想即是：构造一个Markov链，使得它的平稳分布是我们采样的目标分布π。那么从任意状态分布(容易获得样本的分布)出发，经过充分的状态转移，就能获得目标分布π的样本。

Metropolis-Hastings 算法

1. 选择初始状态 x_0。
2. 对于每一步 $n = 1, 2, \ldots, N$:
   - 从提议分布 $q(x'|x_{n-1})$ 中生成候选状态 x'。
   - 计算接受概率：
   $$\alpha = \min\left(1, \frac{\pi(x')q(x_{n-1}|x')}{\pi(x_{n-1})q(x'|x_{n-1})}\right)$$
   - 以概率 α 接受候选状态 x'，否则保持状态 x_{n-1}。

设 X_{n-1} 是当前状态，X_n 是下一个状态，Metropolis-Hastings 算法转移概率可以表示为：

$$P(X_n = x'|X_{n-1} = x) = q(x'|x) \cdot \min\left(1, \frac{\pi(x')q(x|x')}{\pi(x)q(x'|x)}\right)$$

容易check Metropolis-Hastings 算法的平稳分布是目标分布π

$$\pi(x) = \sum_y \pi(y)q(x|y) \cdot \min\left(1, \frac{\pi(x)q(y|x)}{\pi(y)q(x|y)}\right)$$

Langevin Dynamics

定义 4 (Langevin扩散(Langevin Diffusion))

设$V:\mathbb{R}^d\to\mathbb{R}$是一个位势，我们称随机微分方程*(Stochastic Differential Equations, SDEs)*

$$\mathrm{d}X_t=-\nabla V(X_t)\mathrm{d}t+\sqrt{2}\mathrm{d}B_t.$$

的解为V对应的*Langevin扩散/Langevin Dynamics*, 其中B_t为一个标准布朗运动。

命题 1

*Langevin*扩散的不变测度为

$$\pi\propto e^{-V}$$

Langevin算法

定义 5 (Langevin算法)

对*Langevin*扩散*Euler–Maruyama*离散化：

$$X_{(k+1)h} := X_{kh} - h\nabla V(X_{kh}) + \sqrt{2}(B_{(k+1)h} - B_{kh}).$$

我们得到了一种*Langevin*扩散的实现方式，称为*(Unadjusted) Langevin Algorithm, ULA*/*Langevin Monte Carlo, LMC*. 其中h是迭代步长，k是迭代轮数。

目前采样的研究方向

- 采样算法的收敛性/收敛速度分析

依赖于目标分布的性质：Log-concavity, 紧支撑；
Langevin Dynamics 的亚稳态；
耦合方法(Coupling)，泛函不等式(Log-Sobolev不等式、庞加莱不等式)

- 采样算法复杂性分析

- 已有算法的改进

准确性：Metropolis-adjusted Langevin algorithm (MALA),
速度：Hamiltonian Monte Carlo (HMC)

- 新采样算法的设计

Reverse diffusion, Stochastic Localization.

- 离散采样

目录

生成——从数据分布中采样

数据分布是未知的，我们仅有一些样本，扩散模型的基本思想是：

(1) 学习数据分布；(2) 根据学习到的数据分布生成实例

Score Matching Langevin Dynamics, SMLD[1]

Recall that Langevin dynamics

$$\mathrm{d}X_t = -\nabla V(X_t)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t$$

具有不变测度$\pi \propto e^{-V}$，若我们需要采样p_{data}，可以通过

$$\mathrm{d}X_t = \nabla \log p_{data}(X_t)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t.$$

其中$\nabla \log p$称为概率分布p的Score function.

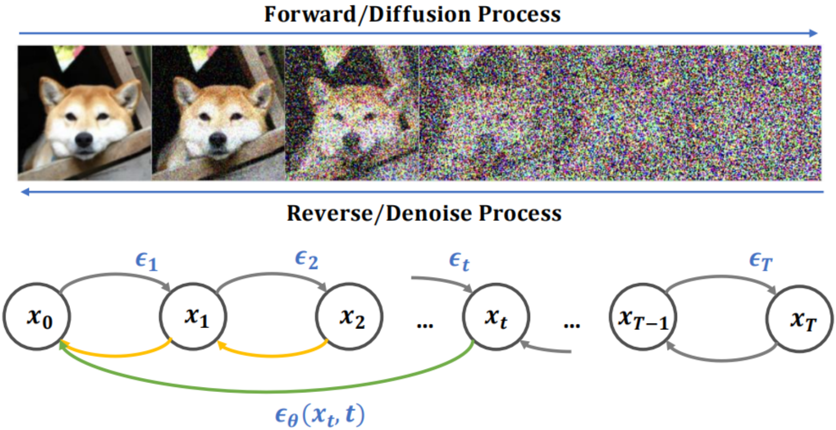
如果p_{data}已知，那么可以显式计算Score，然而生成任务中，我们需要从数据中学习Score $\nabla \log p_{data}$，通过神经网络近似

