随机过程实验报告

班级：　应数拔尖 1501

姓名：　冯洲

学号：　U201510104

指导老师：　王湘君

MCMC Methods
The Metropolis-Hastings Sampler

1. 实验目的：

利用 Metropolis-Hastings 算法，通过 MATLAB 来构造并模拟特定的 Markov 链，进而得到服从目标分布的样本。通过模拟来运用学习到的有关 MCMC 这类方法的知识，进而深刻体会 Metropolis-Hastings 抽样的优势，并通过比较来思考其缺点。为今后进一步学习统计模拟打基础。

2. 实验摘要：

通过对统计模拟参考书【1】学习，我认为 Metropolis-Hastings 算法相对于传统的抽样算法（如拒绝-接受抽样法）主要有两个优势:1,可以从未归一化的目标分布进行抽样，即已知的分布与最终的目标分布（target distribution）成比例即可。2，可以有效摆脱传统方法受到的维数的困扰，进行高维随机向量的抽样。针对这两个优势，我分别设计两次实验进行模拟来体现之。

首先，回忆数理统计课本【2】，参数估计中有关 Bayes 估计的内容可以知道，在 Bayes 公式中的作为分母的样本边际分布，需要根据先验分布（prior）来进行计算，有时很复杂。而我们需要的只是后验分布（posterior），依据其对参数进行估计。所以 MH 算法在此可以发挥其作用，避免了繁琐的计算，用后验分布样本的估计量来估计参数。画出相应的 Markov 链的轨道，最后将样本的分布图与理论解析图进行比较。

然后，由现有知识知道反函数法与拒绝接受法到了高维就很难实现，而 MH 算法不然。选取一二维的非标准正态分布，利用 MH 算法构造服从它的样本，并画出样本的分布图与解析图进行比较分析。

最后，对实验结果，M-H 算法的优缺点进行分析，结合【1】中进一步的知识以及 Wikipedia【3】中的理论介绍对实验进行总结。

关键词：Bayes 后验分布抽样；二维正态分布抽样

3.1： Bayes 后验分布抽样

3.1.1 问题背景介绍及条件给出

在 Bayes 估计中，未知参数基于后验分布 p(θ|y)确定，这是基于观测数据的在可能参数上的概率分布。而后验分布由 Bayes 定理来确定：

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$$

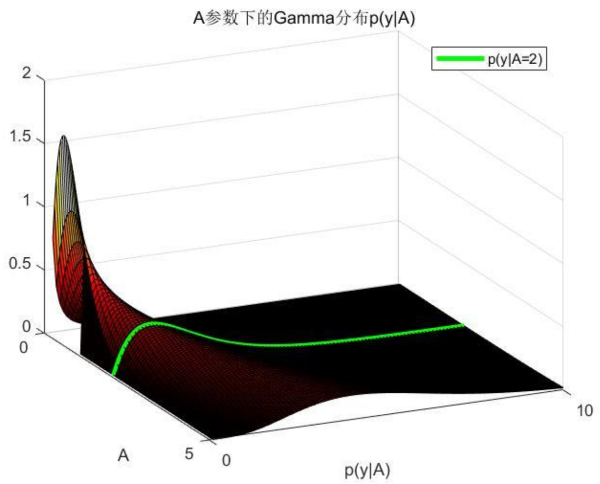
其中，p(y)是一个归一化常数(样本边际分布)，通常很难明确地确定，因为它涉及计算参数的先验分布的积分。然后假设有下面模型:

$$p(y|\theta) = Gamma(y;A,B)$$

即样本服从形状参数为 A，尺度参数为 B 的Γ(A,B)分布：

$$Gamma(y;A,B) = \frac{B^A}{\Gamma(A)}y^{A-1}e^{-By}$$

其中，Γ()是Γ函数。因此，模型的参数为$\theta = [A \quad B]$。下面画出了Γ分布在参数 B=1，A 从 0 到 5 下的密度曲面图，其中$p(y|A=2,B=1)$用绿色的线条标出:



