MCMC与贝叶斯推断简介：从入门到放弃

而今听雨
山东大学 语言学及应用语言学硕士

关注他

> 本文使用 Zhihu On VSCode 创作并发布

写在最前面：这是一份草稿，很多算法我都没加例子和图（虽然很多其他教程也都没加），因为实在是太累了，我也没想到这玩意刨去例子和图都能写这么长。这样**如果哪里讲的不清楚大家跟我在评论里说，我专改那一部分。**

0. 太长不看版以及阅读建议

0.1 太长不看版

贝叶斯推断估计参数的方法是：我们可以算出参数Θ的分布函数$P(\Theta)$，我们用参数分布的数学期望作为对参数的估计值

MCMC的作用是：可以帮我们从任意（无论有没有解析形式的）分布上抽样一批数据，然后用这堆抽样数据的均值作为对这个分布期望的估计

我们用MCMC这种求期望的方法求参数分布期望的估计值，以此求出参数的估计值

0.2 阅读建议

- 如果你只想大致了解MCMC与贝叶斯推断是什么：**只读0.1**
- （⭐新手推荐关卡）如果想大致入门MCMC与贝叶斯推断，**建立概念框架，但不想看繁琐公式和细节**：**只读1、2节不标星号*的部分**
- 如果你有机器学习的知识，不仅想大致入门MCMC与贝叶斯推断，还想知道它跟其他机器学习方法的关系：**读1、2节全部内容**
- （实在太无聊了关卡）如果你对采样方法的整体脉络有兴趣：**参阅第3节**
- 如果你熟悉MCMC的整体框架，但是Markov Chain采样的细节有些疑问：**参阅第4节**
- （新手勇气可嘉、大佬欢迎锤我关卡）如果你除了概念框架，还想了解各种采样方法，并掌握MCMC的一些基础公式的推导：**阅读全文**（新手建议：1、2节带脑子看即可，**第3节建议拿出纸笔**）

0.3 一个小前言

这篇文章最初的框架只有现在的1、2节，原因是现在很多介绍写得都不够低端哈哈哈哈。因为很多人上来就被一大堆分布采样问题给看迷糊了，其实连我们想要做什么都不知道。所以我觉得应该写点什么东西让大家知道采样到底是什么，MCMC大体上在做什么、贝叶斯推断为什么可以用到MCMC，这些简单的问题。

采样问题和技术细节我本来是不打算写的，后来写到这儿了，决定写个框架，因为采样问题的技术细节大佬们比我写得好的，讲的清楚，公式更有美感的多的是。**采样细节问题我也不推荐大家直接来看我这里的东西，我重点写了些大佬们没写到的细节、思路之类的东西（比如大佬们文章里「易得」、「显然」、「容易看出」背后的海量细节2333）**，看了大佬们的文章哪里不清楚再来看看我写进来了没有这样比较好，主要我也害怕误人子弟哈哈哈。

另外证明方面有什么不严谨或者不正确的地方大家直接说就好，有的可能是我表达不到位，有的可能是我理解有误，我会修改的~

1. MCMC是什么

MCMC，Markov Chain Monte Carlo。很多人第一见到这个词可能会奇怪，马尔科夫链和蒙特卡洛两个词是怎么拼到一起的？我们先来看MCMC的后一个MC: Monte Carlo方法到底是什么

1.1 Monte Carlo方法

蒙特卡洛方法是一类方法的统称，简单地说就是，**如果有什么量直接算不好算，你可以把它变成一个随机变量X的统计量V，然后再通过对X进行大量随机采样，通过这些抽样值来估计V的值。**

例如经典的算圆的面积的例子：一个2×2的正方形内接一个半径$r = 1$的圆，在正方形内随机取点，落在圆内的概率为$\frac{\pi}{4}$，于是对这个分布做大量采样，最后就会得到一个接近$\frac{\pi}{4}$的估计值（因为统计频率收敛于概率）

但**现在使用Monte Carlo方法，基本上都是用来算积分的**，这里的积分自然是概率密度函数的积分，所以说也基本上可以理解为Monte Carlo方法基本上是被用来算期望的。这是因为使用Monte Carlo方法算期望，大数定理可以保证它收敛于期望。

对于随机变量X，它的概率密度函数为$p(x)$，因此它的数学期望为：

$$E(x) = \int_{-\infty}^{+\infty} xp(x)dx$$

我们对于这个随机变量随机采样得到n个采样值x_i，根据大数定理，有：

$$\lim_{n \to +\infty} \frac{1}{n} \sum_{i}^{n} x_i = E(X)$$

所以**最常见的一种Monte Carlo方法的使用场景就是：对随机变量进行充分多的采样后，使用这些采样的均值来估计总体的期望**（其实是个非常trivial的事情，多抽样几个数据平均一下来估计总体期望嘛不就是）

1.2 Markov Chain与Monte Carlo的关系

一言以蔽之：在Monte Carlo方法的**采样**过程中，**使用了Markov Chain作为采样方法**的方法，称为Markov Chain Monte Carlo方法。

方法整体是一个蒙特卡洛方法，而MCMC这种蒙特卡洛方法的核心在于使用了马尔科夫链来做采样，所以它叫马尔可夫链蒙特卡洛方法。

至于怎么用Markov Chain来采样是个技术细节，后面会简单说一说，这里帮助大家构建概念网络就够了。

1.3 这部分的FAQ

1.3.1 采样(Sample)是什么?

