ps6\_rmd

Jared Yu

March 6, 2020

library(latex2exp)  
### Problem 2  
g1 <- function(x2, mu1, mu2, sigma1, sigma2, rho) {  
 rnorm(n = 1, mean = mu1 + rho \* (sigma1 / sigma2) \* (x2 - mu2),  
 sd = sqrt((sigma1^2) \* (1 - rho^2)))  
}  
  
g2 <- function(x1, mu1, mu2, sigma1, sigma2, rho) {  
 rnorm(n = 1, mean = mu2 + rho \* (sigma1 / sigma2) \* (x1 - mu1),  
 sd = sqrt((sigma2^2) \* (1 - rho^2)))  
}  
  
gibbs\_sampling <- function(x1\_init, x2\_init, mu1\_init = 0, mu2\_init = 0,  
 sigma1\_init = 1, sigma2\_init = 1, rho\_init, iterations = 1e5) {  
 sampling\_matrix <- matrix(NA, nrow = iterations, ncol = 2)  
   
 x1\_t <- x1\_init; x2\_t <- x2\_init; mu1 <- mu1\_init; mu2 <- mu2\_init  
 sigma1 <- sigma1\_init; sigma2 <- sigma2\_init; rho <- rho\_init  
   
 for (i in 1:iterations) {  
 x1\_star <- g1(x2 = x2\_t, mu1 = mu1, mu2 = mu2,  
 sigma1 = sigma1, sigma2 = sigma2, rho = rho)  
 x2\_star <- g2(x1 = x1\_star, mu1 = mu1, mu2 = mu2,  
 sigma1 = sigma1, sigma2 = sigma2, rho = rho)  
 sampling\_matrix[i,] <- c(x1\_star, x2\_star)  
 }  
 return(sampling\_matrix)   
}  
  
rho0\_sample <- gibbs\_sampling(x1\_init = 0, x2\_init = 0, rho\_init = 0)  
rho1\_sample <- gibbs\_sampling(x1\_init = 0, x2\_init = 0, rho\_init = 0.1)  
rho2\_sample <- gibbs\_sampling(x1\_init = 0, x2\_init = 0, rho\_init = 0.2)  
rho3\_sample <- gibbs\_sampling(x1\_init = 0, x2\_init = 0, rho\_init = 0.3)  
rho4\_sample <- gibbs\_sampling(x1\_init = 0, x2\_init = 0, rho\_init = 0.4)  
rho5\_sample <- gibbs\_sampling(x1\_init = 0, x2\_init = 0, rho\_init = 0.5)  
  
# Sampling Paths  
burn\_in1 <- 100  
burn\_in2 <- 500  
burn\_in3 <- 1e3  
burn\_in4 <- 5e3  
burn\_in5 <- 1e4  
burn\_in6 <- 7e3  
# Reference: https://www.r-bloggers.com/setting-graph-margins-in-r-using-the-par-function-and-lots-of-cow-milk/  
par(mfrow = c(3,2), mar=c(5,6,4,2) + 0.1)  
plot(burn\_in1:1e5, rho0\_sample[burn\_in1:1e5,1],  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
title(main = TeX('$\\rho = 0$, burn-in: 100'))  
plot(burn\_in1:1e5, rho0\_sample[burn\_in1:1e5,2],  
 type = 'l', ylab = TeX('$x\_2^{(t)}$'), xlab = TeX('t'))  
cs1 <- cusum(theta\_hat = mean(rho0\_sample[burn\_in1:1e5,1]),  
 mcdata = rho0\_sample[burn\_in1:1e5,1], burn = burn\_in1)  
cs2 <- cusum(theta\_hat = mean(rho0\_sample[burn\_in1:1e5,2]),  
 mcdata = rho0\_sample[burn\_in1:1e5,2], burn = burn\_in1)  
plot(1:nrow(cs1), cs1, main = TeX('Cusum $x\_1$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
plot(1:nrow(cs2), cs2, main = TeX('Cusum $x\_2$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
lags <- 40  
autocorrelation(mcdata = rho0\_sample[burn\_in1:1e5,1], lag\_i = lags)  
autocorrelation(mcdata = rho0\_sample[burn\_in1:1e5,2], lag\_i = lags)  
  