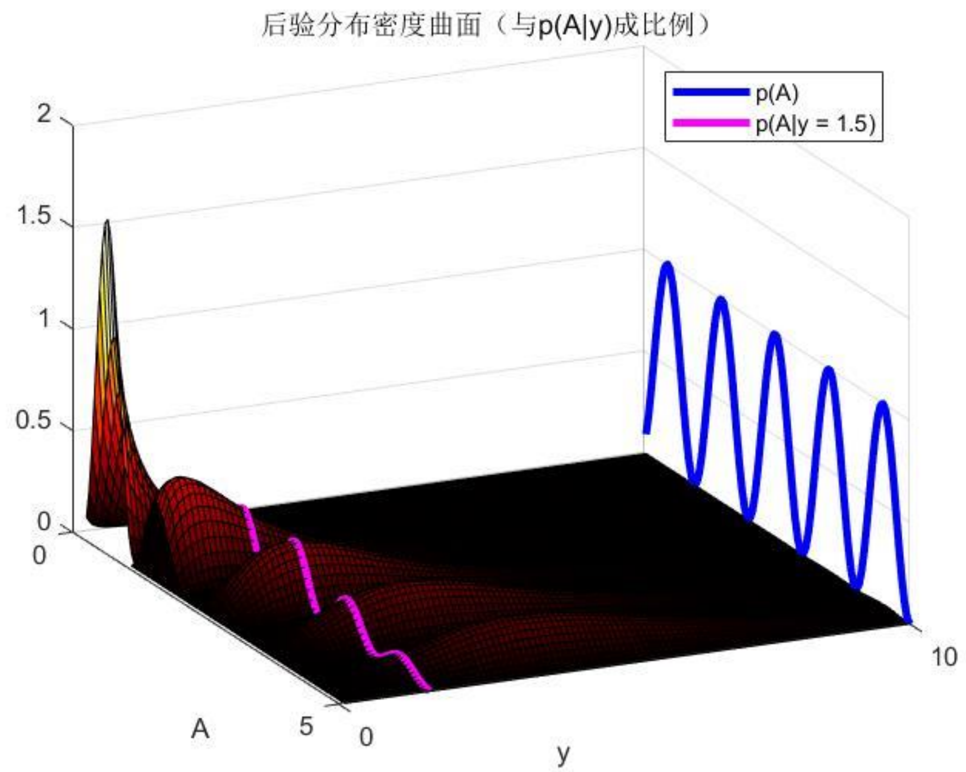
现在，假设对于参数 A 与 B（先验分布）服从以下模型：

$$p(B=1)=1, p(A)=\sin(\pi A)^2$$

第一个条件说明 B 只取 1，所以将 B 当作一个常数即可；第二个条件说明 A 的概率成正弦波动，因此只用估计参数 A 的值即可。请注意这里 A 和 B 的分布都没有归一化（improper），但是不影响我们的抽样，因为考虑这些常数以及分母 p(y)，最后我们有后验分布满足：

$$p(A|y) \propto p(y|A)p(A)$$

下面我用 MATLAB 画出上式右边概率 $p(y|A)p(A)$（从 0 到 10，A 从 0 到 5），p(A)用蓝色曲线标出。而对于目标分布——后验分布，我取 y=1.5 条件下，利用 Metropolis-Hastings 算法对后验分布 $p(A|y=1.5)$ 进行抽样估计，特别的，我将目标分布用红色曲线标出，如下图：



3.1.2Metropolis-Hastings 算法的构造与实现

根据统计模拟参考书【1】中的算法过程及原理，流程图如下：

i.设定 t=0.

