

MCMC Algorithm and Its Application

Mei Sun

Shanghai University of Finance and Economic, Shanghai
Email: 18321805716@163.com

Received: Nov. 24th, 2018; accepted: Dec. 20th, 2018; published: Dec. 27th, 2018

Abstract

This paper mainly introduces the Markov Chain Monte Carlo Method (MCMC), which is mainly the Metropolis method, Hasting method and Gibbs method. The basic steps of these algorithms are introduced. At the same time, the convergence of Markov chain is used, the error of the algorithm is discussed, and the algorithm is improved. Finally, some simple applications of the limit properties are given. This paper is divided into six parts: The first part introduces the nature of the Markov chain. The second part introduces the Metropolis-Hasting (M-H) algorithm and promotion, and introduces two special MCMC algorithms, namely Metropolis algorithm and Hasting algorithm and algorithm implementation process. The third part introduces the improvement of the proposed probability distribution, which reduces the error and improves the convergence. The fourth part introduces the Gibbs algorithm and the Bayesian model. The fifth part gives examples and applications, and finds a suitable implementation method among many methods. The last part, the sixth part, is summarized.

Keywords

MCMC, Metropolis Algorithm, Hasting Algorithm, Gibbs Algorithm, Convergence

MCMC算法及其应用

孙 玫

上海财经大学, 上海
Email: 18321805716@163.com

收稿日期: 2018年11月24日; 录用日期: 2018年12月20日; 发布日期: 2018年12月27日

摘 要

本文主要介绍了马尔可夫链蒙特卡罗方法(MCMC), 主要是Metropolis方法、Hasting方法和Gibbs方法。

介绍了这些算法的基本步骤，同时利用马尔可夫链的收敛性，讨论了算法的误差，并对算法进行改进。最后，给出了极限性质一些简单的应用。本文主要分为六部分：第一部分介绍了马尔可夫链的性质。第二部分主要介绍了Metropolis-Hasting (M-H)算法及推广，介绍了两种特殊的MCMC算法，即Metropolis算法和Hasting算法以及算法的实现过程。第三部分介绍了建议概率分布的改进，降低了误差，提高了收敛性。第四部介绍了Gibbs算法以及贝叶斯模型。第五部分给出了实例与应用，同时在众多方法中找到一个合适的实施方法。最后一部分，也就是第六部分进行了总结。

关键词

MCMC, Metropolis算法, Hasting算法, Gibbs算法, 收敛性

Copyright © 2018 by author and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 前言

在统计学中，经常会遇到高维积分的计算问题，而使用传统的数值方法往往很难解决实际问题。随着计算技术的提高，随机模拟的方法为解决这些高维积分问题提供了一个很好的思路。

马尔可夫链蒙特卡罗(Markov Chain Monte Carlo, MCMC)方法的理论框架来源于Metropolis (1953)和Hasting (1970) [1]，首先在 $X = (X_0, X_1, \dots, X_t, \dots)$ 的状态空间中模拟出一条马尔可夫链，使得该马尔可夫链的稳定分布就是目标分布 $f(x)$ 。此时在这个状态空间游走时，每一个状态 X_t 停留的时间正比于目标分布 $f(x)$ 。为了有效的得到目标分布，此时引入一个容易抽样的建议概率密度函数 $g(\cdot|x_t)$ ，在每一次的抽样过程中，先从 $g(\cdot|x_t)$ 抽样得到一个候选样本 Y_t ，接着再按照后面提到的原则决定是否接受该样本。显然这样产生的随机序列是一个离散时间的时齐的不可约遍历 Markov 链。

这一算法已经在诸多领域进行推广以及改进，例如：应用统计学、金融学、计量经济学、计算物理学等领域。

在过去的几十年里，MCMC 方法将复杂的多元问题简单化已经引起了许多人的注意。首先，在 1953 年，Metropolis, Rosenbluth, Teller [2]在“MANIAC”计算机上实现物质状态方程的蒙特卡罗方法模拟，1970 年 Hasting [1]将建议概率分布从对称分布变为非对称形式，这个算法称为 Metropolis-Hasting (M-H)算法，也称为通用 Metropolis 算法。接下来就是 Tanner、Rubinstein [3]、Wong (1987)、Gelfand 和 Smith (1990)在细节处进行了优化。S.Chib 和 E.Greenberg (1995) [4]对 M-H 算法的理解以及 L.Tierney (1994) [5]马尔可夫链在后验分布中的应用。在 1984 年 Gelman 提出了在研究图像恢复问题时提出了 Gibbs 算法。Liu (1996)提出了混合 Gibbs 算法以及 Swendsen、Wang (1987)在原来的基础上发生了演变，形成了聚类算法。2003 年 C.Andrieu [6]将 MCMC 算法应用于了机器语言，2011 年，S.Brooks [7]对 MCMC 的算法理论原理做出了更为细致的解读。

1.1. 马尔可夫链的基本概念

马尔可夫链理论是研究马尔可夫链蒙特卡罗方法的基础，为此先给出马尔可夫链的定义及基本性质。

1.1.1. 马尔可夫链的定义

如果对于任意时刻 t ，条件概率满足

$$P(X_{t+1} = x_{t+1} | X_t = x_t, \dots, X_1 = x_1, X_0 = x_0) = P(X_{t+1} = x_{t+1} | X_t = x_t)$$

