Mathematical Modeling
(for psychology students...)

Approximate inference (1)—MCMC1

张洳源

2023/03/28

上海交通大学心理与行为科学研究院

上海交通大学医学院附属精神卫生中心

MCMC in psychology

RESEARCH

RESEARCH ARTICLE SUMMARY

NEUROSCIENCE

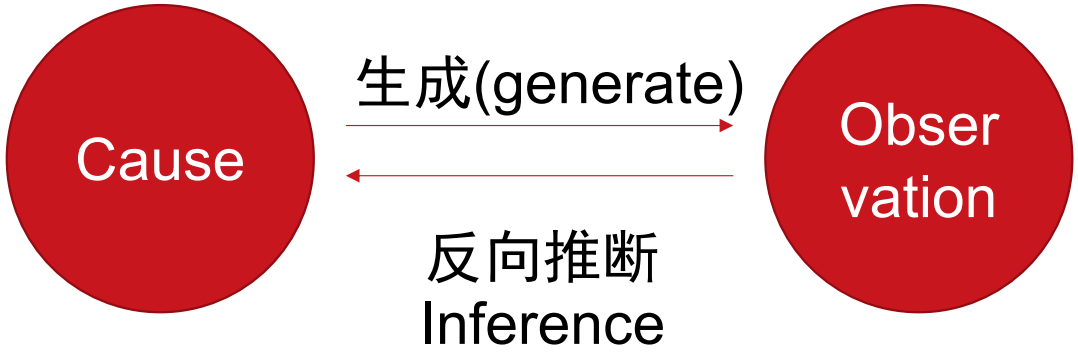
Computational and neurobiological foundations of leadership decisions

Micah G. Edelson*, Rafael Polania, Christian C. Ruff, Ernst Fehr*, Todd A. Hare*

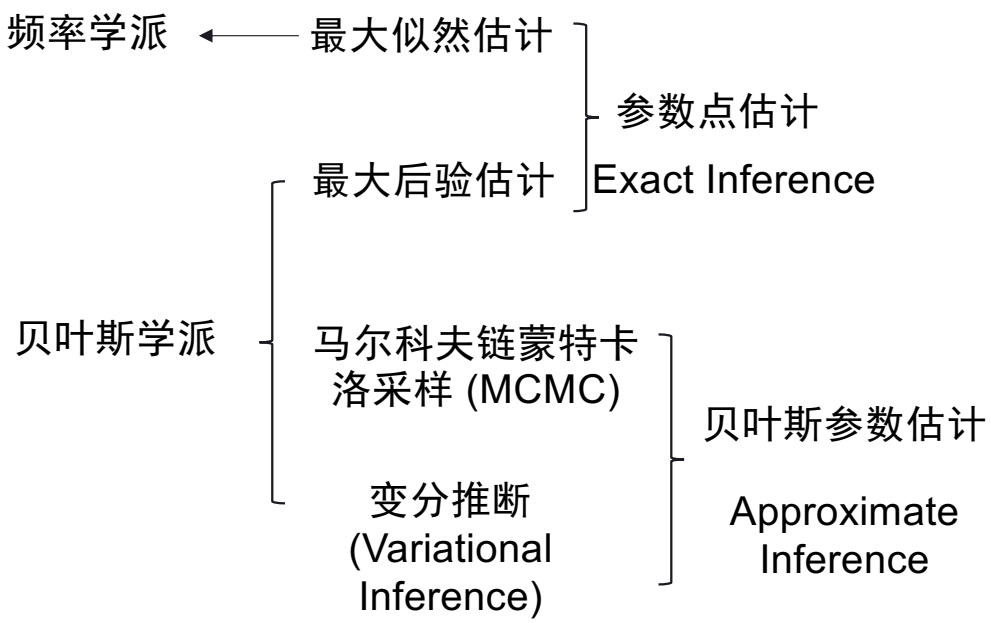
For latent variables at the highest level of the hierarchy (hyper-group parameters), we assumed flat uninformed priors (i.e., uniform distributions). Posterior inference of the parameters in the hierarchical Bayesian models was performed via the Gibbs sampler using the Markov Chain Monte Carlo (MCMC) technique implemented in JAGS (*70*, *71*). A total of 50,000 samples were drawn from an initial *burn-in* sequence, and subsequently a total of 50,000 new samples were drawn using three chains (each chain was derived based on a different random number generator engine, using a different seed). We applied a *thinning* of 50 to this sample, resulting in a final set of 1,000 samples for each parameter. This thinning assured that the final samples were not auto-correlated for all of the latent variables of interest investigated in the study. We conducted Gelman-Rubin tests for each parameter to confirm convergence of the chains. All latent variables in our Bayesian