ii 生成初始状态$x_0 \sim \pi_0$

iii 重复直到 t=M

令 t=t+1

生成建议状态x^*,服从条件分布$q(x|x^{t-1})$

计算建议矫正因子$c = \frac{q(x^{(t-1)}|x^*)}{q(x^*|x^{(t-1)})}$

计算接受率$\alpha = \min\left(1, \frac{p(x^*)}{p(x^{(t-1)})} \times c\right)$

做实验：生成服从均匀分布 U[0,1]的随机数 u

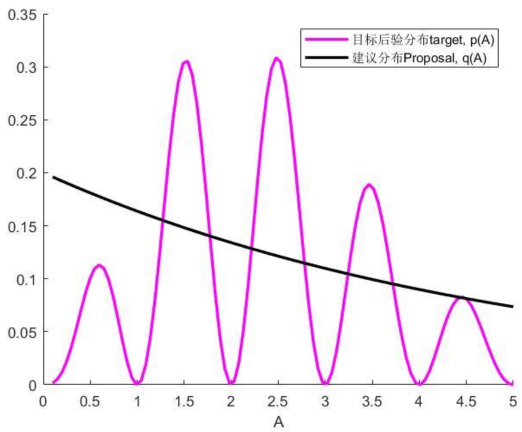
如果 u<α,则接受x^*，令$x^t = x^*$.

反之，令$x^t = x^{t-1}$

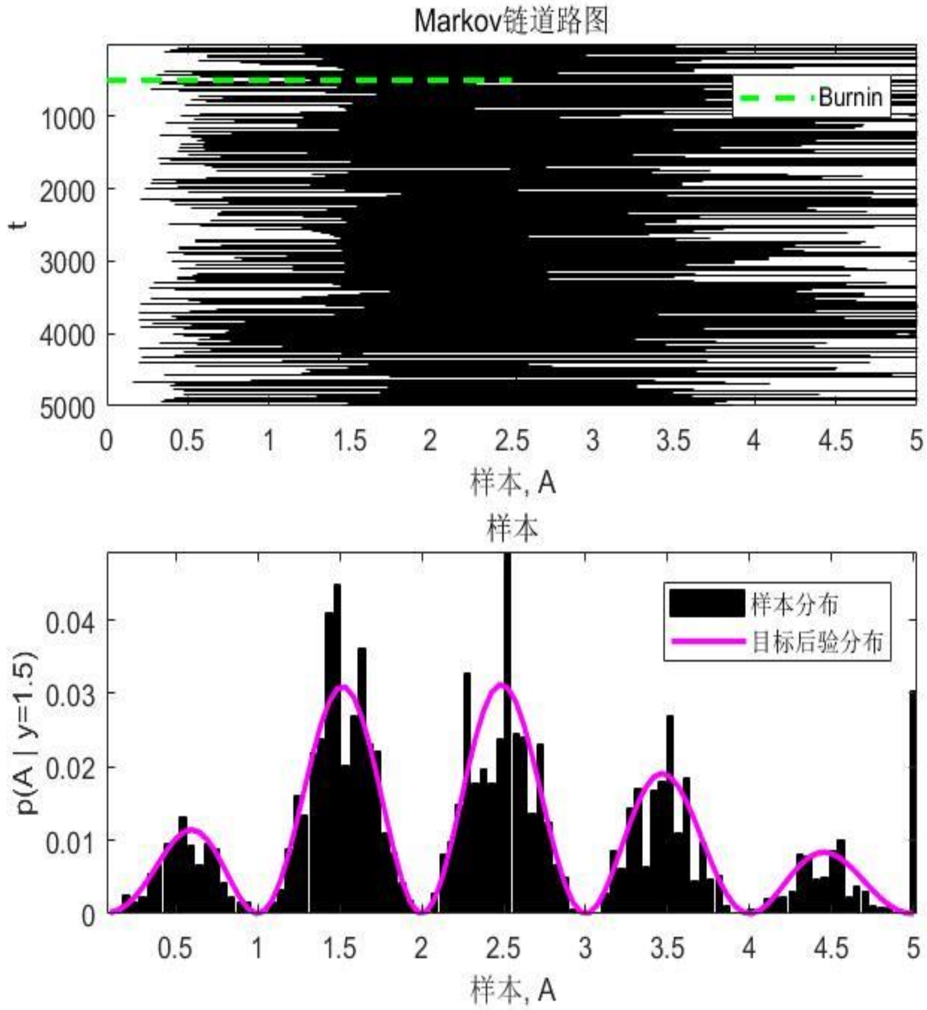
其中 q 为算法中的建议分布，根据不同的实际模型，q 的选取可以不一样，而且根据 Wikipedia 的评判指标，q 也有适用于具体模型的好坏之分。而对于这个模型中参数 A 是大于等于零的，而根据以往的学习取值在正实数的分布比较常见的就是指数分布：