则称此性质为马尔可夫性，若右式与 t 无关，则称 X 为时齐的。

引入记号：

$p(x, y)$ 为过程从状态 x 转移到状态 y 的概率，称为一步转移概率； $p^n(x, y)$ 为过程从状态 x 经过 n 步转移后到达状态 y 的概率，称为 n 步转移概率。

给定状态序列 $x_0, x_1, \dots, x_t, \dots$ ，状态空间 S ，为了简便书写，记 $X_{t+1} = y$ ， $X_t = x$ ，则转移概率为 $P(x, y) = P(x_{t+1} = y | x_t = x)$ ，转移概率 $p(x, y)$ 组成的矩阵

$$P = \begin{bmatrix} p(0,0) & p(0,1) & \cdots \\ p(1,0) & p(1,0) & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

其中， $p(x, y) \geq 0 (x, y \in S)$ ， $\sum_{y \in S} p(x, y) = 1$ ，称此矩阵为一步转移概率矩阵。

文中讨论的马尔可夫链都是时齐的有限马尔可夫链。

令 $\tau_y = \inf \{n \geq 1 : X_n = y\}$ ，则 τ_y 是一个停时，称之为状态 y 的首中时，记

$$f^n(x, y) \triangleq P^x(\tau_y = n), n \geq 1$$

$$f(x, y) \triangleq \sum_{n \geq 1} f^n(x, y) = P^x(\tau_y < \infty)$$

1.1.2. 马尔可夫链的性质

1) 常返性。如果从状态 x 出发以概率1回到状态 x ，即 $f(x, x) = 1$ ，称状态 x 是常返的，否则称状态 x 时暂留的。如果一个马尔可夫链所有状态都是常返的，则称这个马尔可夫链是常返的。记 $\mu_y = E^y[\tau_y]$ 为 y 的平均返回时间，如果平均返回时间 $\mu_y < \infty$ ，其中 $\mu_x = E^x[\tau_x] = \sum_{n \geq 1} n f^n(x, x)$ 。

则状态 x 是正回返状态，如果平均回返时间 $\mu_x = \infty$ ，则状态 x 是零回返状态。

若所有状态是暂留的，则称这个马尔可夫链是暂留的。状态 x 是常返的等价于 $P^x(\limsup\{X_n = x\}) = 1$ 也等价于 $\sum_{n \geq 1} p^n(x, x) = \infty$ 。状态 x 是暂留的等价于 $P^x(\limsup\{X_n = x\}) = 0$ 也等价于 $\sum_{n \geq 1} p^n(x, x) < \infty$ 。

如果 $x, y \in E$ 是互达的，则有 $x, y \in E$ 都是常返的， $P^x(\limsup\{X_n = y\}) = 1$ 且 $\sum_{n \geq 1} p^n(x, y) = \infty$ ；否则 x, y 都是暂留的，即 $P^x(\limsup\{X_n = y\}) = 0$ 且 $\sum_{n \geq 1} p^n(x, y) < \infty$ 。

2) 不可约性。如果所有状态都存在 n ，使得 n 步转移概率 $p^n(x, y) > 0$ ，等价于每一个状态都可以经过有限步到达其他状态，也就是每个状态都是联通的，称这样的状态为不可约状态。有限状态的马尔可夫链是不可约的，则这个马尔可夫链是正回返的。

3) 周期性。 n 步转移概率 $p^n(x, y)$ ，状态 x 的周期数为 $d_x = \text{GCD}\{n \geq 1 : p^n(x, x) > 0\}$ 。

上式中 $\text{GCD}\{\cdot\}$ 表示满足条件 $p^n(x, x) > 0$ 且 $n \geq 1$ 的 n 的集合的最大公约数。若状态 x 的周期数 $d_x = 1$ ，则称状态 x 是非周期状态；若状态 x 的周期数 $d_x > 1$ ，则称状态 x 是周期状态。

4) 遍历性。如果状态 i 是正常返、不可约、非周期的，则称该状态 i 是遍历的。所有状态都是遍历的马尔可夫链，称为遍历的马尔可夫链。

5) 平稳分布。 E 上一个概率分布 $(\pi(y) : y \in E)$ ，如果满足 $\Pi P = \Pi$ ，即 $\sum_{x \in E} \pi(x) p(x, y) = \pi(y), \forall y \in E$ ，则称 Π 为转移矩阵 P 的平稳分布，显然有 $\pi(x) p(x, y) = \pi(y) p(y, x)$ ，也就是 π 是可逆的。

1.2. 马尔可夫链蒙特卡罗方法原理

蒙特卡罗方法是一种随机模拟的方法，也就是求解随机事件发生的概率或者随机变量的期望时，通过设计一种抽样方法，得到某一个特定事件发生的频率，然后使用这个频率来表示相应事件发生的概率。

但是随着维数的增加, 采样所需的样本量(或者说接受的概率)随维数增加而指数增长, 导致计算量太大, 速度太慢, 为了克服这一问题, 也就应运而生了马尔可夫链蒙特卡罗(MCMC)方法。MCMC算法通过构造Markov链的极限平稳分布来模拟计算积分, 转化成从已知概率分布抽样, 也就是先选择一个建议概率分布, 从建议概率分布抽样产生一个候选样本值, 则可以建立可操作的状态转移规则, 用接受概率判断状态是否转移, 重复这个过程, 将产生一条马尔可夫链。其中, Markov链满足大数定律:

假设 $\{X_n, n=0,1,\dots\}$ 为一平稳分布 π 的遍历的Markov链, 则 X_n 依概率收敛到分布为 π 的随机变量 X , 且对任意函数 g , 满足 $E[g(X)]$ 存在, 当 $n \rightarrow \infty$ 时, 有 $\bar{g}_n = \frac{1}{n} \sum_{i=1}^n g(X_i) \xrightarrow{P} E_\pi[g(X)]$ 。

如果生成的马尔可夫链是遍历的, 也就是它满足正常返性、不可约性和非周期性, 经过足够多的状态转移后, 马尔可夫链的转移概率会趋于一个平稳分布, 即事先给定的目标分布, 马尔可夫链的各个状态就是随机变量的样本值, 这样也就实现目标分布的随机抽样。

2. Metropolis-Hasting(M-H)算法及推广

Metropolis算法是MCMC的核心, 其实也就是改进的接受-拒绝抽样方法。接受-拒绝算法的具体步骤如下:

1) 首先选择一个容易抽样的分布作为建议概率分布, 记为 $g(x)$, 接着确定一个常数 C , 其中 $C > 1$, 使得在 x 的定义域上都有 $f(x) \leq Cg(x)$ 。

2) 生成服从建议概率密度函数的 $g(x)$ 的随机数 y 。

3) 接着生成一个服从均匀分布 $U(0,1)$ 的随机数 u 。

4) 计算接受概率 $h(x) = \frac{f(x)}{Cg(x)}$, 如果 $u < h(y)$, 则接受 y , 否则继续转到第(2)步。

5) 不断重复(2)~(4)就可以生成一组服从目标分布 $f(x)$ 的随机数序列。

类似于拒绝-接受抽样方法, 在当前状态 x_t 下产生下一个状态只需在原来拒绝-接受抽样方法下作一点改变得到Metropolis算法[8], 具体如下:

1) 首先构造建议概率分布 $g(\cdot|x_t)$, 并从建议概率分布 $g(\cdot|x_t)$, 其中 $g(y|x) = g(x|y)$, 产生一个随机数 y 作为下一个状态。

2) 接着按从均匀分布 $U(0,1)$ 的随机数 u , 如果 $u \leq \alpha(x,y)$, 则接受该建议的随机数 y , 即 $x_{t+1} = y$, 否则 $x_{t+1} = x_t$ 。

3) 重复上述过程, 我们就可以得到一组仅依赖于上一个随机数并且与前面随机数无关的随机序列。

4) 计算接受概率 $\alpha(x_t, y) = \frac{f(y)}{f(x_t)}$ 。

显然在这里, 建议概率密度函数 $g(x)$ 的作用主要是转移当前的状态, 也就是为每一个状态构造出一个邻域, 并且下一个状态就是这个邻域中的某一个数。

Metropolis算法的应用受到了建议概率分布的对称形式的限制, 因此, Hasting把建议概率分布从对称分布形式推广到一般情况, 形成了Hasting算法, 也称之为通用梅特罗波利斯算法, 此时接受概率变为

$$\alpha(x_t, y) = \min\left(1, \frac{f(x)g(x_t|y)}{f(y)g(y|x_t)}\right)。$$

接受概率的选择过程S. Chib和E. Greenberg (1995) [4]做了详细的解说。

2.1. 收敛性证明

要Metropolis算法收敛也就是证明马尔可夫链收敛到平稳分布, 即证明这里的马尔可夫链具有下列三

个性质。

- 1) $q(y|x)$ 不可约和非周期, 则 $A(x,y) = q(y|x)\alpha(y,x)$ 也不可约且非周期的。
- 2) $A(x,y)$ 可逆且满足细致平衡条件。
- 3) $A(x,y)$ 满足不变性。

Proof. 1) 由马尔可夫链性质知, 当马尔可夫链满足不可约性和非周期性, 那么 m 步转移概率 $q^m(x,y)$ 产生的任何 m 步跟踪路径为大于零的一个值, 由于 m 步转移函数 $A^m(x,y) = (q(y|x)\alpha(y,x))^m = (q(y|x))^m (\alpha(y,x))^m$ 具有正概率, 所以 $A(x,y) = q(y|x)\alpha(x,y)$ 也不可约且非周期的。

2) 因为

$$\begin{aligned} f(x)A(x,y) &= f(x)q(y|x)\alpha(y,x) \\ &= f(x)q(y|x)\min\{1, f(y)/f(x)\} \\ &= \min\{f(x)q(y|x), f(y)q(y|x)\} \\ &= f(y)q(y|x)\min\{f(x)/f(y), 1\} \\ &= f(y)q(y|x)\alpha(x,y) = f(y)A(y,x) \end{aligned}$$

