STA 360: Lab 6

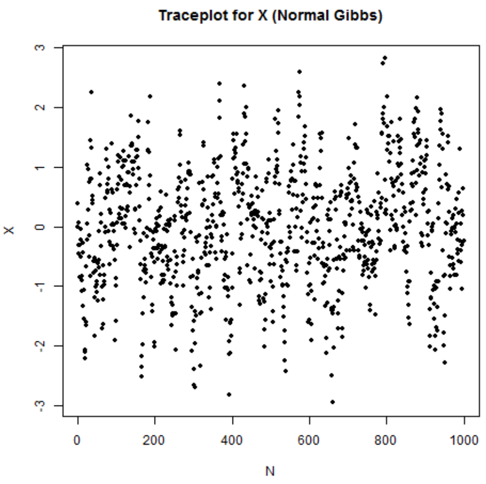
Michael Lin

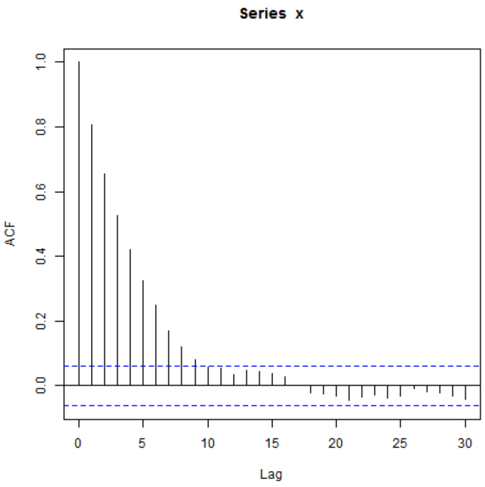
March 5, 2015

1. Since all means are 0, the full conditional for each of the three variables are derived as follows:

$$
\begin{aligned}
X|Y,Z &\sim N(\Sigma_{X,(Y,Z)}\Sigma_{Y,Z}^{-1}\begin{bmatrix}Y\\Z\end{bmatrix}, \Sigma_X - \Sigma_{X,(Y,Z)}\Sigma_{Y,Z}^{-1}\Sigma_{(Y,Z),X}) \\
&= N(\begin{bmatrix}0.9 & 0.1\end{bmatrix}\begin{bmatrix}1 & 0.1\\0.1 & 1\end{bmatrix}^{-1}\begin{bmatrix}Y\\Z\end{bmatrix}, 1 - \begin{bmatrix}0.9 & 0.1\end{bmatrix}\begin{bmatrix}1 & 0.1\\0.1 & 1\end{bmatrix}^{-1}\begin{bmatrix}0.9\\0.1\end{bmatrix}) \\
&= N(0.89899Y + 0.01010Z, 0.18990) \\
Y|X,Z &\sim N(\Sigma_{Y,(X,Z)}\Sigma_{X,Z}^{-1}\begin{bmatrix}X\\Z\end{bmatrix}, \Sigma_Y - \Sigma_{Y,(X,Z)}\Sigma_{X,Z}^{-1}\Sigma_{(X,Z),Y}) \\
&= N(\begin{bmatrix}0.9 & 0.1\end{bmatrix}\begin{bmatrix}1 & 0.1\\0.1 & 1\end{bmatrix}^{-1}\begin{bmatrix}X\\Z\end{bmatrix}, 1 - \begin{bmatrix}0.9 & 0.1\end{bmatrix}\begin{bmatrix}1 & 0.1\\0.1 & 1\end{bmatrix}^{-1}\begin{bmatrix}0.9\\0.1\end{bmatrix}) \\
&= N(0.89899X + 0.01010Z, 0.18990) \\
Z|X,Y &\sim N(\Sigma_{Z,(X,Y)}\Sigma_{X,Y}^{-1}\begin{bmatrix}X\\Y\end{bmatrix}, \Sigma_Z - \Sigma_{Z,(X,Y)}\Sigma_{X,Y}^{-1}\Sigma_{(X,Y),Z}) \\
&= N(\begin{bmatrix}0.1 & 0.1\end{bmatrix}\begin{bmatrix}1 & 0.9\\0.9 & 1\end{bmatrix}^{-1}\begin{bmatrix}X\\Y\end{bmatrix}, 1 - \begin{bmatrix}0.1 & 0.1\end{bmatrix}\begin{bmatrix}1 & 0.9\\0.9 & 1\end{bmatrix}^{-1}\begin{bmatrix}0.1\\0.1\end{bmatrix}) \\
&= N(0.052632X + 0.052632Y, 0.98947)
\end{aligned}
$$

