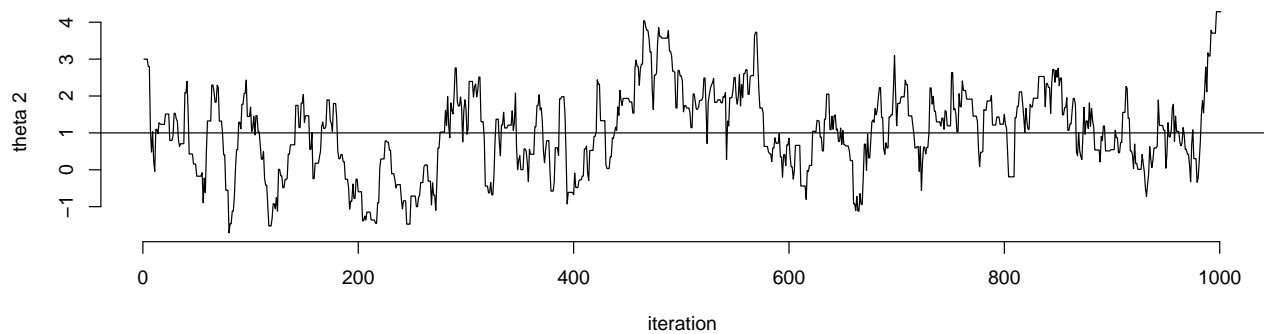
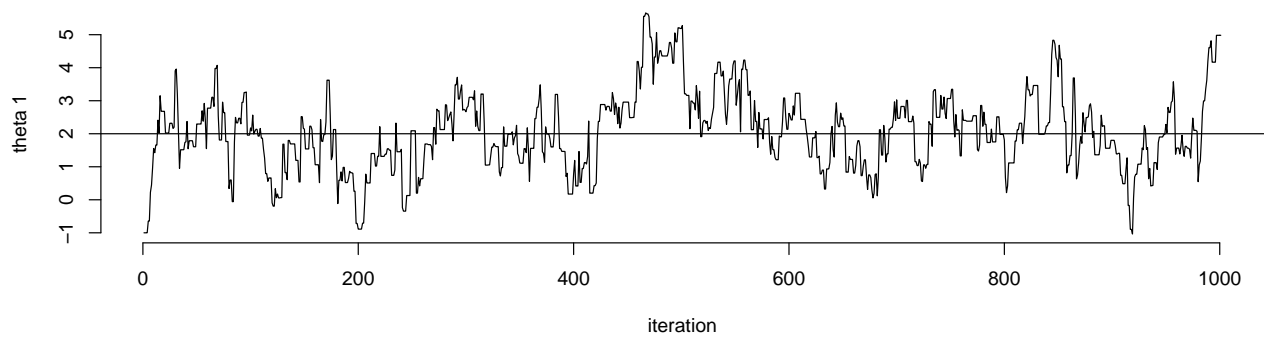


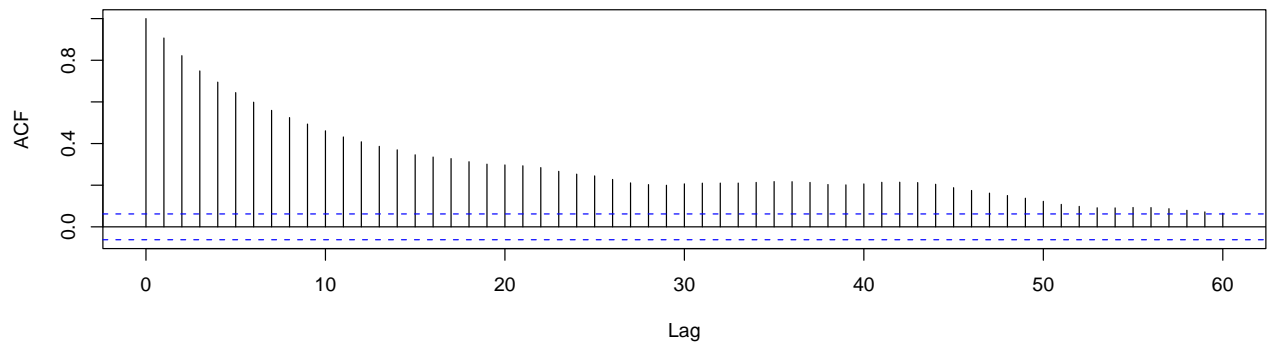
Metropolis-Hastings example. Bivariate Normal Distribution

- Picture shows iteration vs values of θ_1 and iteration vs values of θ_2 . Horizontal lines are located at μ_1 and μ_2 respectively.