plot(burn\_in2:1e5, rho1\_sample[burn\_in2:1e5,1],  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
title(main = TeX('$\\rho = 0.1$, burn-in: 500'))  
plot(burn\_in2:1e5, rho1\_sample[burn\_in2:1e5,2],  
 type = 'l', ylab = TeX('$x\_2^{(t)}$'), xlab = TeX('t'))  
cs3 <- cusum(theta\_hat = mean(rho1\_sample[burn\_in2:1e5,1]),  
 mcdata = rho1\_sample[burn\_in2:1e5,1], burn = burn\_in2)  
cs4 <- cusum(theta\_hat = mean(rho1\_sample[burn\_in2:1e5,2]),  
 mcdata = rho1\_sample[burn\_in2:1e5,2], burn = burn\_in2)  
plot(1:nrow(cs3), cs3, main = TeX('Cusum $x\_1$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
plot(1:nrow(cs4), cs4, main = TeX('Cusum $x\_2$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
autocorrelation(mcdata = rho1\_sample[burn\_in2:1e5,1], lag\_i = lags)  
autocorrelation(mcdata = rho1\_sample[burn\_in2:1e5,2], lag\_i = lags)  
  
plot(burn\_in3:1e5, rho2\_sample[burn\_in3:1e5,1],  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
title(main = TeX('$\\rho = 0.2$, burn-in: 1,000'))  
plot(burn\_in3:1e5, rho2\_sample[burn\_in3:1e5,2],  
 type = 'l', ylab = TeX('$x\_2^{(t)}$'), xlab = TeX('t'))  
cs5 <- cusum(theta\_hat = mean(rho2\_sample[burn\_in3:1e5,1]),  
 mcdata = rho2\_sample[burn\_in3:1e5,1], burn = burn\_in3)  
cs6 <- cusum(theta\_hat = mean(rho2\_sample[burn\_in3:1e5,2]),  
 mcdata = rho2\_sample[burn\_in3:1e5,2], burn = burn\_in3)  
plot(1:nrow(cs5), cs5, main = TeX('Cusum $x\_1$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
plot(1:nrow(cs6), cs6, main = TeX('Cusum $x\_2$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
autocorrelation(mcdata = rho2\_sample[burn\_in3:1e5,1], lag\_i = lags)  
autocorrelation(mcdata = rho2\_sample[burn\_in3:1e5,2], lag\_i = lags)  
  
plot(burn\_in4:1e5, rho3\_sample[burn\_in4:1e5,1],  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
title(main = TeX('$\\rho = 0.3$, burn-in: 5,000'))  
plot(burn\_in4:1e5, rho3\_sample[burn\_in4:1e5,2],  
 type = 'l', ylab = TeX('$x\_2^{(t)}$'), xlab = TeX('t'))  
cs7 <- cusum(theta\_hat = mean(rho3\_sample[burn\_in4:1e5,1]),  
 mcdata = rho3\_sample[burn\_in4:1e5,1], burn = burn\_in4)  
cs8 <- cusum(theta\_hat = mean(rho3\_sample[burn\_in4:1e5,2]),  
 mcdata = rho3\_sample[burn\_in4:1e5,2], burn = burn\_in4)  
plot(1:nrow(cs7), cs7, main = TeX('Cusum $x\_1$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
plot(1:nrow(cs8), cs8, main = TeX('Cusum $x\_2$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
autocorrelation(mcdata = rho3\_sample[burn\_in4:1e5,1], lag\_i = lags)  
autocorrelation(mcdata = rho3\_sample[burn\_in4:1e5,2], lag\_i = lags)  
  