$$\mathbb{E}[\|s_\theta(x) - \nabla \log p_{data}(x)\|_2^2]$$

其中s_θ为参数为θ的神经网络。

[1] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In Advances in Neural Information Processing Systems, pp. 11895–11907, 2019.

Denoising Diffusion Probabilistic Models, DDPM[2]



[2]Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems, 33, 2020

DDPM

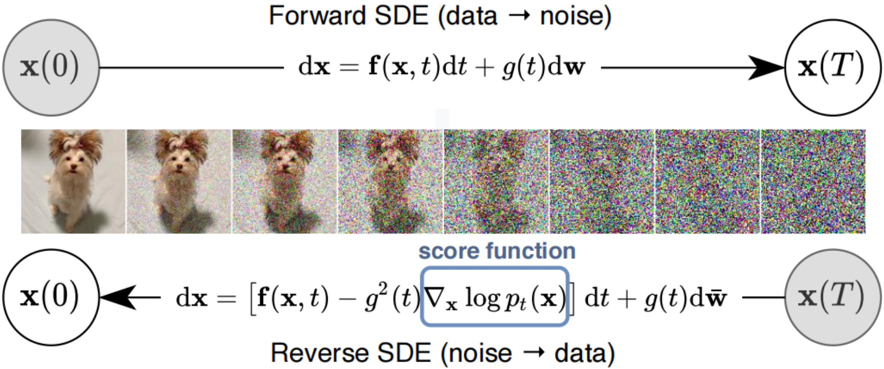
Algorithm 1 Training

```
repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim \mathrm{Uniform}(\{1, \dots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
      $\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar\alpha_t}\mathbf{x}_0 + \sqrt{1-\bar\alpha_t}\boldsymbol{\epsilon}, t) \right\|^2$
until converged
```

Algorithm 2 Sampling

```
$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \dots, 1$ do
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar\alpha_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
end for
return $\mathbf{x}_0$
```

Score-based Generative Models, SGM[3]



[3] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. In Proc. ICLR, 2021.

Score-based Generative Models, SGM

在SDE的框架下：

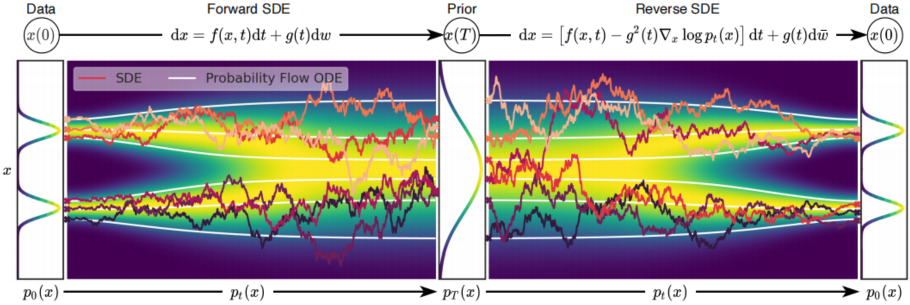
Score Matching Langevin Dynamics 的前向SDE:

$$\mathrm{d}X_t = \sqrt{\frac{\mathrm{d}\left[\sigma^2(t)\right]}{\mathrm{d}t}}\mathrm{d}B_t.$$

Denoising Diffusion Probabilistic Models 的前向SDE:

$$\mathrm{d}X_t = -\frac{1}{2}\beta(t)X_t\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}B_t.$$

Probability Flow ODE



Probability Flow ODE:

$$dx = \left[\mathbf{f}(x,t) - \frac{1}{2}g(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x})\right] dt$$