采样就是抽样。

就是给定一个随机变量X的分布$f(x)$，你怎么得到它的若干采样数值。

又或者说，计算机怎么样来**自动生成服从$f(x)$分布的随机数。**如果你觉得这件事很容易，请你思考第3节的每一个小节标题。

注意：采样指的**并不是**给你一个概率密度函数$f(x)$，然后你采样出若干个函数上的点$(x, f(x))$；采样指的是给你一个概率密度函数$f(x)$，**你要得到一系列的数据，$\{x_0, x_1, \dots, x_n\}$使得这些数据些服从概率密度函数$f(x)$**

1.3.2 *MCMC跟通常的最优化方法有什么区别和联系?

联系：目标函数$f(x)$都是一个可能有很多维的，很可能没有解析表达式的函数。虽然**给定任意的x，我们都能算出来$f(x)$值**，但是我们想知道一些**与这个函数的某些整体性质相关的东西**。它们的区别就是各自关心的整体性质是什么

区别：

1. 最优化方法是想知道$f(x)$的**最值/极值**在哪儿
2. MCMC是想知道，如果X服从$f(x)$这个概率分布，我**怎么获得$E(X)$**

2. MCMC与贝叶斯推断有什么关系

我想，贝叶斯推断应该是绝大多数人学习MCMC的原因

2.1 贝叶斯推断简述

设你有一个模型，模型中包含一系列的参数Θ，并且观测到了一系列数据D，我们可以通过贝叶斯公式得到：

$$P(\Theta|D) = \frac{1}{P(D)}P(D|\Theta)P(\Theta)$$

由于$P(D)$是一个无关紧要的常数，因此上式往往直接写成一个正比关系式：

$$P(\Theta|D) \propto P(D|\Theta)P(\Theta)$$

在贝叶斯推断里：

1. **你的任务是通过$P(\Theta|D)$来得到Θ的估计值**
2. 你**模型的作用给出$P(D|\Theta)$**，也就是说给定某个猜测的Θ值时，模型要利用这些参数算出观测到目前这些数据D的概率是多少，也即likelihood$P(D|\Theta)$
3. 你还可以**通过$P(\Theta)$来对参数的分布情况做一些先验的猜测。**先验项(prior)$P(\Theta)$表达的是参数的先验分布（你根据你的知识猜想的分布），也就是Θ比较可能的取值是哪些，不太可能的取值是哪些，当然如果你什么都不知道，$P(\Theta)$自然可以猜一个均匀分布

那么问题就来到了我们怎么通过等式左侧的$P(\Theta|D)$来获取到Θ的估计值

2.2 怎么通过后验概率$P(\Theta|D)$获取参数Θ的估计值

我们上面表明了$P(\Theta|D)$可以被算出来，但是可以被算出来不代表我们能拿到它的表达式，更不代表它有好看的表达式。所以我们有一些很自然的想法：

1. 使用后验概率的**众数**作为估计值
2. 使用后验概率的**期望**作为估计值

下面的东西写给有机器学习基础的同学。太长不看版是这样：算期望在思路上更简单，因为算一个分布的期望大家一般怎么算呢？**多抽样几个数据算个平均值不就是了！**很好，**MCMC就是教我们怎么在一个没有解析形式的数据上「抽样几个数据算平均值」的方法**，所以贝叶斯推断经常用MCMC来估计参数。

2.2.1 *后验众数(Posterior Mode)

通过后验众数来求参数的估计值，被称为最大后验概率估计(Maximum A Posteriori Estimation, MAP)。

插播一个语言学问题：在当代英语技术论文中，a priori和prior、a posteriori和posterior是可互换的，但词形不同、读音不同，Maximum A Posteriori Estimation里的A Posteriori等于Posterior，是一整个词，并不是一个冠词+名词。

它跟我们高中数学就学过的最大似然估计(Maximum Likelihood Estimation, MLE)有着深刻的爱恨纠葛。

以下2.2.1.1和2.2.1.2讲MLE和MAP的关系，没兴趣的朋友请跳过，不影响阅读。

2.2.1.1 最大似然估计MLE

我们前面说过，Likelihood指的是这个东西：$P(D|\Theta)$，所以MLE估计的是，给定哪种参数时，观察到目前观察到的数据的概率最大。思路就是，我现在已经观察到数据了，那么我们认为这应该是一件trivial的事情，所以不妨让观察到这些数据的概率是最大的。

2.2.1.2 最大后验概率估计MAP

A Posteriori也就是Posterior指的这个东西：$P(\Theta|D)$，所以MAP估计的是，给定目前观察到的数据，我们最有把握确定的参数应该是什么。并且根据贝叶斯定律：