$$q(A) = EXP(\mu) = \mu e^{-\mu A}$$

下面就将其作为建议分布(proposal distribution)，参数μ=5 时，目标分布以及建议分布的密度函数图如下：



然后，按照 Metropolis-Hastings 算法编程得到的 Markov 链的道路图以及最终的抽样结果如下：



其中，道路图中的 Burnin 绿色分割线是根据 Markov 链的理论【1】知道前一部分的样本依赖初始分布 π0 很大，参考价值不高，而到底舍弃多少样本需要更进一步的知识与讨论，所以我就舍弃前 500 个样本（样本总容量为 5000）。其实，这个问题也是 M-H 算法的缺点之一，要“浪费”一些数据点。而从图二可以看出样本的大致趋势还是和目标分布相接近的，所以粗略地讲 Metropolis-Hastings 算法是成功的。

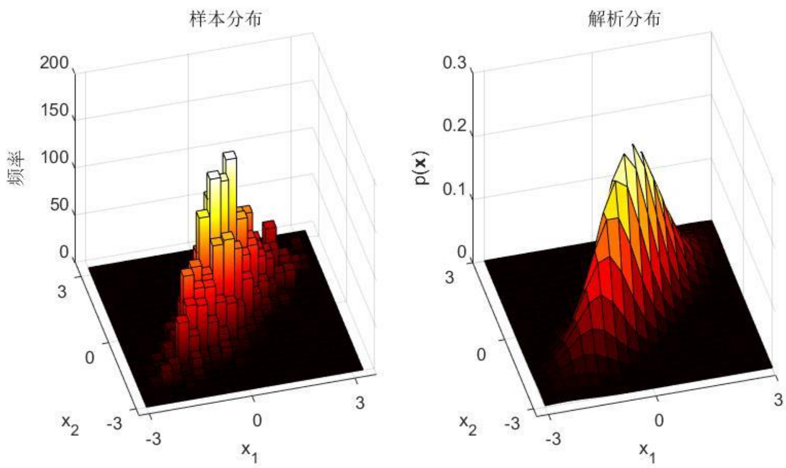
3.2 二维正态分布的抽样

3.2.1 目标分布给出及 M-H 算法构造求解

首先，我们的目标分布为一二维的正态分布，其参数设定如下：

$$p(x)=N(\mu,\Sigma)\text{, 均值 } \mu=[0,0]\text{，协方差矩阵：} \Sigma=\begin{bmatrix}1 & 0.8\\ 0.8 & 1\end{bmatrix},$$

由于 MATLAB 中有 mvnpdf 命令，所以目标分布的解析密度曲面容易生成。而相应的在 Metropolis-Hastings 算法中，我选取的建议分布(proposal distribution)为以前一个状态为中心的，协方差矩阵为单位矩阵 I 的正态分布，而且服从该分布的样本易得，下面为 MATLAB 中运行得到的样本图以及解析的目标分布图：



在构造算法的过程中发现矫正因子 c==1，因为建议分布是对称的，所以运行过程中可以设 c=1，避免重复计算。根据 Wikipedia【3】，建议分布对称时，算法也叫 Metropolis 算法。比较两图可知在二维的情况下 Metropolis-Hastings 算法同样有效。

4 实验分析与总结

通过以上两个 MCMC 的统计模拟的例子，我们可以清楚的看到 Metropolis-Hastings 算法的优势：1，已知分布可以相差常系数（本质上可以从算法推导过程中的接受率的计算可以看出，常系数可约去）。2，高维情形算法同样非常有效。

但是本实验也存在很多待改进之处，这需要更进一步的知识：

1. 两次实验的建议分布都是直接凭经验取得，没有过多讨论其合理性及有效性
2. 在实验一中，抛弃的样本容量大小选取没有定量的依据，根据统计模拟参考书【1】中的改进方法，可以利用 Gibbs-Sampler 改进，做到所谓的“无抛弃”。
3. 两次实验模拟的结果以及理论结果没有定量的进行评价，只是通过图表判断算法的有效性，说服力不强。

个人小结：通过本次实验报告的完成，我初步了解了 MCMC-methods 的大致思想，也体会到了其中代表的 Metropolis-Hastings 算法的优势，锻炼了课外知识获取的能力。最后，我通过 MATLAB 的统计模拟，熟悉了其在统计方面的功能。

5 参考文献及资料

【1】Ross, S. M. (2013). Simulation (Fifth Edition). China Machine Press.