所以 $f(x)A(x,y) = f(y)A(y,x)$, 即 $A(x,y)$ 满足细致平衡条件, 则 $A(x,y)$ 可逆。

3) 根据马尔可夫链的性质知, 满足细致平衡性也就确保了不变性, 所以就可以达到整个马尔可夫链总体的平衡[9]。下面要证明马尔可夫链满足不变性也就是满足

$$f(x)A(x,y) = f(x)$$

在离散和连续的情况下, 有

$$\sum_x f(x)A(x,y) = \sum_x f(y)A(y,x) = f(y)\sum_x A(y,x) = f(y)$$

同理,

$$\int f(x)A(x,y)dx = \int f(y)A(y,x)dx = f(y)\int A(y,x)dx = f(y),$$

所以对于不可约的概率分布 $f(y) > 0$, 不论这个初始分布怎样变化, 它最终都会趋于平稳分布 $f(y)$, 于是有, $\lim_{m \rightarrow \infty} A^m(x,y) = f(y)$ 。

由上可知, 马尔可夫链满足不变性。

2.2. 例子

建议概率密度为支撑集 $(x-a, x+a)$ [10] 上的建议概率分布 $q(y|x) = 1/2a$, 从密度函数 $f(x) \propto \exp\{-x^2/2\}$ 进行采样, 此时接受概率 $\alpha(x,y) = \min\{1, \exp\{(x^2 - y^2)/2\}\}$ 。接受概率随样本变化以及随 x 的变化见图1和图2。

由图1和图2可知, 随着样本数的增加, 接受概率的变化逐渐趋于稳定, 此时在 x 在区间 $[-1,1]$ 接受概率最高。同时算法的收敛性也验证了样本遵从已知概率分布, 也就是, 找到了合适的马尔可夫链, 使得这个马尔可夫链的稳定分布也就是目标分布, 并且, 每一个状态的停留时间也正比于目标概率(程序见附录 A)。