$$P(\Theta|D) \propto P(D|\Theta)P(\Theta)$$

如果你把它变变形：

$$\frac{P(\Theta|D)}{P(\Theta)} \propto P(D|\Theta)$$

你就会发现，如果你对参数的分布没有先验知识，那么：

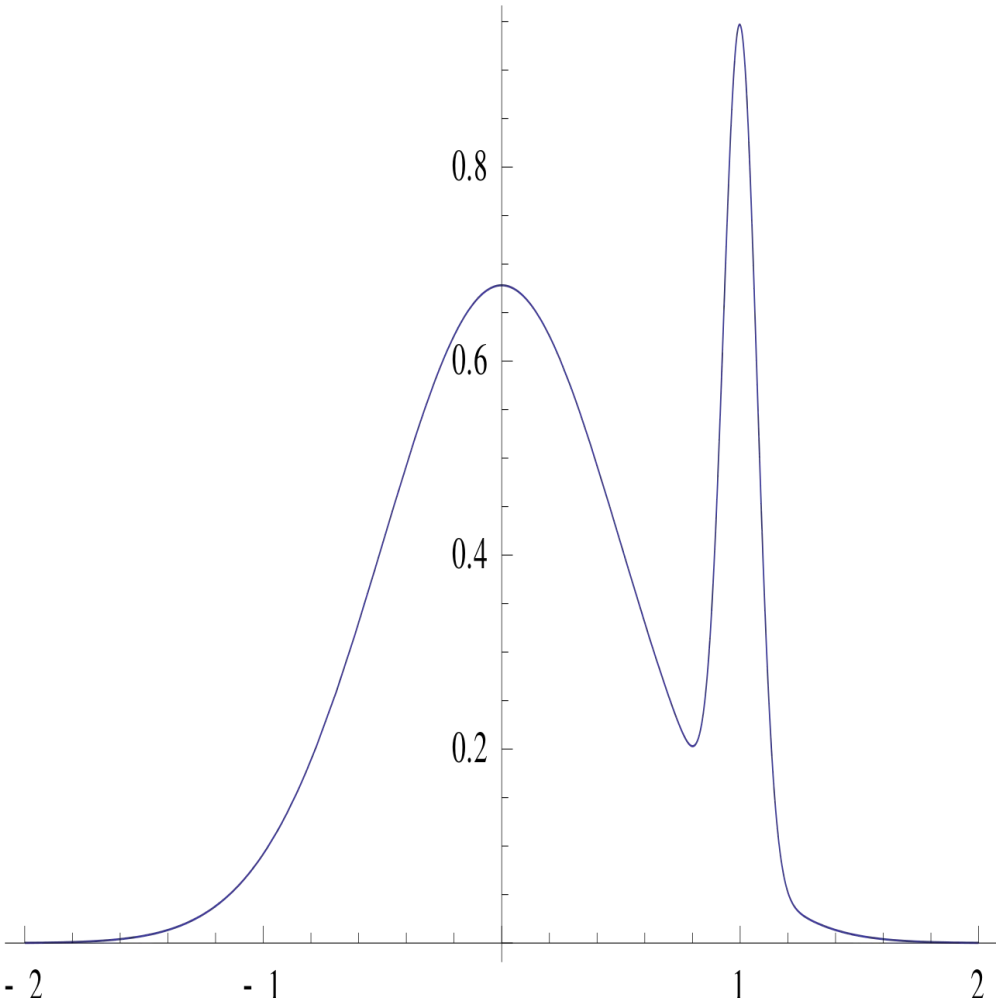
$$P(\Theta|D) \propto P(D|\Theta)$$

求MLE和MAP就是一回事……

2.2.1.3 怎么求MAP

如果求后验分布的众数，你本质上是在**求一个最优化问题**，也即求$\mathbf{argmax}\,\mathrm{p}(\Theta|\mathrm{D})$，这时候你可以随便用任何一种机器学习里的最优化算法来求这个最值，比如如果后验概率是个凸函数，你可以梯度下降（一阶可微）或者牛顿法（二阶可微）。

但是，MAP真的就那么合理吗？我们想要的真的是后验分布最大的那个点吗？维基这张图非常有助于理解：



img

也就是说，**有时候后验众数MAP方法虽然看上去合理，但是并不一定真的能代表分布的整体特性。所以我们不妨考虑一下后验分布的期望。**

2.2.2 后验期望(Posterior Mean)

如果求后验分布的期望，这本质上是在算一个类似于这样的积分：

$$\int_\theta \theta p(\theta) d\theta$$

这时候我们很自然地就想到了蒙特卡洛方法：只需要对$p(x)$做采样，大数定理保证了采样均值收敛于期望。

其实不要被公式骗了，**这是一个非常非常trivial的想法：想知道一个分布的期望什么，多抽样几个数据算个平均值就是了！**

而对复杂函数$p(x)$做采样，非常高效的一种方法就是马尔科夫链采样法。

因此我们说，**我们通过马尔科夫链（采样的）蒙特卡洛（估计）方法，求贝叶斯推断的（参数）后验分布的数学期望，以此作为参数的估计值**

2.3 总结：为什么贝叶斯推断常用MCMC来估计参数的值

1. **MCMC是一种推断分布期望的方法**
2. **贝叶斯推断的主要步骤是求后验分布的期望**
3. **所以我们用MCMC求贝叶斯推断后验分布的期望**

读到这里你大概已经明白了MCMC和贝叶斯推断的基本原理了，采样问题，比我讲的清楚的文章就多得是了，如果你还想看我讲，那我就简单捋一下。

预警：下面开始公式变多

3. *采样方法概述

采样问题大致等价于**计算机怎么来自动生成服从$f(x)$分布的随机数**

这部分我的图很少，大部分需要你拿起笔来自己动手推一推，因为**我发现在采样问题上，画一个概率密度函数的图，很容易让人联系到在函数曲线上采样，可是采样值全在x轴啊！**所以画图无益，不如推公式。

3.0 采样方法的思想

发现很多人这部分看得云里雾里，隔了好几个月再来这里加个3.0。提纲挈领地说，所有的采样方法的基本思路可以分为两类：

- **递推形式**：我先随便给出一个初始值，然后后面的值根据上个生成的值按照某种规则递推出来，这种规则保证了递推出来的这列数整体上服从某个分布。有几个要点：

1. **有的递推算法只要初始值相同，就会生成相同的伪随机数序列**，比如线性同余发生器。这个初始值，就是你写代码生成随机数的时候经常看到的那个所谓的随机种子，一般采用当前的时间戳之类的数，这样不会重复。
2. 相比下面那种形式**速度很快。**
3. 需要动脑子设计生成规则，想要让递推结果整体上满足某种分布往往不是那么容易的。

- **Proposal-Accept形式**：*名字是我自己起的*，这类方法的思路是先用一个已知的采样方法生成随机数，然后再通过一个随机概率来决定采纳不采纳这个结果，最终使得整体的结果接近想要的分布。有几个要点：