【2】数理统计教程/王兆军.邹长亮编著（2013）-北京：高等教育出版社

【3】Wikipedia, https://en.wikipedia.org/wiki/Metropolis-Hastings_algorithm

【4】随机过程（第五版）/刘次华编著-武汉：华中科技大学出版社

6 实验代码

6.1Bayes 后验分布抽样代码

```
% METROPOLIS-HASTINGS bayes 后验分布抽样 by fz
clc,clear
rand('seed',12345)
% 先验分布的尺度参数
B = 1;
% 定义极大似然函数
```

```
likelihood = inline('(B.^A/gamma(A)).*y.^(A-1).*exp(-(B.*y))','y','A','B');
% 计算并展示 Gamma 分布即图 1
yy = linspace(0,10,100);
AA = linspace(0.1,5,100);
likeSurf = zeros(numel(yy),numel(AA));
for iA = 1:numel(AA); likeSurf(:,iA)=likelihood(yy(:),AA(iA),B); end;

figure;
surf(likeSurf);ylabel('p(y|A)');xlabel('A'); colormap hot
% 展示 A=2, B=1 时的密度曲线
Hold on; ly = plot3(ones(1,numel(AA))*40,1:100,likeSurf(:,40),'g','linewidth',3)
xlim([0 100]); ylim([0 100]);    axis normal
set(gca,'XTick',[0,100]); set(gca,'XTickLabel',[0 5]);
set(gca,'YTick',[0,100]); set(gca,'YTickLabel',[0 10]);
view(65,25)
legend(ly,'p(y|A=2)','Location','Northeast');
hold off;
title('A 参数下的 Gamma 分布 p(y|A)');
% 定义形状参数 A 的先验分布
prior = inline('sin(pi*A).^2','A');
% 定义后验分布
p = inline('(B.^A/gamma(A)).*y.^(A-1).*exp(-(B.*y)).*sin(pi*A).^2','y','A','B');
% 计算并展示后验分布
postSurf = zeros(size(likeSurf));
for iA = 1:numel(AA); postSurf(:,iA)=p(yy(:),AA(iA),B); end;
figure
surf(postSurf); ylabel('y'); xlabel('A'); colormap hot
% 展示 A
hold on; pA = plot3(1:100,ones(1,numel(AA))*100,prior(AA),'b','linewidth',3)
% 抽样  p(A | y = 1.5)
y = 1.5;
target = postSurf(16,:);
% 展示后验分布
psA = plot3(1:100, ones(1,numel(AA))*16,postSurf(16,:),'m','linewidth',3)
xlim([0 100]); ylim([0 100]);    axis normal
set(gca,'XTick',[0,100]); set(gca,'XTickLabel',[0 5]);
set(gca,'YTick',[0,100]); set(gca,'YTickLabel',[0 10]);
view(65,25)
legend([pA,psA],{'p(A)','p(A|y = 1.5)'},'Location','Northeast');
hold off
title('后验分布密度曲面（与 p(A|y)成比例）');
% 初始话 Metropolis—Hastings 算法
% 定义建议分布 proposal distribution
q = inline('exppdf(x,mu)','x','mu');
% 建议分布的参数设置
mu = 5;
% 展示目标分布以及建议分布
figure; hold on;
th = plot(AA,target,'m','Linewidth',2);
qh = plot(AA,q(AA,mu),'k','Linewidth',2)
legend([th,qh],{'目标后验分布 target, p(A)','建议分布 Proposal, q(A)'});
xlabel('A');
% 一些限制条件：取值范围及样本容量
nSamples = 5000;
burnIn = 500;
minn = 0.1; maxx = 5;
% 初始化样本
x = zeros(1 ,nSamples);
x(1) = mu;
t = 1;
% 跑动算法  METROPOLIS-HASTINGS
while t < nSamples
    t = t+1;
    % 建议状态
    xStar = exprnd(mu);
    % 矫正因子
    c = q(x(t-1),mu)/q(xStar,mu);
    % 接受率
    alpha = min([1, p(y,xStar,B)/p(y,x(t-1),B)*c]);

    % 拒绝或接受
    u = rand;
    if u < alpha
        x(t) = xStar;
    else
        x(t) = x(t-1);
    end
```

```
end
% 展示 MARKOV 链的道路
figure;
subplot(211);
stairs(x(1:t),1:t, 'k');
hold on;
hb = plot([0 maxx/2],[burnIn burnIn],'g--','Linewidth',2)
ylabel('t'); xlabel('样本, A');
set(gca , 'YDir', 'reverse');
ylim([0 t])
axis tight;
xlim([0 maxx]);
title('Markov 链道路图');
legend(hb,'Burnin');
% 展示样本
subplot(212);
nBins = 100;
sampleBins = linspace(minn,maxx,nBins);
counts = hist(x(burnIn:end), sampleBins);
bar(sampleBins, counts/sum(counts), 'k');
xlabel('样本, A' ); ylabel( 'p(A | y=1.5)' );
title('样本');
xlim([0 10])
% 展示目标分布
hold on;
plot(AA, target/sum(target) , 'm-', 'LineWidth', 2);
legend('样本分布',sprintf('目标后验分布'))
axis tight
```

