Statistics 572
Homework 8, Extra Credit

Alejandro Robles

December 13th, 2018

1 Bivariate Normal Distribution using Monte Carlo

Generate Monte Carlo from Bivariate Normal with Mean and Sigma described below.

$$\mu = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

and

$$\Sigma = \begin{bmatrix} 1.5 & 0.6 \\ 0.6 & 1 \end{bmatrix}$$

1. Plot each Monte Carlo and denisty histograms.
2. Plot scatterplot X_1 versus X_2
3. Give means, standard deviations, skewness, and Kurtosis of each chain.

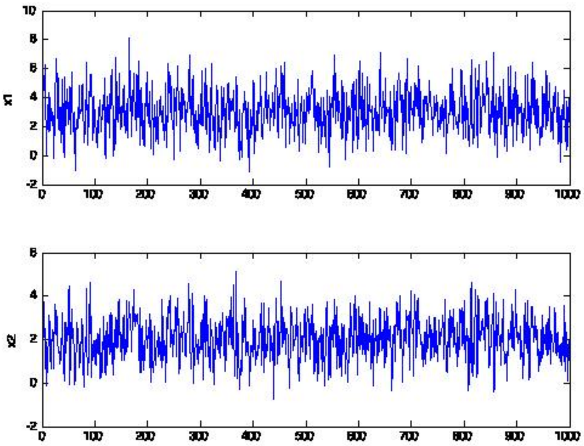
MATLAB Code:

```
mus = [3 ; 2];
sigmas = [1.5 0.6 ; 0.6 1];
[x1, x2] = mc_bn(mus,sigmas, 1000);
```

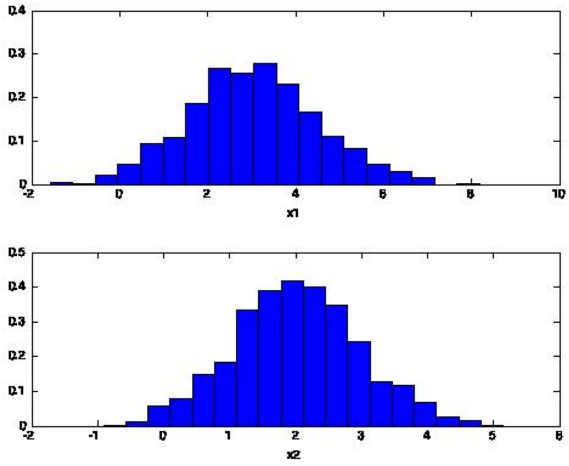
Result:

```
Mean for x1 and x2 are: (3.0439, 2.01738)
Standard Deviations for x1 and x2 are: (1.46935, 0.960939)
Kurtosis for x1 and x2 are: (2.91271, 2.89512)
```

Plots:



(a) Monte Carlo Chain



(b) Denisty Histogram

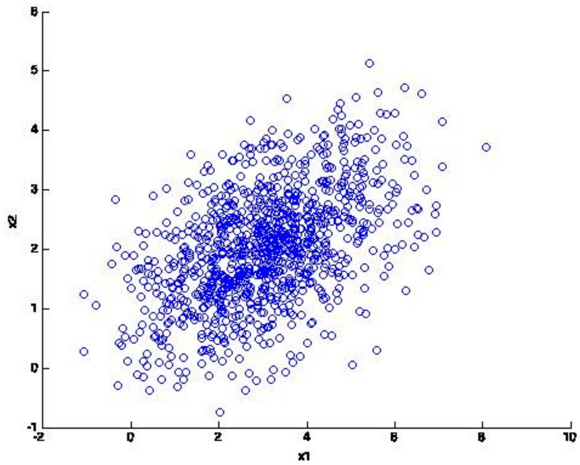


Figure 1: Scatter Plot

Discussion: As we can see, the plots show that for x1 the mean is around 3 and for x2 the mean is about 2. The mean and standard deviations also correspond to the inputs we wanted. The scatter plot is showing data that look like it follows a bivariate normal distribution with the parameters we wanted. The monte carlo chains also oscillate around 3 and 2 for x1 and x2.

User Defined Functions:

```
function [MCx, MCy] = mc_bn(mus, sigmas, n)
sigma1 = sigmas(1);
sigma2 = sigmas(4);
rho = sigmas(2)/(sqrt(sigmas(1)) * sqrt(sigmas(4)));

xgibbs = zeros(n,2);

y = mus; % This is the mean.
sig = sqrt(1-rho^2);
% Initial point.
x10 = y(1) + sigma1*randn(1);
x20 = y(2) + rho*(sigma2/sigma1)*(x10 - y(1)) + sigma2*sig*randn(1);
xgibbs(1,:) = [x10, x20];
% Start the chain.
for i = 2:n
   mu = y(1) + rho*(sigma1/sigma2)*(xgibbs(i-1,2)-y(2));
   xgibbs(i,1) = mu + sigma1*sig*randn(1);
   mu = y(2) + rho*(sigma2/sigma1)*(xgibbs(i,1) - y(1));
   xgibbs(i,2) = mu + sigma2*sig*randn(1);
end

MCx = xgibbs(:,1);
MCy = xgibbs(:,2);

figure(1)
scatter(MCx,MCy)
xlabel('x1')
ylabel('x2')


figure(2)
subplot(211)
plot(MCx)
ylabel('x1')
subplot(212)
plot(MCy)
ylabel('x2')


figure(3)
nn = length(MCx);

%h = .1;
h = 3.5 * std(MCx) * nn^(-1/3);
t0 = min(MCx)-0.5;
tm = max(MCx)+0.5;
bins = t0:h:tm;
vk = histc(MCx,bins);
vk(end) = [];
fhat = vk/(nn*h);

tm = max(bins);
bc = (t0+h/2):h:(tm-h/2);
subplot(211)
```

```
bar(bc,fhat,1,'w')
xlabel('x1')

subplot(212)
nn = length(MCy);

%h = .1;
h = 3.5 * std(MCy) * nn^(-1/3);
t0 = min(MCy)-0.5;
tm = max(MCy)+0.5;
bins = t0:h:tm;
vk = histc(MCy,bins);
vk(end) = [];
fhat = vk/(nn*h);

tm = max(bins);
bc = (t0+h/2):h:(tm-h/2);
bar(bc,fhat,1,'w')
xlabel('x2')

means = mean([MCx MCy]);
stds = std([MCx MCy]);
ks = kurtosis([MCx MCy]);

sprintf('Mean for x1 and x2 are: (%g, %g)' , means(1), means(2))
sprintf('Standard Deviations for x1 and x2 are: (%g, %g)' , stds(1), stds(2))
sprintf('Kurtosis for x1 and x2 are: (%g, %g)' , ks(1), ks(2))
```