1. 那个**已知的采样方法必须能cover掉你目标分布的取值范围。**
2. 这类算法往往速度**比较慢**，优化目标是让proposal被采纳的概率尽可能地大，否则算法会面临大量的rejection。
3. 对于**任意分布都能搞定**，如果你不介意效率的话。

下面方法中：

1. 线性同余发生器和Gibbs采样本质上都是递推形式的采样方法，速度都很快，但是都有限制，前者只能生成均匀分布，后者只能生成边缘分布可以采样的分布。
2. 拒绝采样、重要性采样、MH采样本质上都是proposal-accept这个思路，MH采样的accept函数设计得很巧妙，所以速度往往比前两者好一些。前两者效率有可能火葬场（一堆rejection，如果你的分布不理想的话），虽然极端分布下，MH采样同样有可能跪在一堆rejection面前。但是这些方法都是可以对任意确定的概率密度函数做采样的，通用性很好。

此外，**对已知分布的采样结果做某些数学变换来产生一个新分布的结果**同样是非常重要和常见的方法，但是这本质上不是一个采样方法的思路（本质上只是对结果的变换），所以就不列了。

3.1 均匀分布的采样

首先不需要讨论的是怎么生成均匀分布的采样，这基本上都是基于一些伪随机数算法生成的伪随机数。例如线性同余发生器：

$$x_{n+1} = (ax_n + c) \pmod M$$

然后你只需要随便选个x_n让算法开始递推下一个随机数就OK，这样上面算法就生成了$\{0, 1, \dots, M-1\}$的伪随机序列，并且显然是个周期序列，周期$T \leq M$

让上面的序列x_n/M，并且取恰当的a, c, M，你可以得到一个充分好的服从$U(0,1)$的均匀分布（毕竟要知道浮点数的精度是有限的，所以这个算法做到充分好并不那么难），那么$U(0,1)$的线性变换就可以得到任何均匀分布

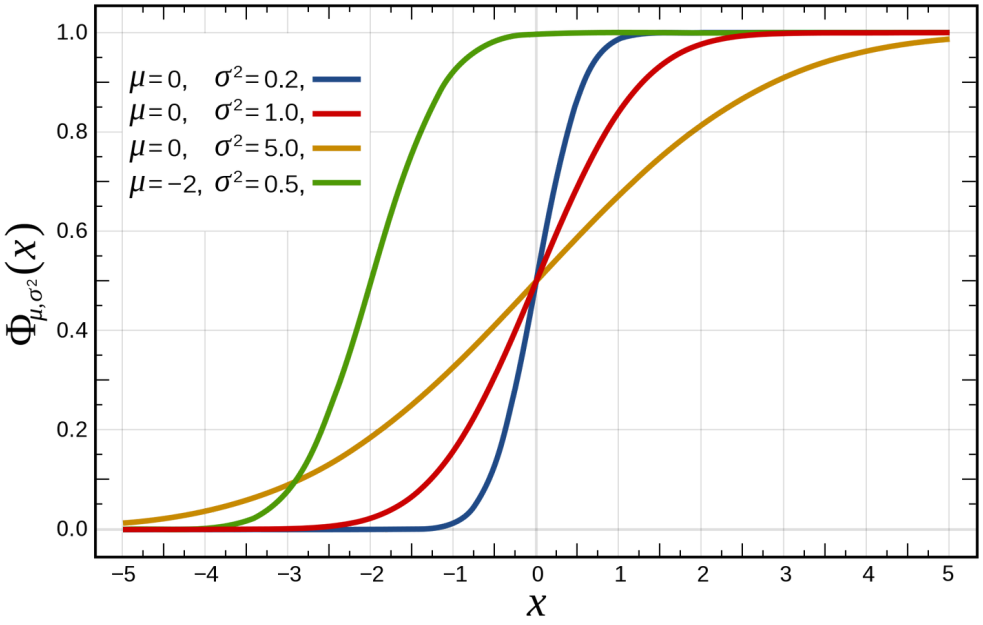
其余的伪随机算法请自行google

总之这些都是你直接 `random.random` 或者 `numpy.random.rand` 的原理.....

3.2 逆变换采样(Inverse Sampling)：从均匀分布到正态分布

除了均匀分布，最常用的分布自然是正态分布。通常大家讲正态分布的采样会直接甩一个Box-Muller 变换（不许说是 `numpy.random.randn` ！），但是不告诉你为什么。

其实原理很简单，我们现在已经有了一个$U(0,1)$，那么思考$[0,1]$这个区间，是不是很自然地就想到了累积分布函数的值域？任何分布的累积分布函数CDF值域都是从0到1，例如正态分布的CDF如图：



Cumulative distribution function for the Normal distribution

那么假设一个随机变量的累积分布函数cdf为$y = f(x)$，其中x, y分别为样本值和取值范围为$[0,1]$的变量y，此时的变量y是服从均匀分布的，那么如果取反函数$x = f^{-1}(y)$，再令y为均匀分布的采样，我们就可以通过一个均匀分布的采样，经过反累积函数的变换，就可以得到任意分布的采样。

Box-Muller 变换为啥是同时两个采样？回去想想怎么给正态分布的概率密度函数积分你就明白了，直接给正态分布的cdf取反函数，是没有初等表达式的，但是拆成两个变量，做个三角代换，就有初等表达式了.....