3. 建议概率分布改进

由于上述的 Metropolis-Hasting (M-H) 算法存在收敛速度慢、接受概率低、抽样效率不高、样本呈现

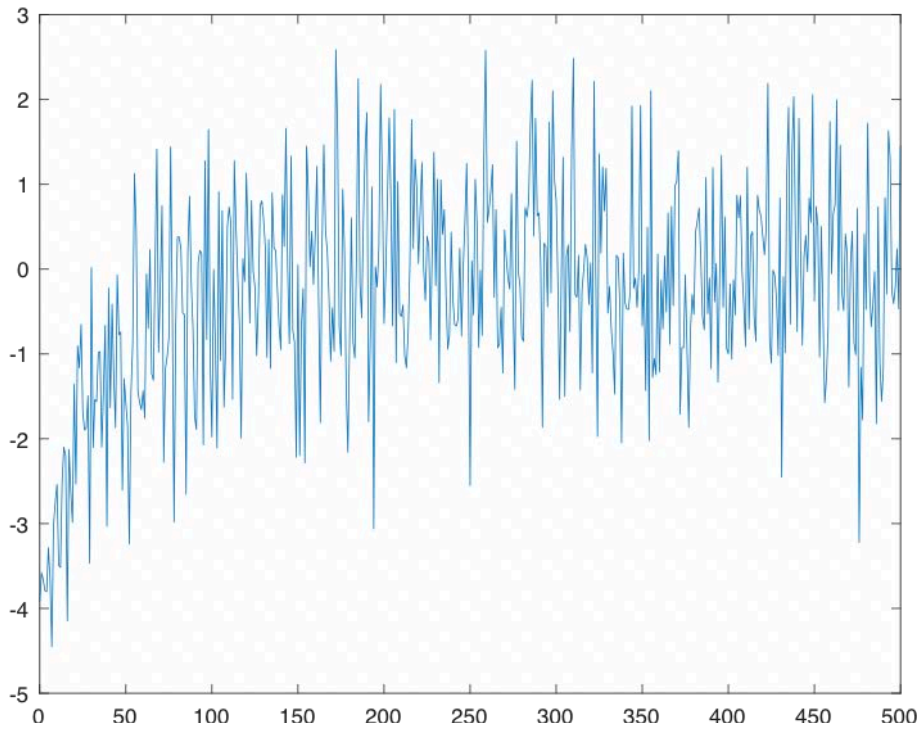


Figure 1. Acceptance probability varies with sample size
图 1. 接受概率随样本数变化

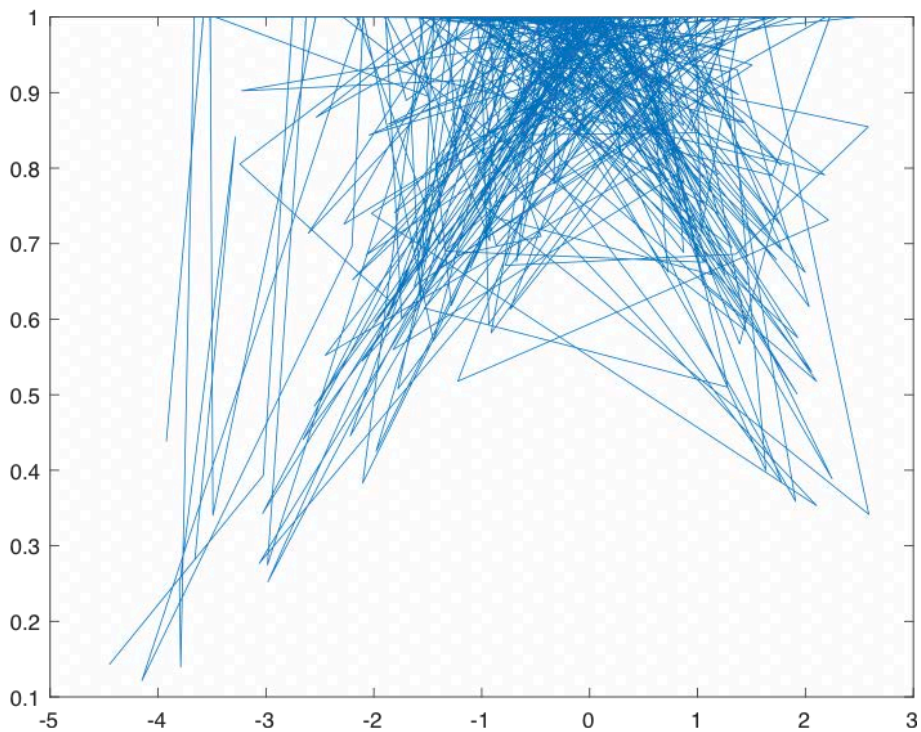


Figure 2. Acceptance probability varies with x
图 2. 接受概率随 x 变化

出一系列的相关性等问题，于是尝试改进建议概率分布，下面采用随机游走算法对建议概率分布进行改

进。此时随机游走算法的分布为对称分布，这里取建议概率分布为简单随机行走分布，从 $q(y|x_t)=1/2a$ 中抽样的样本值记为 Z_t ，接受概率为 $\alpha(X,Y)=\min\left(1,\frac{f(Y)}{f(X)}\right)$ 。

那么随机游走算法就变成了：

- 1) 首先从建议概率分布 $q(y|x_t)$ 抽样产生样本值 Z_t ，记候选样本值 $Y_t = X_t + Z_t$ 。
- 2) 接着按从均匀分布 $U(0,1)$ 的随机数 u ，如果 $u \leq \alpha(X_t, Y_t)$ ，则接受该建议的随机数 Y_t ，即 $X_{t+1} = Y_t$ ，否则 $X_{t+1} = X_t$ 。
- 3) 重复上述过程，得到一组仅依赖于上一个随机数并且与前面随机数无关的随机序列。

4) 计算接受概率 $\alpha(X,Y)=\min\left(1,\frac{f(Y_t)q(X_t|Y_t)}{f(X_t)q(Y_t|X_t)}\right)=\min\left(1,\frac{f(Y_t)}{f(X_t)}\right)$ 。

例子

同上述例子，概率密度函数 $f(x) \propto \exp\{-x^2/2\}$ ，由直接抽样的变换算法抽样，样本值记为 $X = (-2 \log(U_1))^{1/2} \cos(2\pi U_2)$ ，选取建议概率分布 $q(y|x_t)=1/2a$ 通过计算得到接受概率为 $\alpha(X,Y)=\min\left(1,\frac{f(Y)}{f(X)}\right)=\min\left(1,\exp\{-Y^2/2\}/\exp\{X^2/2\}\right)$ 。

结果详见图3和图4。

在这里，通过改进建议概率分布，采用了直接行走算法，这里选用最简单的随机行走均匀分布，也就是概率密度函数是完全已知的概率分布。下面，再给定初始值和随机行走因子。最终，得到了抽样效