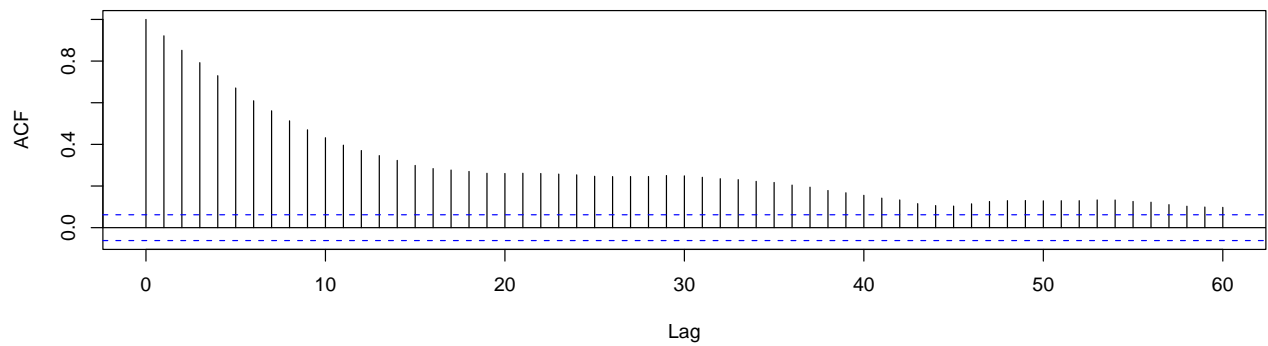


- Autocorrelation plots for values of θ_1 and values of θ_2 .

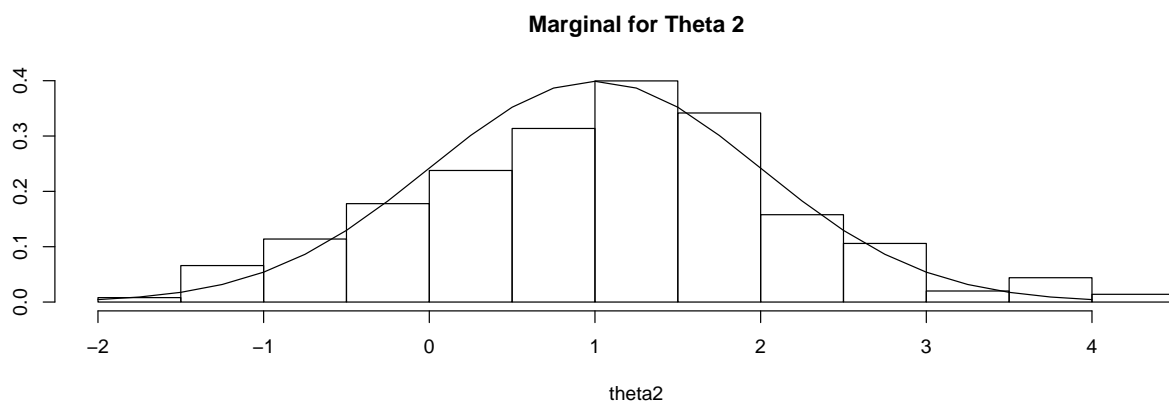
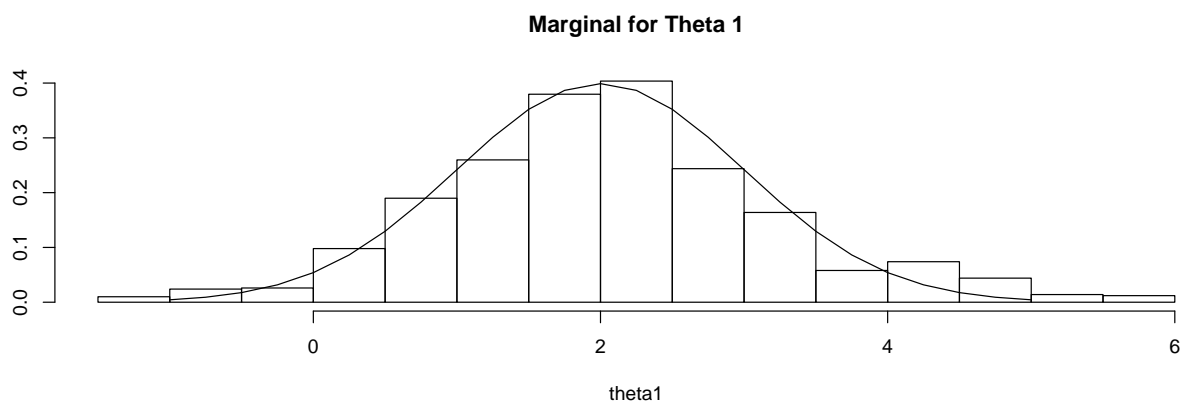
Theta 1