所以其实问题就来了，**不一定所有分布累积分布的反函数都有初等表达式，或者好算的方法**，这才是采样问题的难点。

3.3 拒绝采样(Rejection Sampling)

拒绝采样很多讲解直接给图，看图是那种你乍一看会明白，仔细一想总觉得哪里不对。所以我就不给图了，请拿起笔自己推：

假设我有一个pdf为$f(x)$的分布要做采样，但是我不会做。然后我有一个pdf为$g(x)$的分布，这个分布采样我会做，可以是正态分布也可以是均匀分布（如果$f(x)$的定义域也是$[a,b]$这种闭区间的话）

那么我先按照$g(x)$采一个样x_i，**这时候取x_i的概率密度是$g(x_i)$，我想要把它变成$f(x_i)$，我要怎么做呢?**

很显然，只要配一个合理的项p等式就成立了：

$$f(x_i) = pg(x_i) = \frac{f(x_i)}{g(x_i)}g(x_i)$$

这是一句废话，因为这个式子现在毫无意义

但是一个巧妙的主意是：**如果p代表一个概率，这个式子突然就有意义起来了!**

我只需要在以$g(x_i)$的概率密度取到它之后，再以$p = \frac{f(x_i)}{g(x_i)}$的概率接受它（否则就拒绝它），这样上面这个式子不就成立了!

虽然这个想法很美丽，但是聪明的你可能发现了盲点：我怎么保证$p = \frac{f(x_i)}{g(x_i)} \leq 1$恒成立呢？如果这个数还有可能大于1，它显然不能被当作一个概率项。**实际上，$p \leq 1$几乎可以说恒不成立**，因为当如果两个分布不是同一个分布的话，你不可能保证概率密度函数恒有$g(x_i) \leq f(x_i)$，毕竟他们的积分都是1。

所以我们退而求其次，用一个骚操作：

$$\frac{1}{k}f(x_i) = p'g(x_i) = \frac{f(x_i)}{kg(x_i)}g(x_i)$$

你只需要保证$kg(x_i) \geq f(x_i)$恒成立就行了，你取充分大的k，这个肯定必然成立。**这个时候$p' = \frac{f(x_i)}{kg(x_i)}$就可以被当作一个概率项了。**

多说一句，你求一个$p' = \frac{f(x_i)}{kg(x_i)}$这个概率的方法很简单，就是在$U(0, kg(x_i))$的均匀分布上采个样，比$f(x_i)$小就接受，大就拒绝，当然你有其他花哨的方法实现这个概率也随便，所以注意，无论你在任何地方看到这个均匀分布采样，不要做过多的思考，它跟你选定的那个分布$g(x)$的采样没有一丝关系。**这个均匀分布采样不重要，只是教你一个最简单的实现p'的方法而已。**

这样你可以保证采样到一个服从$f(x)$的采样，问题在于你的采样概率也是$\frac{1}{k}f(x)$这样一个概率，k越大，你接受采样的概率就越小，采样效率可能很低很低

3.4 重要性采样(Importance Sampling)和采样重要性重采样

注意：**通常说的Importance Sampling是一种通过采样来估计期望的方法**（所以其实是一种Monte Carlo方法），而**不是一种采样方法**，但是我们可以对采样得到的序列重新采样来做采样，这称为Sampling Importance Resampling(SIR)。都说一下

3.4.1 Importance Sampling

重要性采样，或者我觉得是不是翻译成权重采样更好呢?

总之Importance Sampling是为了算分布$f(x)$的数学期望，也就是积分：

$$E(f) = \int_x xf(x)dx$$

假设有一个分布$g(x)$可以取得采样，那么积分可以变为：

$$E(f) = \int_x x\frac{f(x)}{g(x)}g(x)dx$$

如果假设分布$g(x)$的累积分布函数为$P = G(x)$，那么$\frac{d}{dx}G(x) = g(x)$，于是：

$E(f)=\int_x x\frac{f(x)}{g(x)}g(x)dx=\int x\frac{f(x)}{g(x)}dP$

令$x'=x\frac{f(x)}{g(x)}$，那么：

$E(f)=\int x\frac{f(x)}{g(x)}dP=\int x'dP$

如果令随机变量X满足$P(X\leq x\frac{f(x)}{g(x)})=P=G(x)$，则$E[f]=E[X]$

所以我们就用$g(x)$来采样，采样出来的x_i按照$x'=x\frac{f(x)}{g(x)}$变成x_i'，那么x_i'的均值是$E[X]$的近似，因此也是$E(f)$的近似

说到底，**如果$g(x)$是一个均匀分布，这跟普通的无限分割求定积分没有任何区别**

但是特定情况下，**使用特殊的$g(x)$可以减小估计的方差**，这才是目的

归根结底，这是个蒙特卡罗方法，而不是个采样方法

3.4.2 Sampling Importance Resampling

类似思想的采样方法叫做Sampling Importance Resampling。这个名字大概可以翻译成「采样重要性重采样」。正确的断句应该是：**（根据上次采样得到的采样权重的）重新采样**

还是假设我有一个pdf为$f(x)$的分布要做采样，但是我不会做。然后我有一个pdf为$g(x)$的分布，这个分布采样我会做

采样

先采样若干$x_i(1\leq i\leq n)$使得它们服从$g(x)$

计算重要性

对于$i=1,\dots,n$，令$c_i=\frac{f(x_i)}{g(x_i)}$，再做归一化：$p_i=\frac{c_i}{\sum_{i=1}^n c_i}$。你可能看明白了，$c_i$就对应$x_i$被取到的权重，归一化之后，$p_i$就可以被当作概率来用。

重采样

对于$j=1,\dots,m$，令$y_i=x_i$的概率为p_i

如果你想不到怎么做，那就还是采样一个$u_i\sim U(0,1)$，然后：

$$y_i=\begin{cases}x_1, 0\leq u_1\leq p_1\\ \dots\\ x_i, \sum_{j=1}^{i-1}p_j\leq u_i\leq\sum_{j=1}^{i}p_j\\ \dots\\ x_n, \sum_{j=1}^{n-1}p_j\leq u_n\leq 1\end{cases}$$