Generative Process and Reverse Inference

- Generative process (由原因生成观察数据)
- Reverse Inference (由观察数据反推原因)



频率学派 vs. 贝叶斯学派



马尔科夫链蒙特卡洛采样 (MCMC)和变分推断(Variational Inference)后面的内容会说到

Bayesian Theory

后验分布　　似然函数　　先验分布

$$p(\theta|d) = \frac{p(d|\theta) * p(\theta)}{p(d)}$$

$p(d)$ ⟶ 归一化因子

有时候写成

$$p(\theta|d) \propto p(d|\theta) * p(\theta)$$

认知心理学中概率模型的本质就是求解后验概率分布$p(\theta|d)$

Background

- 按照参数是精确估计还是近似估计来划分Inference的方法
  - Exact Inference
    - 精确的找到参数值
    - e.g., Maximum Likelihood Estimation (MLE)
    - e.g., Maximum a Posterior (MAP)
  - Approximate Inference
    - 对参数取值的一个近似估计
    - e.g., Markov Chain Mento Carlo Sampling
    - e.g., Variational Inference

Background

- 按照参数估计是针对likelihood function还是posterior distribution的方法
  - 优化likelihood
    - Maximum Likelihood Estimation (MLE), exact inference, point estimate
  - 优化posterior
    - Maximum A Posterior (MAP), exact inference, point estimate
    - Markov Chain Mento Carlo methods (MCMC)
      - Approximate inference
      - Obtain a full posterior distribution
    - Variational Bayesian methods
      - Approximate inference
      - Obtain a full posterior distribution (maybe in a different format)

Background

- 在认知神经科学中，最常用的是MLE, MAP和MCMC

- 相较MLE/MAP, MCMC的优势
  - MLE基于似然函数，很多时候似然函数里面包含多个隐变量和多重积分，无法得到解析形式，无法优化求解。MCMC不需要解析求解任何积分。
  - MLE只是关于参数的点估计。MCMC得到整个后验概率分布，可以得到参数估计的uncertainty
  - MLE很容易会收敛到local minima。MCMC一般不会。

- 相较MLE/MAP, MCMC的劣势
  - 收敛速度慢
  - 得到后验概率之后，不方便做统计以及其他分析。(如何根据sampling近似的后验概率做各种分析是一个热门话题！)

Background

- 对于非machine learning专业的人(我说的就是心理学学生。。)来说，理解MCMC是个非常大的挑战
- 理解MCMC，对于理解probabilistic Bayesian modeling来说是个里程碑
- 理解MCMC的三个层次
  - 完全理解所有数学推导和应用代码实现(excellent !)
  - 部分理解背后数学推导但是可以实现采样算法(good !)
  - 部分或者不理解背后数学推导但是可以借助例如stan软件实现采样算法(still good !)
- 认知神经科学的人基本需要做到2和3就可以了

Background

- 常见感受(我说的就是心理学学生。。)
  - 啊，数学好复杂，说人话
  - 好吧，好不容易看懂数学，和我(心理学)有什么关系。。。
  - 似乎有点关系哦，那代码怎么整。。。