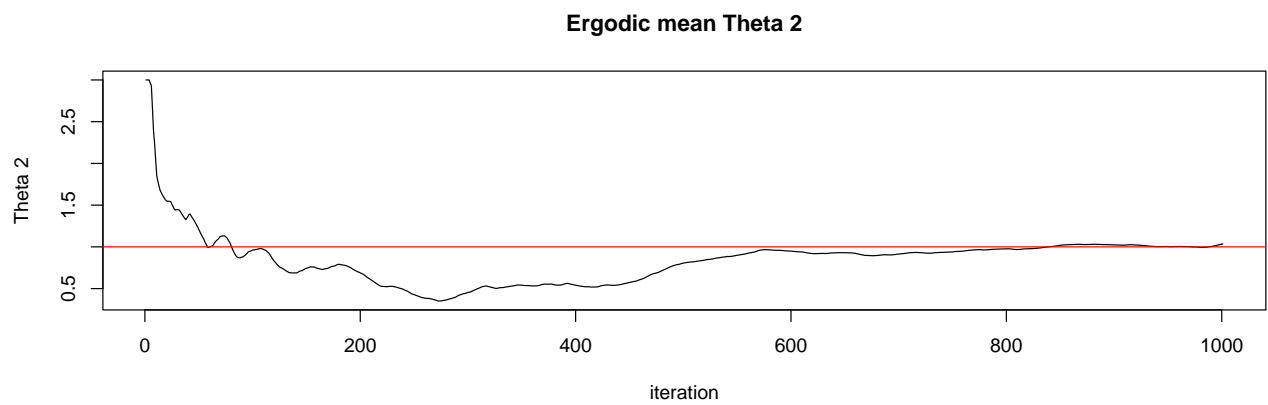
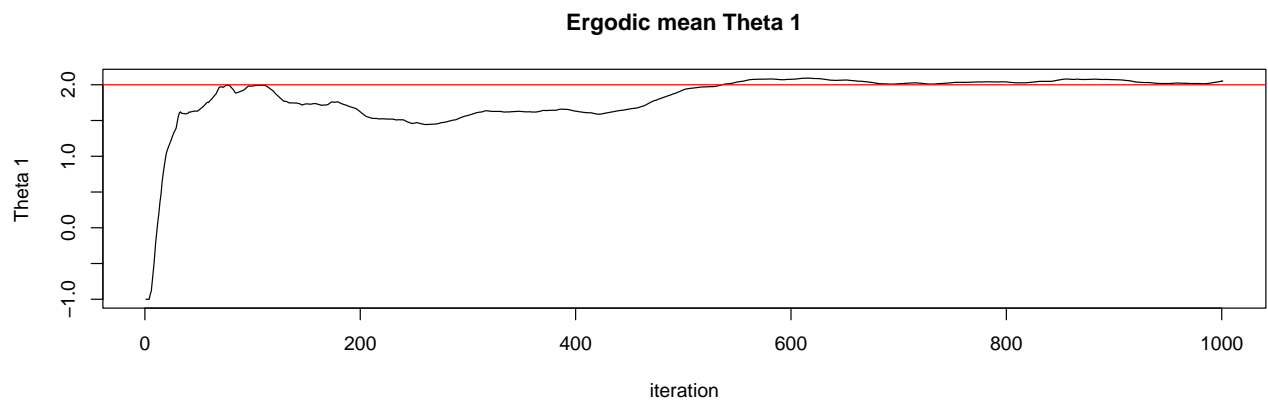
Theta 2