上面这个东西就被叫做采样重要性重采样SIR

你如果想问这个东西为什么work，那么**其实它本质跟Rejection Sampling没有什么区别**：

对于任意x_i在第一步被采样到的概率密度是$g(x_i)$，在第二步被采样到的概率密度是$p_i=\frac{c_i}{\sum_{i=1}^n c_i}$，那么整体被采样到的概率密度就是：

$\frac{1}{\sum_{i=1}^n c_i}g(x_i)c_i=\frac{1}{\sum_{i=1}^n c_i}f(x_i)$

如果记$k = \sum_{i=1}^{n} c_i$，那么上式变为

$$\frac{1}{k} f(x_i)$$

是不是跟Rejection Sampling没有什么区别，唯一的区别在于这里给出了一个切实可行的让你算k的方法，而Rejection Sampling里没说（有很多骚方法，但是没这种普适）

终于要开始正题了~

预警：长篇公式预警

4. 马尔科夫链采样

简单说说思路吧。马尔可夫链采样是一个绝妙的主意。

对于采样问题，一个理想的状态是，**随便输入一个什么值，然后算法都能按照指定的概率分布来递推下一个随机样本的值。**例如我们上面说过的线性同余发生器：

$$x_{n+1} = (ax_n + c) \pmod M$$

对上一个值做线性变换再对一个足够大的数取模，就可以递推来生成一串近似服从均匀分布的序列。

我们想，**如果有个什么东西，能够对于任意分布，都能从上一个采样数据按照分布递推出下一个采样数据就好了**，而我们发现，有个东西天然具有这种性质，那就是稳态马尔科夫链。

4.1 马尔科夫链的稳态

如果你学过一些线性代数和概率论，你应该知道什么是稳态马尔科夫链，如果你不知道，我们干脆抛开马尔科夫链这些名词，直接定义一些概念和结论：

假设我们现在有若干随机变量$X1, \ldots, X_t, \ldots$，任意随机变量X_t的概率密度函数是$p_t(x_t)$，注意这里使用x_t表示一个变量，而不是一个具体的数值，例如x_t和x_{t+1}是两个完全不同的变量。

再定义一个多元函数$k(x, y) = P(X_{t+1} = y | X_t = x)$，它是一个条件概率函数，定义了两个相邻的随机变量$X_t, X_{t+1}$之间存在概率关系。

为了强调两个变量之间是条件关系，我们下面把这个函数写成$k(y|x)$或$k(x_{t+1}|x_t)$的形式

那么显然有递推式：

$$p_t(x_t)k(x_{t+1}|x_t) = p_{t+1}(x_{t+1})$$

也就是说，第一个随机变量X_1的分布$p_1(x_1)$一旦给定，后面所有随机变量的概率分布都是递推出来的。

（学过Markov Chain的同学：这其实就是一条由无限个状态的马尔科夫链，$p_t(x)$是t时刻各状态的概率分布，$k(x, y)$是状态转移函数）

那么显然从第一个随机变量的分布推到t是一个这样的递推式：

$$p_t(x_1)[k(x_{t+1}|x_t)]^{t-1} = p_t(x_{t+1})$$

以下是我不打算给大家证明的，感兴趣的同学可以自行google的定理：

可以证明，当$k(x, y)$满足一定条件，且t充分大时，有：

$$p_{t+1}(x_{t+1}) = p_t(x_t)k(x_{t+1}|x_t) = p_t(x_t) = p(x)$$

我们称$p(x)$为平稳分布，且其充分条件为：

$$p(x_t)k(x_{t+1}|x_t) = p(x_{t+1})k(x_{t+1}, x_t)$$

我不打算给大家证明的，感兴趣的同学可以自行google的定理到此为止

4.2 Markov Chain采样

现在有一个绝妙的主意，如果我们有个不会采样的分布$f(x)$，如果我们能找到它对应的$k(x, y)$，我们就可实现我们刚才的想法。

首先随便生成一个x_0（从什么分布里生成都无所谓，假设从 $g(x)$ 这个已知采样方法的分布比较好），然后反复通过$P(x_{t+1}|x_t) = k(x_{t+1}|x_t)$的概率来生成下一个采样值
$x_1, x_2, \dots x_n, x_{n+1}, \dots$。必然有：

$$\lim_{t\to\infty} g(x)[k(x, y)]^t = f(x)$$

假设当$t \geq n$时，我们认为它充分大了，可以近似认为或者实际上的确已经收敛了，那么折之后的所有采样，也就是所有的：

$$\{x_i|i \geq n\}$$

都可以视为近似服从$f(x)$的分布。

注意一个问题，我们直接抛弃掉了所有 $t < n$ 的数据，因为我们认为这时候分布还没有达到稳态，这些数据是不合格的采样值。**有很多人把这个行为称为burn-in**，说是把前面的数据给烧掉了。其实我更乐意管这个过程叫做warm-up，它是在热身，或者像你冬天启动车子的时候需要「暖车」一样，**前面这些采样只是为了让模型更接近稳态的热身过程**，所以不要理解成前面这些也是采样值，但是我们给抛弃掉了。

现在问题就变成了给定$f(x)$怎么找到$k(x, y)$了

4.2.1 配一个$k(x, y)$出来

如果我们随便找一个$q(x, y)$会发生什么呢？显然：

$$f(x_t)q(x_{t+1}|x_t) \neq f(x_{t+1})q(x_t|x_{t+1})$$

那我们能不能把它凑成相等呢？例如：

$$f(x_t)q(x_{t+1}|x_t)\alpha(x_t, x_{t+1}) = f(x_{t+1})q(x_t|x_{t+1})\alpha(x_{t+1}, x_t)$$