- 完全理解数学推导
  - 博客资料: https://www.cnblogs.com/pinard/p/6625739.html
  - B站视频: https://www.bilibili.com/video/av32430563/

Outlines

- What will we learn today?
  - Markov Chain
  - Monte Carlo Methods
  - Sampling
  - Markov Chain Monte Carlo (MCMC)的推导
  - 手写 Metropolis-Hasting Algorithm
  - MCMC的PYMC3的实现
  - MCMC在心理学的应用

Outlines

马尔科夫链 蒙特卡洛 采样

Markov Chain Monte Carlo Sampling

Outlines

马尔科夫链 蒙特卡洛 采样

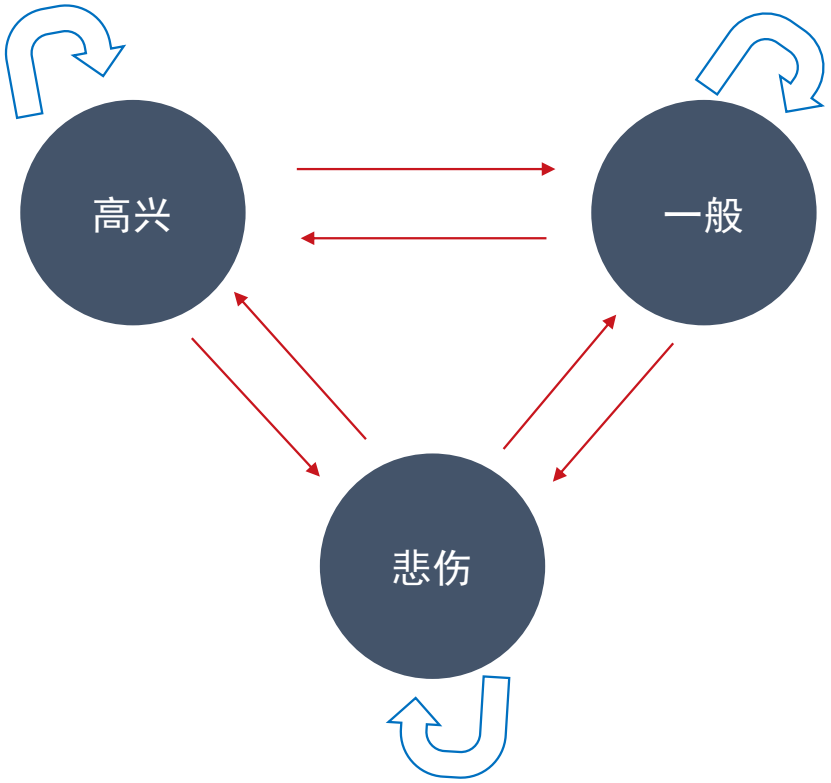
Markov Chain Monte Carlo Sampling

Markov Chain

- Markov Chain
  - 马尔科夫链的定义
  - 离散分布和状态转移矩阵
  - 状态转移矩阵和马尔科夫稳态分布
  - 马尔科夫链的细致平稳条件
  - 马尔科夫采样