plot(burn\_in5:1e5, rho4\_sample[burn\_in5:1e5,1],  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
title(main = TeX('$\\rho = 0.4$, burn-in: 10,000'))  
plot(burn\_in5:1e5, rho4\_sample[burn\_in5:1e5,2],  
 type = 'l', ylab = TeX('$x\_2^{(t)}$'), xlab = TeX('t'))  
cs9 <- cusum(theta\_hat = mean(rho4\_sample[burn\_in5:1e5,1]),  
 mcdata = rho4\_sample[burn\_in5:1e5,1], burn = burn\_in5)  
cs10 <- cusum(theta\_hat = mean(rho4\_sample[burn\_in5:1e5,2]),  
 mcdata = rho4\_sample[burn\_in5:1e5,2], burn = burn\_in5)  
plot(1:nrow(cs9), cs9, main = TeX('Cusum $x\_1$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
plot(1:nrow(cs10), cs10, main = TeX('Cusum $x\_2$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
autocorrelation(mcdata = rho4\_sample[burn\_in5:1e5,1], lag\_i = lags)  
autocorrelation(mcdata = rho4\_sample[burn\_in5:1e5,2], lag\_i = lags)  
  
plot(burn\_in6:1e5, rho5\_sample[burn\_in6:1e5,1],  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
title(main = TeX('$\\rho = 0.5$, burn-in: 7,000'))  
plot(burn\_in6:1e5, rho5\_sample[burn\_in6:1e5,2],  
 type = 'l', ylab = TeX('$x\_2^{(t)}$'), xlab = TeX('t'))  
cs11 <- cusum(theta\_hat = mean(rho5\_sample[burn\_in6:1e5,1]),  
 mcdata = rho5\_sample[burn\_in6:1e5,1], burn = burn\_in6)  
cs12 <- cusum(theta\_hat = mean(rho5\_sample[burn\_in6:1e5,2]),  
 mcdata = rho5\_sample[burn\_in6:1e5,2], burn = burn\_in6)  
plot(1:nrow(cs11), cs11, main = TeX('Cusum $x\_1$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
plot(1:nrow(cs12), cs12, main = TeX('Cusum $x\_2$'),  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
autocorrelation(mcdata = rho5\_sample[burn\_in6:1e5,1], lag\_i = lags)  
autocorrelation(mcdata = rho5\_sample[burn\_in6:1e5,2], lag\_i = lags)  
  
### Problem 3  
# part (a)  
mcdata1 <- scan(file.choose())  
mcdata2 <- scan(file.choose())  
  
par(mfrow = c(2,1), mar=c(5,6,4,2) + 0.1)  
plot(1:length(mcdata1), mcdata1, main = 'MCdata1',  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
plot(1:length(mcdata2), mcdata2, main = 'MCdata2',  
 type = 'l', ylab = TeX('$x\_1^{(t)}$'), xlab = TeX('t'))  
dev.off()  
  
# part (b)  
burn\_in1 <- 5001  
burn\_in2 <- 10001  
theta\_hat1 <- mean(mcdata1[burn\_in1:length(mcdata1)])  
theta\_hat2 <- mean(mcdata2[burn\_in2:length(mcdata2)])  
  
cusum <- function(theta\_hat=theta\_hat1, mcdata=mcdata1, burn=burn\_in) {  
 cusum\_data <- matrix(NA, nrow = length(mcdata) - burn + 1)  
 cusum\_diff <- matrix(NA, nrow = length(mcdata) - burn + 1)  
 mcdata <- mcdata[burn:length(mcdata)]  
 for (i in 1:length(mcdata)) {  
 cusum\_diff[i] <- mcdata[i] - theta\_hat  
 cusum\_data[i] <- sum(cusum\_diff, na.rm = TRUE)  
 }  
 return(cusum\_data)  
}  
  