这样我就可以说我们找到$k(x, y)$了：

$$k(x, y) = q(x, y)\alpha(x, y)$$

那么怎么确定$\alpha(x, y)$呢，我们直接利用对称性暴力配平：

当$x_{t+1} \neq x_t$时（这是重点，先记住，一会会考），令

$$\alpha(x, y) = f(y)q(y, x)$$

于是上面变成：

$$f(x_t)q(x_{t+1}|x_t)f(x_{t+1})q(x_t|x_{t+1}) = f(x_{t+1})q(x_t|x_{t+1})f(x_t)q(x_{t+1}|x_t)$$

显然成立

所以我们知道了：

$$k(x_{t+1}|x_t) = P(x_{t+1}|x_t) = q(x_{t+1}|x_t)\alpha(x_t, x_{t+1}) = q(x_{t+1}|x_t)f(x_{t+1})q(x_t|x_{t+1})$$

4.2.2 怎么执行这个$k(x, y)$

这个式子看起来没什么卵用，因为我们现在知道了x_t，我们要按照$P(x_{t+1}|x_t)$这个概率获取x_{t+1}，可是我们在还没取到x_{t+1}的时候怎么能算出来式子右端的后面两项，也就是$\alpha(x_t, x_{t+1}) = f(x_{t+1})q(x_t|x_{t+1})$

时刻记住：并不是公式里有这一项，你写程序的时候就一定能算出这一项来

那么最简单的方案就是分步骤来：

1. 先按照$q(x_{t+1}|x_t)$的概率分布采样获得一个x_*
2. 有了x_*之后，计算$\alpha(x_t, x_*)$，由于它一定是个$[0, 1]$范围上的数，我们把它当作一个概率：

- 以 α 的概率接受$x_{t+1} = x_*$
- 以$1 - \alpha$的概率接受$x_{t+1} = x_t$

好了，我们要时刻对「采样」这个词保有怀疑：你怎么能在第一步说采样一个$q(x_{t+1}|x_t)$就能采样一个$q(x_{t+1}|x_t)$呢？所以虽然理论上我们的$q(x_{t+1}|x_t)$是任意选取的，实际上我们只能选那些我们知道怎么来采样的分布，例如高维的正态分布，迪利克雷分布等等。

这样我们把上面的两个东西起个名字：

- $q(x_{t+1}|x_t)$叫做proposal distribution，我们每次都用它来生成新备选
- $\alpha(x_t, x_*)$叫做acceptance ratio，我们每次用它来决定是使用新备选还是使用上一个采样值（原地不动）

4.2.3 一个遗留问题：$P(x_t|x_t)$

慢着，为什么要以$1 - \alpha$的概率接受x_t作为新的采样值呢？不应该是放弃掉这次采样重新采吗？**这是一个很关键的问题。**

回到开始我们在配平α的时候，我说$x_t \neq x_{t+1}$时我们可以配出来α，那而当$x_{t+1} = x_t$时呢？

$\alpha(x, y) = f(y)q(y, x)$这个结论是从等式
$f(x_t)q(x_{t+1}|x_t)\alpha(x_t, x_{t+1}) = f(x_{t+1})q(x_t|x_{t+1})\alpha(x_{t+1}, x_t)$中配出来的，但是
$x_t = x_{t+1}$时是推不出任何关于α的信息的，因为这个时候它等于任何值等式都成立

所以

$$P(x_{t+1}|x_t) = q(x_{t+1}|x_t)\alpha(x_t, x_{t+1}) = q(x_{t+1}|x_t)f(x_{t+1})q(x_t|x_{t+1})$$

当且仅当$x_t \neq x_{t+1}$时成立

仔细观察一个问题，如果上面的α对于任意x_{t+1}都成立的话，显然有：

$$\int_{x_{t+1}} q(x_{t+1}|x_t)dx_{t+1} = 1$$

而$\alpha(x_{t+1}|x_t) <= 1$显然对任意x_{t+1}成立，且并不总能取到等号，于是：

$$\int_{x_{t+1}} P(x_{t+1}|x_t)dx_{t+1} = \int_{x_{t+1}} q(x_{t+1}|x_t)\alpha(x_t, x_{t+1})dx_{t+1} < 1$$

这是个大问题，其实按照这样来算α，你得到的$k(x_t, t_{t+1})$积分积不到1（概率密度函数积分积出比1小的数这河里嘛）

那么，既然我们算的结果在$x_t \neq x_{t+1}$时都成立，也就是说只有$x_t = x_{t+1}$的时候的概率是不对的，那么我们把剩下的概率就匀给$x_t = x_{t+1}$就是了，准确地说，用上面那个α算出来的$P(x_t|x_t)$会比真实概率小，差值是剩下的那些概率，应该匀给它：

$$
\begin{aligned}
P(x_t|x_t) - q(x_t|x_t)\alpha(x_t|x_t) &= 1 - [\int_{x_i} q(x_t, x_i)\alpha(x_t, x_i)dx_i] \\
&= \int_{x_i} q(x_t, x_i)dx_i - [\int_{x_i} q(x_t, x_i)\alpha(x_t, x_i)dx_i] \\
&= \int_{x_i} q(x_t, x_i)[1 - \alpha(x_t, x_i)]dx_i
\end{aligned}
$$