Markov Chain

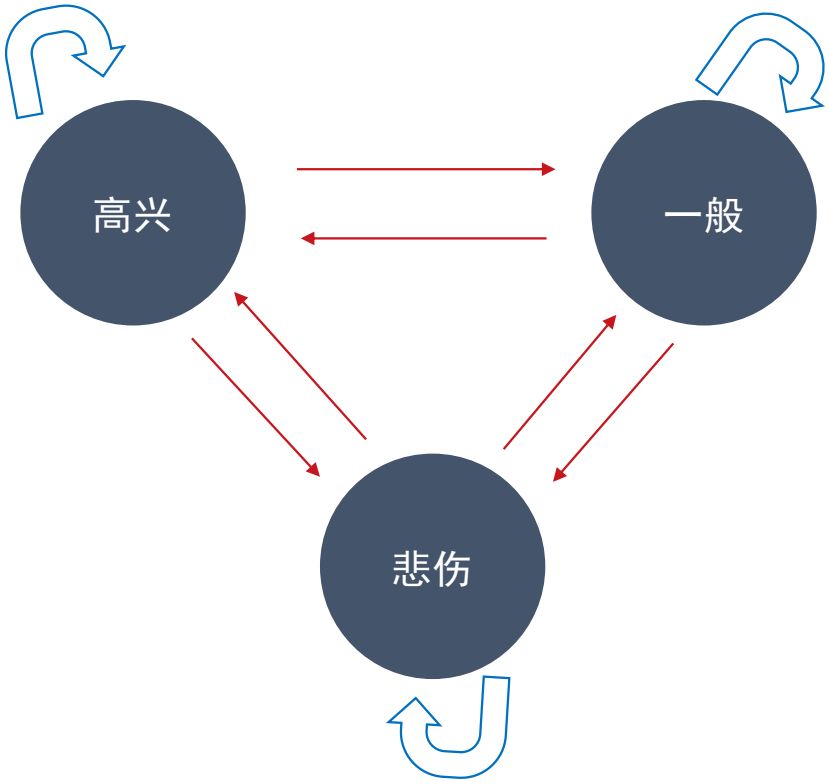
- 马尔科夫链
  - 下一个状态S_{t+1}只和当前状态S_t有关，和再之前的状态(e.g., S_{t-1})无关



假定人的情绪有三种状态，然后可以在三种状态分别转移，然后转移矩阵是P

假定人在t0时刻的初始情绪可能性分布是$p_0 = [0.5, 0.3, 0.2]$, 我们来一步一步计算人的情绪会如何演化

离散分布和状态转移矩阵



假定人的情绪有三种状态，然后可以在三种状态分别转移，然后转移矩阵是P

假定人在t0时刻的初始情绪分布是$e_0 = [0.5, 0.3, 0.2]$, 我们来一步一步计算人的情绪会如何演化

Markov Chain

- 马尔科夫链的性质
  1. 对于一个初始的离散分布e_0，如果固定了一个转移矩阵P，那么多次迭代之后我们会发现收敛到一个最终的稳定的分布
  2. 无论初始状态分布如何，都会收敛到那个稳态分布

- 马尔科夫链
  - 离散分布，$p(x)$是概率质量函数，$Q(x_2 \mid x_1)$是状态转移矩阵
  - 连续分布，$p(x)$是概率密度函数，$Q(x_2 \mid x_1)$是状态转移函数

Markov Chain

见 1markovchain.ipynb

Sampling in Markov Chain

- 什么是sampling (采样)?
  - 采出来的样本对应的是概率分布的x轴
  - 才出来很多样本之后，画histogram的图，应该近似概率分布

- 问题：
  - 如何从一个离散分布中采样？例如 [0.5, 0.3, 0.2]
  - 如果不知道概率质量函数？如何从中采样？？
    - 利用Markov的性质，如果我们，但是只要找到其对应的转移矩阵，也可以采样
    - 我们只要随意指定一个初始状态，然后沿着状态转移矩阵来采样转换状态，最终会达到稳态

- 马尔科夫链的细致平稳条件: 满足一下条件的分布p就是转移矩阵Q的细致平稳分布

$$p(x_1) * Q(x_2 \mid x_1) = p(x_2) * Q(x_1 \mid x_2)$$

Markov Chain

见 2markovchain.ipynb

Outlines

马尔科夫链 蒙特卡洛 采样

Markov Chain Monte Carlo Sampling

Monte Carlo

- Monte Carlo是一种思想或者一类方法

- 为什么我们要用采样的方法?
  - 因为我们经常要在一个分布上做积分。。
  - e.g., 求期望，方差，做其他积分
  - 但是一旦分布的概率函数比较复杂，不能做积分，就很难办。。