2. See R code. Below are the traceplot and autocorrelation function plot for X:



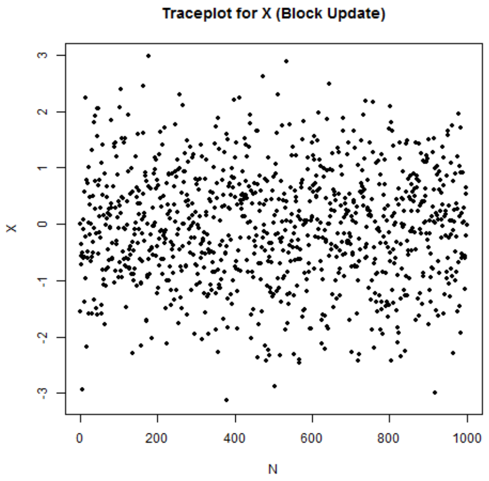


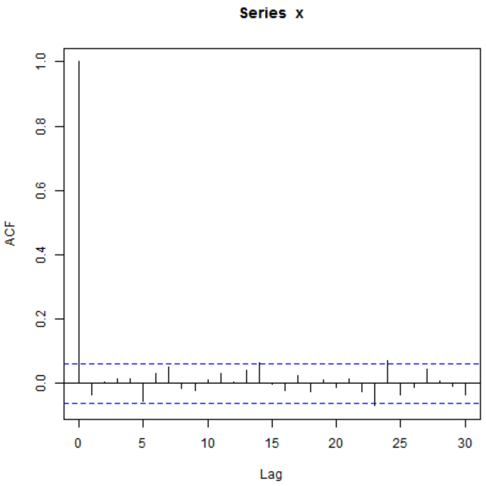
The traceplot suggests that the sampler has traversed through the entire probability. However, the correlation between successive samples seems to be high, as indicated in the autocorrelation function plot for small lag values. This means that although the sampler is able to cover the entire probability space, it does so very slowly since each successive sample is correlated with previous samples. As a result, each update cannot move very far from previous point.

3. Using similar approach as above, the full conditionals are:

$$\begin{aligned} X, Y|Z &\sim MVN(\Sigma_{(X,Y),Z}\Sigma_Z^{-1}Z, \Sigma_{X,Y} - \Sigma_{(X,Y),Z}\Sigma_Z^{-1}\Sigma_{Z,(X,Y)}) \\ &= MVN(\begin{bmatrix}0.1\\0.1\end{bmatrix} \cdot 1 \cdot Z, \begin{bmatrix}1 & 0.9\\0.9 & 1\end{bmatrix} - \begin{bmatrix}0.1\\0.1\end{bmatrix} \cdot 1 \cdot \begin{bmatrix}0.1 & 0.1\end{bmatrix}) \\ &= MVN(\begin{bmatrix}0.1Z\\0.1Z\end{bmatrix}, \begin{bmatrix}0.99 & 0.89\\0.89 & 0.99\end{bmatrix}) \\ Z|X, Y &\sim N(0.052632X + 0.052632Y, 0.98947) \end{aligned}$$

Below are the traceplot and autocorrelation function plot for X:





The traceplot suggests that this sample has also traversed through the entire probability space. It was able to do so very quickly because successive samples are not highly correlations, as seen in the autocorrelation function plot.

4. Although both samplers seemed to have covered the entire probability space, the normal Gibbs sample was less precise and look longer than block updates because of relatively high autocorrelation. The block updates is more efficient because there was less dependence on conditional variable, i.e., each update requires conditioning twice, whereas the normal Gibbs sampler required conditioning three times.

R code for normal Gibbs sampler:

```
N = 1000
x = rep(0,N)
y = rep(0,N)
z = rep(0,N)

x[1] = rnorm(1, mean = 0, sd = 0.18990^0.5);
y[1] = rnorm(1, mean = 0, sd = 0.18990^0.5);
z[1] = rnorm(1, mean = 0.052632*x[1]+0.052632*y[1], sd = 0.98947^0.5)


for(i in 2:N){
  x[i] = rnorm(1, mean = 0.89899*y[i-1]+0.01010*z[i-1], sd = 0.18990^0.5)
  y[i] = rnorm(1, mean = 0.89899*x[i]+0.01010*z[i-1], sd =0.18990^0.5)
  z[i] = rnorm(1, mean = 0.052632*x[i]+0.052632*y[i], sd = 0.98947^0.5)
}

png("tracenorm.png")
plot(x, pch=20, xlab="N", ylab="X", main="Traceplot for X (Normal Gibbs)")
dev.off()

png("acfnorm.png")
acf(x)
dev.off()
```

R code for block updates:

```
library(MASS)
N = 1000

xy.samp = matrix(rep(0,2*N), nrow=N, ncol=2, byrow=T)
colnames(xy.samp) = c("x","y")

z = rep(0,N)
z[1] = rnorm(1, mean = 0, sd = 0.98^0.5)

for(i in 2:N){
  xy.mean = c(0.1*z[i-1],0.1*z[i-1])
  xy.var = matrix(c(0.99,0.89,0.89,0.99), nrow=2, ncol=2, byrow = T)

  xy.samp[i,] = mvrnorm(1, xy.mean, xy.var)
  z[i] = rnorm(1, mean = 0.052632*xy.samp[i,1]+0.052632*xy.samp[i,2], sd = 0.98947^0.5)
}
x = xy.samp[,1]
y = xy.samp[,2]

png("traceblock.png")
plot(x, pch=20, xlab="N", ylab="X", main="Traceplot for X (Block Update)")
dev.off()

png("acfblock.png")
acf(x)
dev.off()
```