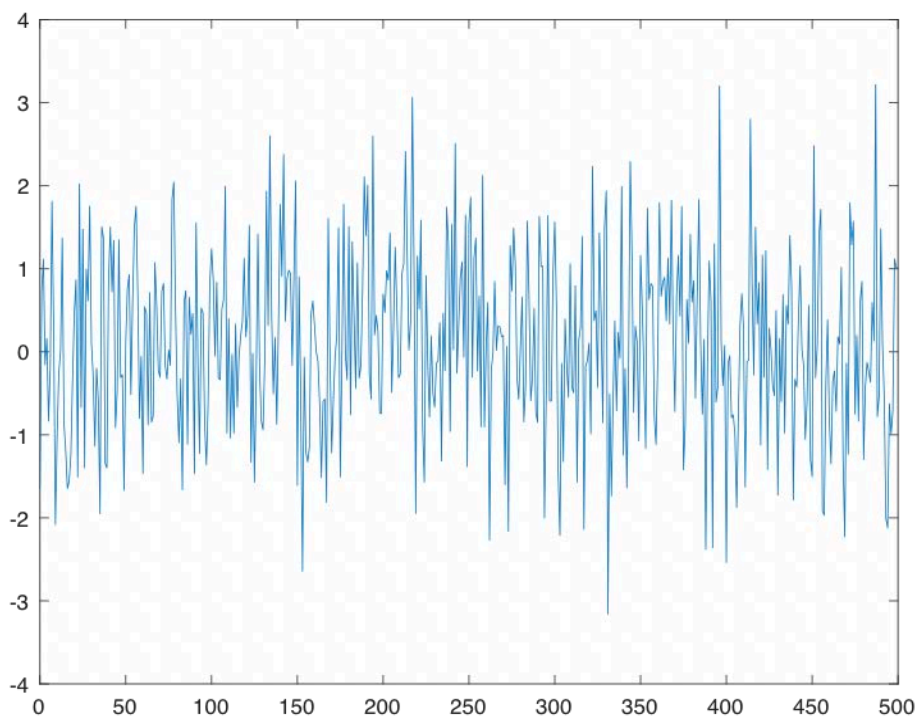


Figure 3. Acceptance probability varies with sample size
图3. 接受概率随样本数变化

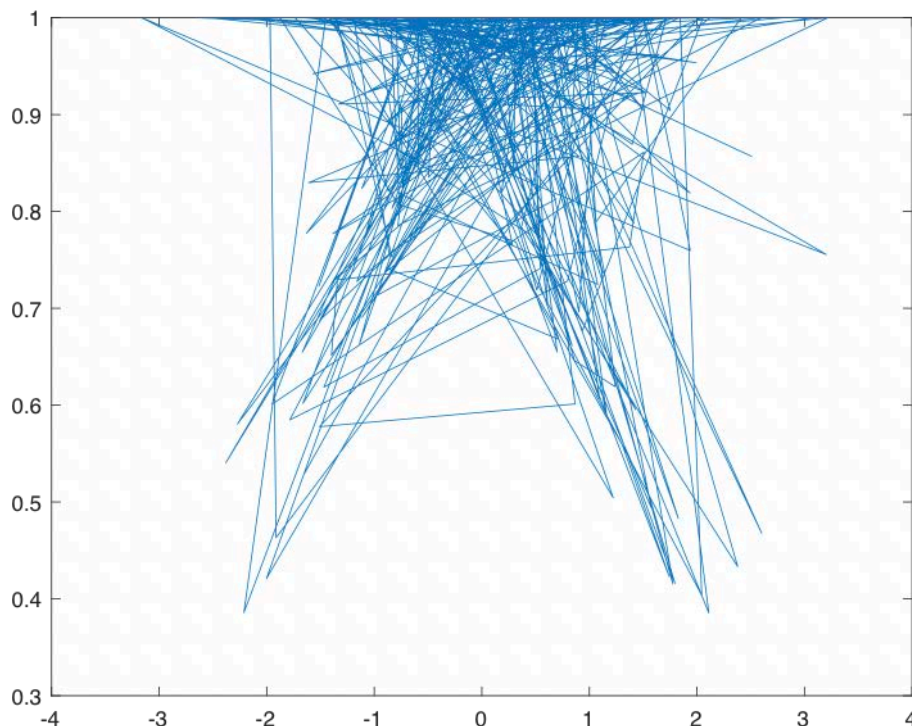


Figure 4. Acceptance probability varies with x
图 4. 接受概率随 x 变化

率更高的抽样算法(程序见附录B)。

4. Gibbs 方法

如果已知概率分布是全条件概率分布, 则马尔可夫链状态转移的规则就是建立在全条件概率分布抽样基础上的, 此时, 使用全条件的抽样方法, 使得抽样的结果满足随机性特点, 同时每一步都是用全条件概率分布来构建马尔可夫链的。

Gibbs方法是将高维问题[11]化为一维问题, 在对每个分量进行抽样的时候等价于对其他所有分量的条件分布进行抽样。所以下面将随机变量 X 分解成单随机变量 $X = (X_1, X_2, \dots, X_m)$, 随机变量 X 的概率分布 $f(x) = f(x_1, x_2, \dots, x_m)$, 单个随机变量 x_j 的全条件概率分布

$$f(x_j | x_{-j}) = f(x_j | x_1, x_2, \dots, x_{j-1}, x_{j+1}, \dots, x_m), j = 1, 2, \dots, m$$

其中, x_{-j} 表示 $x_1, x_2, \dots, x_{j-1}, x_{j+1}, \dots, x_m$ 。

Gibbs算法就是

- 1) 将已知的概率分布分解为多个单变量的全条件概率分布。
- 2) 通过对多个单变量全条件概率分布中产生候选点, 从 $f(x_j | x_{-j})$, 其中, $j = 1, 2, \dots, m$ 中产生候选点 $X_j(t)$, 更新 $x_j = X_j(t)$, 来实现对已知概率分布的抽样。

例子

假设 X_1, X_2, \dots, X_5 [12]是相互独立的服从指数分布的随机变量, 且均值分别为1, 2, 3, 4, 5, 然后用模拟方法估计 $P\left\{\prod_{i=1}^5 X_i > 120 \mid \sum_{i=1}^5 X_i = 15\right\}$, 使用Gibbs算法。随机选择 X , X , 参数分别 $\mu = \frac{1}{i}$, $\lambda = \frac{1}{j}$

(其中 $i > j$, 即 $\mu < \lambda$), 求在 $X_i + X_j = a$ 的情况下 X_i 的条件分布 $f_{X_i|X_i+X_j}(x|a) = c \exp\{-(\lambda - \mu)x\}$, $0 < x < a$ 。令初始状态 (x_1, x_2, \dots, x_5) 是和等于15的任意五个整数。从1, 2, 3, 4, 5中随机取出两个整数, 不妨假设是1和4, 则在给定其他值下 X_1 和 X_4 的条件分布为在其和等于 $15 - x_2 - x_3 - x_5$, 且参数为