求期望

- 给定一个分布$p(x)$, 求其期望

$$E(x) = \sum_{i}^{n} x * p(x)$$

离散分布

$$E(x) = \int x * p(x) dx$$

连续分布

如果$p(x)$特别复杂，那么积分可能很难求, 例如

$$p(x) = \frac{1}{1.2113}(0.3 * e^{-(x-0.3)^2} + 0.7 * e^{\frac{(x-2)^2}{0.3}})$$

求积分

- 给定一个分布$p(x)$, 求另外一个函数$f(x)$在其的积分

$$E(x) = \sum_{i}^{n} f(x_i) * p(x_i)$$ 离散分布

$$E(x) = \int f(x) * p(x) dx$$ 连续分布

如果$p(x)$或者$f(x)$特别复杂，那么积分可能还是很难求

$$p(x) = \frac{1}{1.2113}(0.3 * e^{-(x-0.3)^2} + 0.7 * e^{-\frac{(x-2)^2}{0.3}})$$

$$f(x) = e^{-(\log(x-1))^2} + x^3 + x^2$$

求期望

- 假如这个分布$p(x)$本身很复杂，或者待积分函数$f(x)$很复杂，可以用采样近似的方法求解积分

我们从$p(x)$里面采集一系列的样本$x_1, x_2, x_3, x_4, \dots x_i$

例如，求期望

$$E(x) = \sum_i^n x_i * p(x_i) \approx \frac{1}{n}\sum_i^n x_i$$

离散分布

$$E(x) = \int x * p(x)dx \approx \frac{1}{n}\sum_i^n x_i$$

连续分布

Monte Carlo

- 一句话总结：就是避开利用概率公式进行精确的解析的方式进行积分求解，可以通过采样的方式简单暴力解决一些问题

- 注意，这里我们已知概率函数$p(x)$, 只是这个$p(x)$可能比较复杂，直接积分比较难求

- 例子：模拟股票走势

应用1: 模拟股票走势

- 假定某个股票目前的股价是S_t, 预测时间Δt之后的股价S_{t+1}， 金融工程中有公式

$$s_{t+1} = s_t + \hat{u} * s_t * \Delta t + \sigma * s_t * \epsilon * \sqrt{\Delta t}$$

其中,
$\hat{u}$: 股票收益率的期望值,
σ : 波动率
Δt : 假定单位是一个交易日
ϵ: 满足N(0,1)，决定了是一个随机过程

我们可以模拟一下，假定初始股价s_0=10, 一年之后(244个交易日)的股价分布

Markov Chain

见 3montecarlo_stock.ipynb

Outlines

马尔科夫链 蒙特卡洛 采样

Markov Chain Monte Carlo Sampling

Sampling

- 什么是sampling (采样)?
  - 采出来的样本对应的是概率分布的x轴
  - 采出来很多样本之后，画histogram的图，应该近似概率分布

- Sampling (采样)
  - 逆变换采样
  - 接受拒接采样
  - Metropolis采样

Sampling

- 如何从一个分布中采样的到一个样本？
  - 我们在例如python或者matlab里面可以很轻松的借助内置函数来从一些常见分布中采样
  - 关键问题: 假如不允许使用这些函数，如何采样？

```
1  from scipy.stats import norm, gamma
2  a = norm.rvs()
3  print(a)
4  b = gamma.rvs(1)
5  print(b)
```

✓ 0.0s

```
1.1777041140046502
0.2294693345513522
```

- 例如，给定一个均匀分布[0,1]的样本，如何利用该均匀分布从一个高斯分布中采样?

Sampling

- 采样的三种方法
  - 逆变换采样 (inverse transform sampling)
  - 接受拒绝采样(accept-rejection sampling)
  - Metropolis采样

Sampling

- 问题: 只给定均匀分布[0,1]的样本，如何利用该均匀分布从一个高斯分布中采样?