6.2 二维正态分布抽样代码

```
% METROPOLIS-HASTINGS 二维正态分布抽样
rand('seed' ,12345);
D = 2; % #  维数
nBurnIn = 100;
%  二维正态分布（目标）
p = inline('mvnpdf(x,[0 0],[1 0.8;0.8 1])','x');
%建议分布 proposal 为标准正态分布
q = inline('mvnpdf(x,mu)','x','mu')
nSamples = 5000;
minn = [-3 -3];
maxx = [3 3];
%  初始化抽样
t = 1;
x = zeros(nSamples,2);
x(1,:) = randn(1,D);
%运行抽样
while t < nSamples
	t = t + 1;
	%  建议状态
	xStar = mvnrnd(x(t-1,:),eye(D));
	%  矫正因子(直接可以取  1)
	c = q(x(t-1,:),xStar)/q(xStar,x(t-1,:));
	%  计算接受率
	alpha = min([1, p(xStar)/p(x(t-1,:))]);

	%  拒绝或者接受
	u = rand;
	if u < alpha
		x(t,:) = xStar;
	else
		x(t,:) = x(t-1,:);
	end
end
%  展示
nBins = 20;
bins1 = linspace(minn(1), maxx(1), nBins);
bins2 = linspace(minn(2), maxx(2), nBins);
%  展示样本分布
ax = subplot(121);
bins1 = linspace(minn(1), maxx(1), nBins);
bins2 = linspace(minn(2), maxx(2), nBins);
sampX = hist3(x, 'Edges', {bins1, bins2});
```

```
hist3(x, 'Edges', {bins1, bins2});
view(-15,40)
% 根据高度给柱状图上色
colormap hot
set(gcf,'renderer','opengl');
set(get(gca,'child'),'FaceColor','interp','CDataMode','auto');
xlabel('x_1'); ylabel('x_2'); zlabel('频率');
axis square
set(ax,'xTick',[minn(1),0,maxx(1)]);
set(ax,'yTick',[minn(2),0,maxx(2)]);
title('样本分布');
% 展示解析密度
ax = subplot(122);
[x1 ,x2] = meshgrid(bins1,bins2);
probX = p([x1(:), x2(:)]);
probX = reshape(probX ,nBins, nBins);
surf(probX); axis xy
view(-15,40)
xlabel('x_1'); ylabel('x_2'); zlabel('p({\bfx})');
colormap hot
axis square
set(ax,'xTick',[1,round(nBins/2),nBins]);
set(ax,'xTickLabel',[minn(1),0,maxx(1)]);
set(ax,'yTick',[1,round(nBins/2),nBins]);
set(ax,'yTickLabel',[minn(2),0,maxx(2)]);
title('解析分布')
```