$1 - \frac{1}{4} = \frac{3}{4}$ 的指数分布值, 下面令 x_1 等于这个值, 再重新设 x_5 的值使得 $\sum_{i=1}^5 X_i = 15$ 。重复上述过程就可以得到概率值的估计。

5. 实例与应用

设 $(\Omega, F, P, \{\mathcal{F}_t\})$ 是一个带 σ 域流的概率空间[13], P 为风险中性测度且服从 $dS_t = rS_t dt + \sigma S_t dX_t$, $\{\mathcal{F}_t, 0 \leq t \leq T\}$ 为 P 测度下产生的自然 σ 域流。设 S_t 为股票价格, $S_0 = S$, 无红利美式股票卖权协议价格为 K , 期限为 T , 无风险利率 r , σ 为股票波动率, Z_t 为标准布朗运动。令 $V(S(t), t)$ 表示股票在时刻 t 的价格, 所以我们可以构造一种投资组合来得到 $V(S(t), t)$, 利用 Δ 对冲技巧, 可以给出期权定价的数学模型, 形成投资组合 $\Pi = V - \Delta S$, 选取适当的 Δ 使得在 $(t, t + dt)$ 时段内, Π 无风险。设在时刻 t 形成的投资组合 Π , 并在 $(t, t + dt)$ 内, 不改变份额 Δ , 由于 Π 无风险, 在时刻 $t + dt$, 投资组合的回报 $\frac{\Pi_{t+dt} - \Pi_t}{\Pi_t} = r dt$,

即 $dtV_t - \Delta dS_t = \Pi_{t+dt} - \Pi_t = r\Pi_t dt = r(V_t - \Delta S_t) dt$ 。运用Itô公式, 并且代入上式, 得到Black-Scholes方程:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS \frac{\partial V}{\partial S} - rV = 0$$

对于欧式看涨期权, 也就是 $V(S(T), T) = (S(T) - K)^+$, 满足Black-Scholes方程的解为:

$V(s(t), t) = \exp\{-r(T-t)\} \int_{-\infty}^{\infty} (x - K)^+ f(s, s(t)) ds$ 其中, $f(s, s(t))$ 表示给定当前资产为 $s(t)$, 资产价格 S 在期权到期日 T 时的密度函数。因此股票价格波动是一个随机过程, 股票价格为

$S_t = S_{t-1} \exp\{(r - \sigma^2/2)\Delta t + \sigma\sqrt{\Delta t} X_t\}$ 。其中, $\{X_t\}$ 为相互独立的服从 $N(0, 1)$ 的随机变量。 c 表示该期权在零时刻的价格, 因此 $c = \exp\{-rT\} E[S_0 - K]^+$, 由于这个期权是一种路径依赖的期权, 计算包含一个多重积分, 积分的维数为 n , 所以求解过程很复杂, 因此采用马尔可夫链蒙特卡罗方法求解。

方法一: 随机向量 $X = (X_1, X_2, \dots, X_n)^T$, 其概率密度函数为

$$f(x) \propto H(x) \exp\{-x^T x / 2\},$$

式中, $H(x) = (S(t_n)K)^+ I(\min_{1 \leq i \leq n} S(t_i) \leq \beta)$ 。概率密度函数 $f(x)$ 归一化因子未知, 是不完全已知概率分布, 其抽样方法需要使用马尔可夫链蒙特卡罗方法。

此时, 建议概率分布为 $q(\lambda | d, X)$, 随机方向 d 服从 s 维单位超球体上均匀分布, 有

$$d = \left(\frac{Z_1}{\|Z\|}, \frac{Z_2}{\|Z\|}, \dots, \frac{Z_s}{\|Z\|} \right)^T,$$

$$\|Z\| = \sqrt{Z_1^2 + Z_2^2 + \dots + Z_s^2}, Z_1, Z_2, \dots, Z_s \sim N(0, 1)。$$

$$\text{接受概率为 } \alpha(x, y) = \min\left(1, \frac{f(y)q(\lambda | d, x)}{f(x)q(\lambda | d, y)}\right)。$$

当前样本值为 X_t , 所以我们的算法如下:

- 1) s 维单位超球体上均匀分布抽样产生随机方向 d 。
- 2) $q(\lambda | d, X_t)$ 抽样产生 λ , 计算 $Y_t = X_t + \lambda d$ 。
- 3) 若 $U \leq \alpha(X_t, Y_t)$, $X_{t+1} = Y_t$; 否则, $X_{t+1} = X_t$ 。

方法二：取 S_0 为零时刻的资产价格， $T = nh$ ，那么 n 就是平均时所选取的时间周期数， h 为时间增量， $npath$ 为仿真的路径数。我们模拟 $npath$ 条路径，每一条路径都产生独立的收益。

运用算法如下：

1) 取记号 $a1, a2$ ，其中 $a1 = \exp\{(r - 0.5 * \sigma^2) * h\}$ ， $a2 = \sigma * \sqrt{X}$ ，计算 $a1, a2$ ；

2) 若 $S(T) > K$ ，则 $V(S(T), T) = S(T) - K$ ，否则 $V(S(T), T) = 0$ ；

3) 计算价格 $S(t)$ 。这里取利率 $r = 0.05$ ， $x_0 = 50$ ， $\sigma = 0.3$ ， $T = 1$ ， $n = 16$ ， $npath = 25000$ ， $K = 55$ ，可以求解到股票价格 $S = 25.2245$ 。