- 方法1：逆变换采样 (inverse transform sampling)
- 算法:
  1. 从均匀分布[0,1]采样得到 x
  2. 从高斯分布的积累分布函数CDF的反函数$y = f^{-1}CDF(x)$, y就是需要获得的样本

- 几乎所有的已知分布都可以用这种方法采样(matlab和python内部就是这么做的)

- 可是如果一个复杂的分布，不知道其积累分布函数CDF的反函数，如何采样?

Sampling

- 问题: 给定一个均匀分布[0,1]，如何从以下复杂分布中采样?

$$p(x) = \frac{1}{1.2113}\left(0.3 * e^{-(x-0.3)^2} + 0.7 * e^{\frac{-(x-2)^2}{0.3}}\right)$$

- 方法2： 接受拒绝采样(Accept-Rejection sampling)

接受拒绝采样

- 采样前, 先准备
  1. 先找到一个建议分布(proposed distribution)G的概率密度函数是g(x)，可以是任意分布(e.g., 高斯分布)
  2. 再找到一个常数C, 是的对于任意的x, 都有$\mathrm{C} * \mathrm{g}(\mathrm{x}) \geq p(x)$

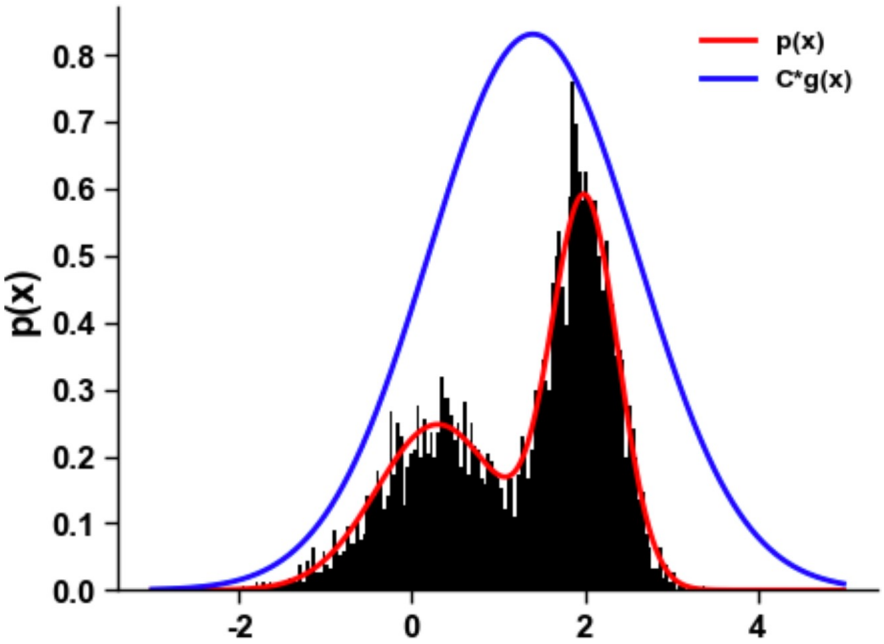
- 算法:
  1. 从建议分布G中进行采样，获取采样样本y
  2. 从[0, 1]的均匀分布中进行采样，采取一个数值u
  3. 如果 $u \leq \frac{p(y)}{C*g(y)}$,那么我们就接受这个样本y，保存下来; 如果不满足，就拒绝并丢弃这个采样值，重来

Sampling

- 接受拒绝采样的直观理解：
  - 在每一个x的点上，都有 $p(y) \leq C * g(y)$，即$0 < \frac{f(y)}{C*g(y)} \leq 1$
  - 从图上看，就是蓝线和红线的比值。

(推导证明见资料)

补充材料1_接受拒绝采样的证明



Sampling

接受拒绝采样的例子
4acceptrejectsampling.ipynb

Sampling

- 可是接受拒绝采样需要找到一个准确的常数c, 如果这个常数c不好直接得到怎么办?