- Histograms for samples of θ_1 and θ_2 (no burn-in). Curves are the actual pdfs corresponding to the marginals.

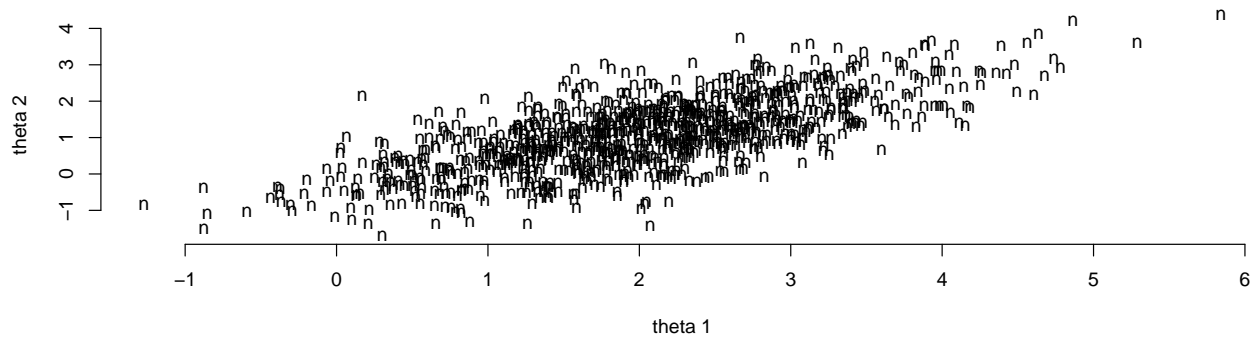


- These are ergodic means for samples of θ_1 and θ_2 . Horizontal lines correspond to the true means.

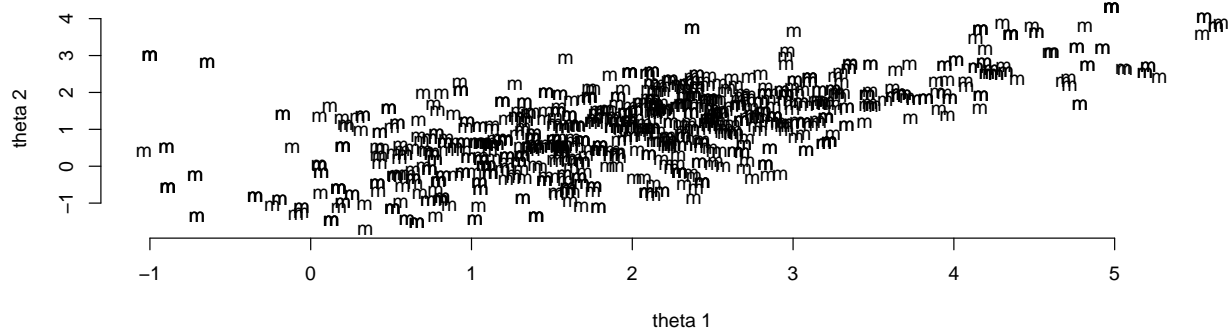


- Comparison of simulated values. Upper plot are the simulations coming directly from the Bivariate Normal. Lower plot are the simulations obtained with the Metropolis-Hastings.