6. 总结

由定义可知马尔可夫链是某一时刻取值当且仅当与之前相邻的一个时刻相关而与之前时刻没有联系，也就是它可以忘记过去，并且，当它经过一段时间的转移以后，就与初始时刻没有了联系，状态也会从非平稳到达平稳，马尔可夫链也就变得收敛。为了使得马尔可夫链尽快忘记初始状态，然后比较接近平稳状态，降低随机抽样的误差性，所以需要降低方差。因此，通过改变接受概率，减少模拟抽样的时间，加快收敛速度，从而达到较高的模拟效率。

通过随机模拟，可以将复杂的模型[14]简单化，以上示例显示了 MCMC 算法的便捷性，只要通过选择合适的建议概率分布，就可以很方便的构造平稳的马尔可夫链，并得到随机样本并估计出相关的参数值。

参考文献

- [1] Hasting, W. (1970) Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika*, **57**, 97-109. <https://doi.org/10.1093/biomet/57.1.97>
- [2] 刘军. 科学计算中的蒙特卡罗策略[M]. 北京: 高等教育出版社, 2005.
- [3] Rubinstein, R. (1981) Simulation and Monte Carlo Method. John Wiley, New York.
- [4] Chib, S. and Greenberg, E. (1995) Understanding the Metropolis-Hasting Algorithm. *Taylor and Francis*, **49**, 327-335.
- [5] Tierney, L. (1994) Markov Chains for Exploring Posterior Distributions. *The Annals of Statistics*, **22**, 1701-1728. <https://doi.org/10.1214/aos/1176325750>
- [6] Andrieu, C. (2003) An Introduction to MCMC for Machine Learning. *Machine Learning*, **50**, 5-43. <https://doi.org/10.1023/A:1020281327116>
- [7] Brooks, S., Gelman, A. and Jones, L. (2011) Handbook of Markov Chain Monte Carlo. A Chapman and Hall Book.
- [8] Roberts, G. (2004) General State Space Markov Chains and MCMC Algorithms. *Probability Surveys*, **1**, 20-71. <https://doi.org/10.1214/154957804100000024>
- [9] Liang, F. and Liu, C. (2010) Advanced Markov Chain Monte Carlo Methods. John and Sons Ltd. Publication. <https://doi.org/10.1002/9780470669723>
- [10] 康崇禄. 蒙特卡罗方法理论和应用[M]. 北京: 科学出版社, 2014.
- [11] 肖柳青. 随机模拟方法与应用[M]. 北京: 北京大学出版社, 2014.
- [12] 张波. 应用随机过程[M]. 北京: 中国人民大学出版社, 2017.
- [13] Dagpunar, J. (2007) Simulation and Monte Carlo: With Applications in Finance and MCMC. John Wiley and Sons Ltd.
- [14] Lyas, A. (2016) Importance Sampling for Least-Square Monte Carlo Methods. *Degree Project in Mathematics*.

附 录:**A 附录1**

```
function [x,m] = rw(x0,a,iter)
x=x0;
for t=0:(iter+1)
    y=rand () *2*a-a+x;
    m=min(1, exp((x^2-y^2)/2));
    u=rand ();
    if u<=m
        x=y;
    end
end
x0=-4;
>> a=0.5;
>> n=500;
>> x=zeros(1, n+1);
>> iter= [0:1: n];
>> for t=0: n
[x(t+1), m(t+1)] =rw(x0,a,iter(t+1));
plot(iter,x)
plot(x,m)
```

B 附录2

```
function [x,m] = rws(x0,a,iter)
x=x0;
for t=0:(iter+1)
    u1=rand ();
    u2=rand ();
    u=rand ();
    x=(-2*log(u1))^(1/2)*cos(2*pi*u2);
    y=x+a*(2*u-1);
    m=min(1, exp((-x^2+y^2)/2));
    if u<=m
        x=y;
    end
end
x0=-4;
>> a=0.5;
>> n=500;
>> x=zeros(1, n+1);
>> iter=[0:1: n];
>> for t=0: n
[x(t+1), m(t+1)] =rw(x0, a,iter(t+1));
plot(iter,x)
plot(x,m)
```

C 附录3

```
function [price1, price2, x]=MCMC(r,x,sigma,T,n,npath,K)
h=T/n;
xc=0;
a1=exp((r-0.5*sigma^2) *h);
a2=sigma*sqrt(h);
s1=0;
```

```

s2=0;
for i=1: npath
    z=normrnd (0,1);
    x=x*a1*exp(a2*z);
    xc=xc+x;
    i=i+1;
end
if x>K
    R=x-K;
else
    R=0;
end
s1=s1+R;
s2=s2+R^2;
theta=s1/npath;
stderr=sqrt((s2-s1^2/npath)/npath/(npath-1));
if x-K>0
    price1=exp(-r*T) *theta;
    price2=0;
else
    price2=exp(-r*T) *stderr;
    price1=0;
end
[price1, price2, x]=MCMC(0.05,50,0.3,1,16,25000,55)
price1 =          25.2245
price2 =          0
x=          6.6300e+05

```

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2324-7991, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: aam@hanspub.org