- 方法3：Metropolis-Hasting采样
- 算法:

1. 从任意一个x_0开始
2. 根据当前的样本x_t构建一个提议分布$g(x|x_t)$ (e.g., $g(x|x_t)$可以是一个高斯分布)，然后采样得到x^*
3. 然后计算接受率 $\alpha = \min\left[1, \frac{p(x^*)*g(x_t|x^*)}{p(x_t)*g(x^*|x_t))}\right]$
4. 从[0,1]采样一个值u,
   - 如果 u< α, 接受该样本$x_{t+1} = x^*$
   - 否则，继续维持原来的样本$x_{t+1} = x_t$
5. 重复2-4步直到达到预订的样本数量

后面会解释这个算法为什么成立。。。

Outlines

- What will we learn today?
  - Markov Chain
  - Monte Carlo Methods
  - Sampling
  - Markov Chain Monte Carlo (MCMC)的推导
  - 手写 Metropolitan-Hasting Algorithm
  - MCMC的PYMC3的实现
  - Bayesian modeling in cognitive neuroscience

MCMC的证明

- Markov Chain Monte Carlo Sampling 关键词的含义
  - Markov Chain: 我们采样是沿着一条Markov链在进行状态转移
  - Monte Carlo Sampling: 我们是用采样的方法来逼近某个分布

MCMC的证明

- 把前面讲的两部分结合起来
- 回到Metropolis-Hasting sampling

1. 从任意一个x_0开始
2. 根据当前的样本x_t构建一个提议分布$g(x|x_t)$，然后采样得到x^*
3. 然后计算接受率 $\alpha = \min\left[1, \frac{p(x^*)*g(x_t|x^*)}{p(x_t)*g(x^*|x_t))}\right]$
4. 从[0,1]采样一个值u,
   - 如果 u< α, 接受该样本$x_{t+1} = x^*$
   - 否则，继续维持原来的样本$x_{t+1} = x_t$
5. 重复2-4步直到结束

这里有两个问题
1. 这里的g是由我们自己随意定的，凭什么随意定一个g都可以能从$p(x)$进行采样？
2. 为什么接受率定成这个形式了就可以对$p(x)$进行采样？

MCMC的证明

前面知道, 根据Markov采样定理, 我们要$p(x)$从中采样, 但是$p(x)$比较复杂。 一种方法是借助与该分布对应的马尔科夫链状态转移函数来进行采样。

那么怎么找到该分布的马尔科夫链状态转移函数呢? 我们随便假定一个转移函数Q显然是不行的。即不满足细致平稳条件，即

$$p(x_i) * Q(x_j|x_i) \neq p(x_j) * Q(x_i|x_j) \tag{1}$$

但是我们可以对上面的式子做一个改造, 使细致平稳分布条件成立。方法是引入一个$a(x_j|x_i)$, 使得上面的式子可以取等号, 即

$$p(x_i) * Q(x_j|x_i) * a(x_j|x_i) = p(x_j) * Q(x_i|x_j) * a(x_i|x_j) \tag{2}$$

问题是什么样子的$a(x_j|x_i)$可以满足该要求呢? 其实很简单,只要满足下面条件:

$$a(x_j|x_i) = min(1, \frac{p(x_j) * Q(x_i|x_j)}{p(x_i) * Q(x_j|x_i)}) \tag{3}$$

把公式3带入公式2,

$$\begin{aligned} p(x_i) * Q(x_j|x_i) * a(x_j|x_i) &= p(x_i) * Q(x_j|x_i) * min(1, \frac{p(x_j) * Q(x_i|x_j)}{p(x_i) * Q(x_j|x_i)}) \\ &= min(p(x_i) * Q(x_j|x_i), p(x_j) * Q(x_i|x_j)) \\ &= p(x_j) * Q(x_i|x_j) * min(\frac{p(x_i) * Q(x_j|x_i)}{p(x_j) * Q(x_i|x_j)}, 1) \\ &= p(x_j) * Q(x_i|x_j) * a(x_i|x_j) \end{aligned}$$