目前扩散生成模型的研究方向

- 收敛性/收敛速度分析

随机分析方法：Girsanov定理、泛函不等式
离散时间估计方法

- 已有模型的改进

加速，加噪schedule设计

- 新模型的设计

Flow-based Model, Bridge-based Model, Stochastic interpolants

- 离散生成

- 应用/AI for Science

图像、视频/蛋白质设计、药物设计

目录

模拟退火算法

回顾Boltzmann分布/Gibbs测度，在一个由 N 个粒子组成的系统中，当系统处于温度 T 时，粒子在能量为 E_i 的状态下的概率 P_i 由以下公式给出：

$$P_i = \frac{e^{-\beta E_i}}{Z}$$

其中$\beta = \frac{1}{k_B T}$, $Z \triangleq \sum_j e^{-\beta E_j}$.

考察$T \to \infty$时，对于每个状态$i \in \mathcal{S}$,

$$e^{-\beta E_i} = e^0 = 1$$

所以高温状态下，所有可能出现的状态是等概率的

$$\lim_{T \to \infty} P_i = \frac{1}{|\mathcal{S}|}$$

模拟退火算法

考察退火过程，$T \to 0$时，

$$
\begin{aligned}
\lim_{T\to 0} P_i &= \lim_{T\to 0} \frac{1}{Z} \exp\left(-\frac{E_i}{kT}\right) \\
&= \lim_{T\to 0} \frac{\exp\left(-\frac{E_i - E_{\min}}{kT}\right)}{\sum_{i\in\mathcal{S}} \exp\left(-\frac{E_i - E_{\min}}{kT}\right)} \\
&= \lim_{T\to 0} \frac{\exp\left(-\frac{E_i - E_{\min}}{kT}\right)}{\sum_{i\in\mathcal{S}_{\min}} \exp\left(-\frac{E_i - E_{\min}}{kT}\right) + \sum_{i\notin\mathcal{S}_{\min}} \exp\left(-\frac{E_i - E_{\min}}{kT}\right)} \\
&= \begin{cases} \frac{1}{|\mathcal{S}_{\min}|}, & i \in \mathcal{S}_{\min} \\ 0, & i \notin \mathcal{S}_{\min} \end{cases}
\end{aligned}
$$

其中$E_{\min} = \min_{i\in\mathcal{S}} E_i$， $\mathcal{S}_{\min} = \{i : E_i = E_{\min}\}$. 这表明，随着温度降低，粒子会倾向于能量最低的状态中去。

模拟退火算法

模拟退火(Simulated Annealing)是一种优化算法，它的基本想法是：把优化的目标函数 f 当作是物理中的能量函数，然后通过退火，即$T \to 0$，求得 f 的全局最小值。

那么优化问题转化成了：如何采样Boltzmann分布

模拟退火算法能保证收敛到全局最小值[4].

模拟退火能处理组合优化问题(Combinatorial Optimization Problems).

[4] Holley R A, Kusuoka S, Stroock D W. Asymptotics of the spectral gap with applications to the theory of simulated annealing[J]. J. Funct. Anal, 1989, 83(2): 333-347.

目录

Reverse diffusion Sampling

生成模型的本质是从数据分布中采样。对于采样算法，只要找到合适的方式从数据中学习分布，那么就可以产生新的生成模型。

本质上，扩散模型中的Reverse过程即是采样的过程，近一年中，发展了基于Reverse diffusion

$$\mathrm{d}\boldsymbol{x}_t = -\boldsymbol{x}_t \mathrm{d}t + \sqrt{2}\mathrm{d}B_t, \quad \boldsymbol{x}_0 \sim p_0$$

$$\mathrm{d}\tilde{\boldsymbol{x}}_t = (\tilde{\boldsymbol{x}}_t + 2\nabla \ln p_{T-t}(\tilde{\boldsymbol{x}}_t))\mathrm{d}t + \sqrt{2}\mathrm{d}\tilde{B}_t$$

的采样方法[5][6][7]

[5] Huang X, Dong H, Yifan H A O, et al. Reverse diffusion monte carlo[C]//The Twelfth International Conference on Learning Representations. 2024.

[6] Phillips A, Dau H D, Hutchinson M J, et al. Particle Denoising Diffusion Sampler[C]//Forty-first International Conference on Machine Learning. 2024.

[7] Richter L, Berner J. Improved sampling via learned diffusions[C]//The Twelfth International Conference on Learning Representations. 2024.

Thanks & Questions