cs\_1 <- cusum(theta\_hat = theta\_hat1, mcdata = mcdata1, burn = burn\_in1)  
cs\_2 <- cusum(theta\_hat = theta\_hat2, mcdata = mcdata2, burn = burn\_in2)  
  
par(mfrow = c(2,1))  
plot(1:nrow(cs\_1), cs\_1, main = 'Cusum MCdata1',  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
plot(1:nrow(cs\_2), cs\_2, main = 'Cusum MCdata2',  
 type = 'l', ylab = 'Cusum', xlab = TeX('t'))  
dev.off()  
  
# part (c)  
C\_i <- function(mcdata=mcdata1, i=40) {  
 n <- length(mcdata)  
 xbar <- mean(mcdata)  
 autocovariance\_matrix <- matrix(NA, nrow = (n - i))  
 for (t in 1:(n - i)) {  
 autocovariance\_matrix[t] <- (mcdata[t] - xbar) \*  
 (mcdata[t + i] - xbar)  
 }  
 mean(autocovariance\_matrix)  
}  
  
C\_0 <- function(mcdata=mcdata1) {  
 xbar <- mean(mcdata)  
 mean((mcdata - xbar)^2)  
}  
  
autocorrelation <- function(mcdata=mcdata1, lag\_i=40) {  
 c\_0 <- C\_0(mcdata = mcdata)  
 acf\_matrix <- matrix(NA, nrow = lag\_i)  
 for (i in 1:lag\_i) {  
 c\_i <- C\_i(mcdata=mcdata, i=i)  
 acf\_matrix[i] <- c\_i / c\_0  
 }  
 plot(1:lag\_i, acf\_matrix, ylim = c(0,1),  
 pch = 19, xlab = 'Lags', ylab = 'acf')  
 segments(x0 = 1:lag\_i, y0 = 0, x1 = 1:lag\_i, y1 = acf\_matrix)  
 abline(h = 0)  
}  
  
lags <- 40  
par(mfrow = c(2,1))  
acf1 <- autocorrelation(mcdata = mcdata1, lag\_i = lags)  
title('ACF of MCdata1')  
acf2 <- autocorrelation(mcdata = mcdata2, lag\_i = lags)  
title('ACF of MCdata2')  
dev.off()  
  
### Problem 4  
# part (a)  
mc1 <- scan(file.choose())  
mc2 <- scan(file.choose())  
mc3 <- scan(file.choose())  
mc4 <- scan(file.choose())  
mc5 <- scan(file.choose())  
mc6 <- scan(file.choose())  
mc7 <- scan(file.choose())  
  
gelman\_rubin <- function(D=0, L=1000, first\_part=1000) {  
 J <- 7  
 mc <- cbind(mc1, mc2, mc3, mc4, mc5, mc6, mc7)  
 mc <- mc[1:first\_part,]  
 mc <- mc[(D+1):(D+L),]  
  
 xbar\_j <- (1 / L) \* colSums(mc)  
 xbar <- mean(xbar\_j)  
  
 B <- (L / (J - 1)) \* sum((xbar\_j - xbar)^2)  
 sj2 <- (1 / (L - 1)) \* apply(matrix(1:J), 1, function(x) {  
 sum((mc[,x] - xbar\_j[x])^2)  
 })  
 W <- mean(sj2)  
 R <- (((L - 1) / L) \* W + (1 / L) \* B) / W  
 return(c(B, W, sqrt(R)))  
}  
  
# part(a-f)  
gelman\_rubin(D = 0, L = 1000)  
gelman\_rubin(D = 500, L = 500)  
gelman\_rubin(D = 0, L = 500, first\_part = 500)  
gelman\_rubin(D = 250, L = 250, first\_part = 500)  
gelman\_rubin(D = 0, L = 50, first\_part = 50)  
gelman\_rubin(D = 25, L = 25, first\_part = 50)