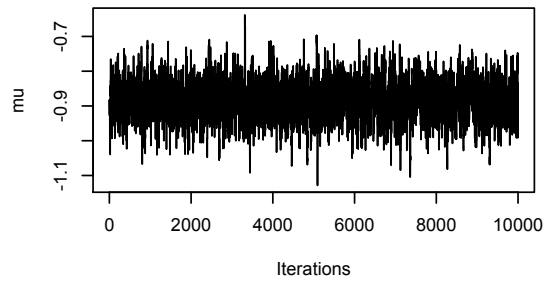
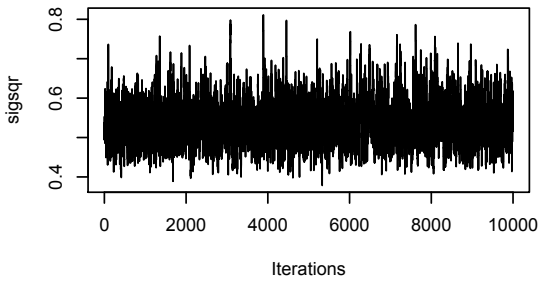
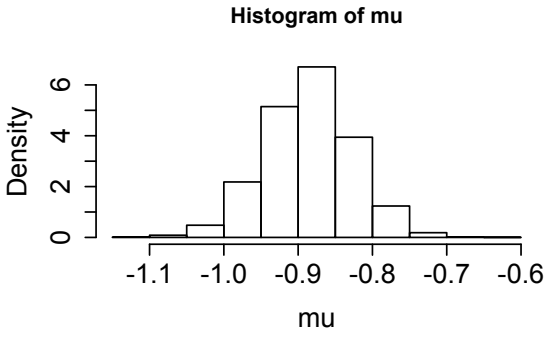
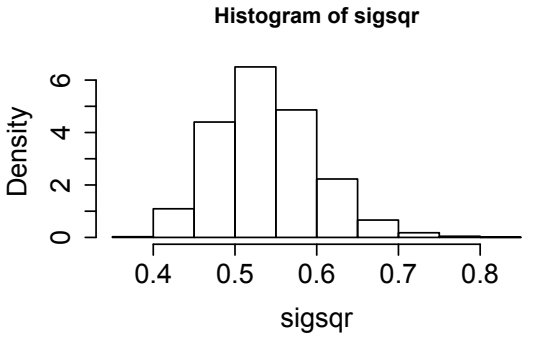
Lisa Over

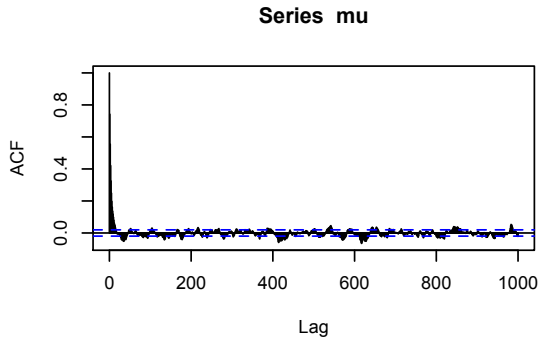
Homework 8 Appendix (Part 2)

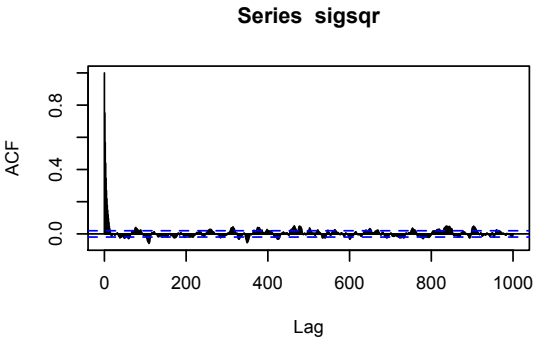












lag 50
b=0.07
c=0.07

acceptance probability for mu = 0.6532
acceptance probability for sigsqr = 0.6593