你会发现我们匀给它的概率，就是当$q(x_t, x_i)$采样出任意一个x_i时，$1 - \alpha$的概率。

4.3 MH采样

如果你看明白了上面的东西，下面的内容就是玩……

太长不看：MH采样是一种通过对α变形来加快收敛速度的MC采样方法

上面的$\alpha(x, y)$虽好，但是很多时候，这个东西：

$$\alpha(x_{t+1}|x_t) = f(x_{t+1})q(x_t|x_{t+1})$$

它实在是太小了，它小的后果在于整个式子$\lim_{t\to\infty} g(x)[k(x, y)]^t = f(x)$收敛得慢，也就是我们可能得需要一个十分大的$n$，当$t > n$时我们采样的才开始有效，这是我们比较难以接受的。

所以我们需要做点小trick。观察：

$$f(x_t)q(x_{t+1}|x_t)\alpha(x_{t+1}|x_t) = f(x_{t+1})q(x_t|x_{t+1})\alpha(x_t|x_{t+1})$$

不妨设$\alpha(x_{t+1}|x_t) \leq \alpha(x_t|x_{t+1})$，那么移项：

$$f(x_t)q(x_{t+1}|x_t)\frac{\alpha(x_{t+1}|x_t)}{\alpha(x_t|x_{t+1})} = f(x_{t+1})q(x_t|x_{t+1})1$$

如果令$A(x_{t+1}|x_t) = \min\{\frac{\alpha(x_{t+1}|x_t)}{\alpha(x_t|x_{t+1})}, 1\}$，我们就有：

$$f(x_t)q(x_{t+1}|x_t)A(x_{t+1}|x_t) = f(x_{t+1})q(x_t|x_{t+1})A(x_t|x_{t+1})$$

很漂亮的公式，且$A(x, y) \leq 1$恒成立，所以它可以是个概率值，使用了两个α的比值，那么A肯定整体上比它大多了，那么我们用A来代替上文里的α就是一个收敛得更快的方法

4.4 Gibbs采样

Gibbs采样是MH采样的特殊情况

Gibbs采样与MH采样的不同在于proposal distribution $q(x, y)$是指定的，而后者是任取的

Gibbs采样指定$q(x, y)$的目的是使得$A(x, y) = 1$恒成立从而加快收敛与采样速度

太长不看：Gibbs采样是一种指定了$q(x, y)$从而使得$A(x, y) = 1$恒成立的MH采样的特殊情况

上面的变量，是标量还是向量都没问题，但是Gibbs采样必须是向量。我们换成向量表述：

$$\boldsymbol{x_t} = [x_t^1, x_t^2, \ldots, x_t^n]^T$$

如果变量服从$\boldsymbol{f(x)}$这样一个联合分布，我们如何使用MH采样法采样呢？我们需要先确定一个proposal distribution $\boldsymbol{q(x)}$，这里我们选择这样的一个分布：

$$q(\boldsymbol{x_{t+1}}|\boldsymbol{x_t}) = f(x_t^i|x_t^1x_t^2\ldots x_t^{i-1}x_t^{i+1}\ldots x_t^n)$$

为了方便，我们下面记：

$$x_t^1 x_t^2 \dots x_t^{i-1} x_t^{i+1} \dots x_t^n \to x_t^{-i}$$

这样：

$$q(x_{t+1}|x_t) = f(x_{t+1}^i|x_t^{-i})$$

用文字描述一下，上面式子的意思是，从x_t到x_{t+1}这一步，我只改变向量x_t的第i个维度，其他的维度保持不变，而proposal distribution就是当其他值都取原来的值的时候，第i个维度上的条件概率。

这个时候我们来算一下MC采样法里的α：

$$\begin{aligned}\alpha(x_{t+1}|x_t) &= f(x_{t+1})q(x_t|x_{t+1}) \\ &= f(x_{t+1})f(x_t^i|x_t^{-i}) \\ &= f(x_{t+1}^i|x_t^{-i})f(x_t^{-i})f(x_t^i|x_t^{-i})\end{aligned}$$

$$\begin{aligned}\alpha(x_t|x_{t+1}) &= f(x_t)q(x_{t+1}|x_t) \\ &= f(x_t)f(x_{t+1}^i|x_t^{-i}) \\ &= f(x_t^i|x_t^{-i})f(x_t^{-i})f(x_{t+1}^i|x_t^{-i})\end{aligned}$$

我们来算一下MH采样法里的A：

$$\frac{\alpha(x_{t+1}|x_t)}{\alpha(x_t|x_{t+1})} = \frac{f(x_{t+1}^i|x_t^{-i})f(x_t^{-i})f(x_t^i|x_t^{-i})}{f(x_t^i|x_t^{-i})f(x_t^{-i})f(x_{t+1}^i|x_t^{-i})} = 1$$

于是

$$A(x_{t+1}|x_t) = \min\{1, 1\} = 1$$

证毕

因此每次proposal distribution产生的新x_*我们可以直接接受为下一个样本$x_{t+1} = x_*$

但是需要注意的是：使用Gibbs采样的前提是你这个分布的条件分布必须好算，也就是说它并不是任意分布都能直接来采样的，你必须保证 $q(x_{t+1}|x_t) = f(x_{t+1}^i|x_t^{-i})$ 这个东西是好采样的。

要知道我们在MH为proposal distribution定义了一个非常非常好的性质：它是什么分布都可以。所以我们可以尽情的选用我们会采样的分布。但是Gibbs采样为了效率牺牲了这一点，它本质上是在从联合分布的条件分布上给整个分布采样，所以前提就是条件分布比较好采样。

编辑于 2023-06-12 17:56 · IP 属地山东

MCMC采样　贝叶斯统计　机器学习

发布一条带图评论吧

77 条评论　默认　最新

Aaron

你真是中文的吗……

2021-10-12　回复　24

而今听雨 作者

哈哈哈哈，现在学中文也已经卷起来了哦🥰🥰🥰

2021-10-12　回复　6