可以发现，如果另一个新的转换函数$M(x_j|x_i) = Q(x_j|x_i) * a(x_j|x_i)$

那么上式就可以写成

$$p(x_i) * M(x_j|x_i) = p(x_j) * M(x_i|x_j) \tag{4}$$

我们发现就满足了细致平稳分布条件。

注意，这里的$Q(x_j|x_i)$可以是任意的转换函数(e.g., 高斯函数)，上面的式子恒成立。

补充材料2_MH算法证明

MCMC的证明

让我们回到最开始的目的

认知心理学中概率模型的本质就是求解后验概率分布$p(\theta|d)$

$$p(\theta|d) = \frac{p(d|\theta) * p(\theta)}{p(d)}$$

根据贝叶斯公式，我们带进去

$$\begin{aligned} a(x_j|x_i) &= min(1, \frac{p(x_j|data) * Q(x_i|x_j)}{p(x_i|data) * Q(x_j|x_i)}) \\ &= min(1, \frac{p(data|x_j) * p(x_j)/p(data) * Q(x_i|x_j)}{p(data|x_i) * p(x_i)/p(data) * Q(x_j|x_i)}) \end{aligned}$$

- 如果我们假定的是uniform的prior，那么$p(x_i) = p(x_j)$。
- 如果我们假定的是对称的提议分布(比如高斯分布，实际上前面说过任意提议分布都可以，一般都选择高斯)，那么$Q(x_i|x_j) = Q(x_j|x_i)$

那么上式就可以进一步简化成

$$a(x_j|x_i) = min(1, \frac{p(data|x_j)}{p(data|x_i)}) \tag{1}$$

可以看到，右边直接就是一个likelihood的比值

MCMC的证明

所以在真实参数估计的时候，在假定uniform prior的条件下，实际的算法是

1. 从任意一个x_0开始
2. 根据当前的样本x_t构建一个提议分布$g(x|x_t)$，然后采样得到x^*
3. 然后计算接受率 $\alpha = \min\left[1, \frac{p(\mathrm{d} \mid x^*)}{p(d \mid x_t))}\right]$
4. 从[0,1]采样一个值u,
   - 如果 u< α, 接受该样本$x_{t+1} = x^*$
   - 否则，继续维持原来的样本$x_{t+1} = x_t$
5. 重复2-4步直到结束

编程练习

见MHsampling的实例

5MHsampling.ipynb

小结

总体目标，求后验概率$p(\theta|data)$

怎么求？从后验概率$p(\theta|data)$分布采很多样本去近似

Markov chain → 任何一个分布，都有其对应的一个平稳细致转移函数(或矩阵) → 从分布的任意状态出发，沿着转移函数(或矩阵)进行转移，就可以对分布采样

Monte Carlo → 避免求积分的简单粗暴的近似思想

Sampling → 逆变换采样 (常见分布)

接受拒绝采样 (复杂分布，但需要常数C)

先采样再按比例接收的思想

任意一个转移函数Q，满足细致平稳条件的转移矩阵

$$M = \mathrm{Q} * \alpha = Q * \min\left[1, \frac{p(\mathrm{d} \mid x^*)}{p(d \mid x_t))}\right]$$

Metropolis-Hasting算法

1. 从任意一个x_0开始
2. 根据当前的样本x_t构建一个提议分布$g(x|x_t)$，然后采样得到x^*
3. 然后计算接受率 $\alpha = \min\left[1, \frac{p(\mathrm{d} \mid x^*)}{p(d \mid x_t))}\right]$
4. 从[0,1]采样一个值u,
   - 如果 u< α, 接受该样本$x_{t+1} = x^*$
   - 否则，继续维持原来的样本$x_{t+1} = x_t$
5. 重复2-4步直到结束