Lisa Over

CODE

```
metro_hastings <- function(data,m0,s0,b,c,N,lag,burnin) {

	ybar = mean(data)
	ssqr = var(data)
	n = length(data)

	N <- N*lag + burnin

	m.s = NULL
	s.s = NULL

	mu.cnt = 0
	sig.cnt = 0

	for(i in 1:N) {

		mstar = rnorm(1,m0,b)
		mratio = exp(-(1/(2*s0))*(n*(ybar-mstar)^2 + (mstar^2)/10 -
n*(ybar-m0)^2 - m0^2/10))
		accept = FALSE
		if(runif(1,0,1) < mratio) {
			accept = TRUE
		}
		if(i > burnin) {
			if(i %% lag == 0) {
				if(accept) {
					mu.cnt = mu.cnt + 1
					m.s = c(m.s, mstar)
					m0 = mstar
				}
				else {
					m.s = c(m.s, m0)
					m0 = m0
				}
			}
		}
		sstar = abs(rnorm(1,s0,c))
		#restrict sstar to be greater than 0.0001 so astar does not go to
infinity
		while(sstar < 0.0001) {
			sstar = abs(rnorm(1,s0,c))
		}
		#print("sstar")
		#print(sstar)
```

```
                astar = (1/sstar)^((n/2)+4)
                #print("astar")
                #print(astar)
                bstar = (1/(2*sstar))*((n - 1)*ssqr + n*(ybar-m0)^2 + (m0^2)/10 + 2)
                azero = (1/s0)^((n/2)+4)
                bzero = (1/(2*s0))*((n - 1)*ssqr + n*(ybar-m0)^2 + (m0^2)/10 + 2)
                sratio = exp(log(astar) - bstar - log(azero) + bzero)

                dstar = dnorm(s0, sstar, c)/(1-pnorm(0, sstar, c))
                dzero = dnorm(sstar, s0, c)/(1-pnorm(0, s0, c))
                hratio = dstar/dzero

                accept = FALSE
                if(runif(1,0,1) < sratio*hratio) {
                        accept = TRUE
                }
                if(i > burnin) {
                        if(i %% lag == 0) {
                                if(accept) {
                                        sig.cnt = sig.cnt + 1
                                        s.s = c(s.s, sstar)
                                        s0 = sstar
                                }
                                else {
                                        s.s = c(s.s, s0)
                                        s0 = s0
                                }
                        }
                }
        }
        vectors <- list("mu" = m.s, "mcount" = mu.cnt, "sigsqr" = s.s, "scount" =
sig.cnt)
 return(vectors)
}


data2 = read.table(file.choose(), header=F)
attach(data2)
data2 = data2$V1

#Set burnin, lag, m0 of data, and s0; run metro to obtain N realizations
N = 10000
lag = 50
burnin = 0
b = .07
c = .07
```

Lisa Over

```
m0 = mean(data2)
s0 = var(data2)

vectors = metro_hastings(data2,m0,s0,b,c,N,lag,burnin)

mu = vectors$mu
sigsqr = vectors$sigsqr
mcount = vectors$mcount
scount = vectors$scount

mcount/N
scount/N


#alpha = length(data2)/2 + 3.5
#lambda = ((length(data2)-1)*var(data2))/2 + 2
alpha = length(data2)/2 - 1
lambda = ((length(data2)-1)*var(data2))/2

par(mfrow=c(3,2)) #split plotting window into 3 rows and 2 columns
ts.plot(mu,xlab="Iterations")
ts.plot(sigsqr,xlab="Iterations")
hist(mu,probability=T, cex.lab=1.5, cex.axis=1.5)
hist(sigsqr,probability=T, cex.lab=1.5, cex.axis=1.5)
yy = seq(10,70,by=.001)
lines(yy, ((lambda^alpha)/gamma(alpha))*((1/yy)^(alpha+1))*exp(-lambda/yy))
acf(mu, lag.max=1000)
acf(sigsqr, lag.max=1000)
```