1000 simulations from the Bivariate Normal



1000 simulations from the M-H algorithm



R-code for Metropolis-Hastings example

```
# Sta 590 Statistical Computing
# R code for Metropolis-Hastings example
# Target distribution is a Bivariate Normal
#
# Function to evaluate Multivariate Normal
dnormm<-function(x,m,P){
  exp(-0.5*(t(x-m)%*%P%*(x-m)))
}
# Mean vector and cov. target
m<-c(2,1)
S<-matrix(c(1,0.7,0.7,1),2,2)
iS<-solve(S)
cS<-t(chol(S))
#
# Simulations from target dist.
y<-matrix(NA,1000,2)
for(i in 1:1000)
{
  y[i,]<-m+cS%*%rnorm(2)
}
# Mean vector and cov. proposal
V<-diag(c(.6,.4))
cV<-t(chol(V))
iV<-solve(V)
```

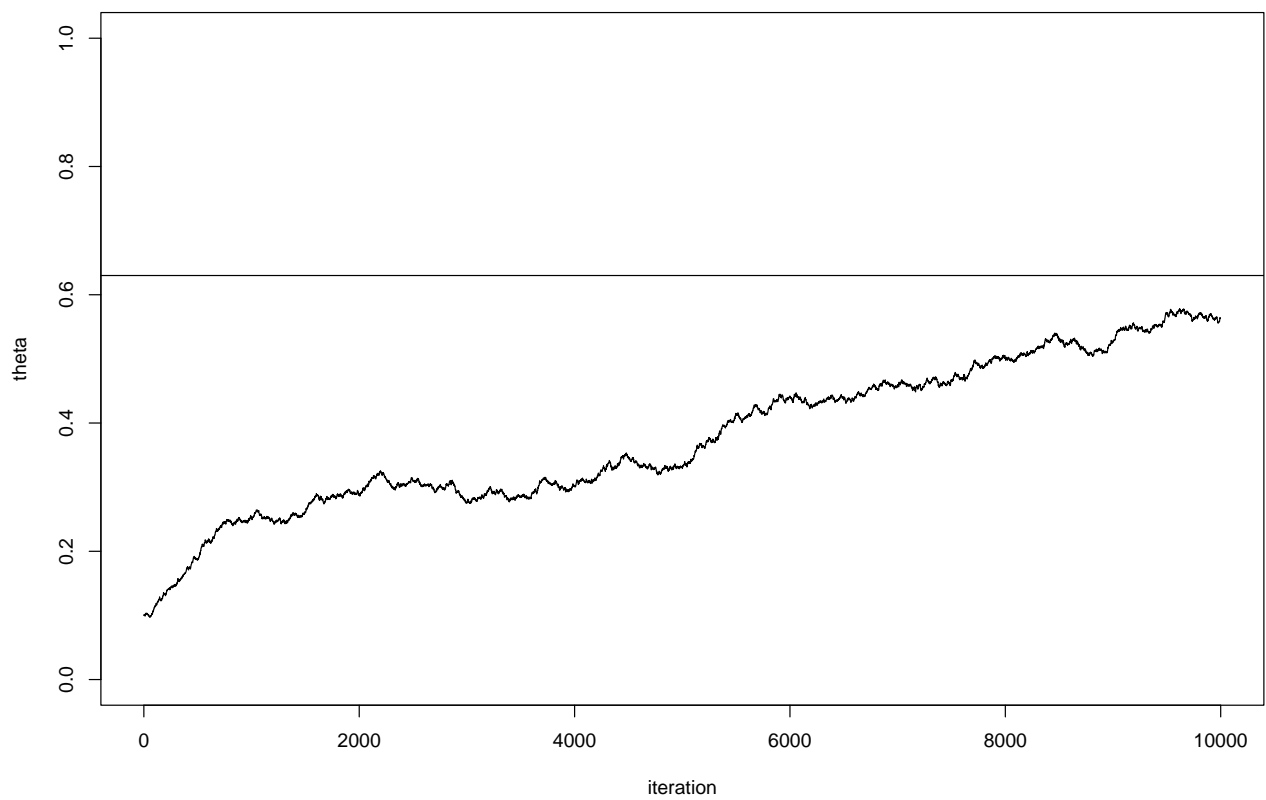
```

# No. of iterations and initial value
M<-1000
th <-c(-1,3)
xs<-th
accepted<-0
# Metropolis-Hastings iterations
for (i in 1:M)
{
print(i)
thnew<-th+cV%*%rnorm(2)
acceptance<-min(c(1,dnormm(thnew,m,iS)/dnormm(th,m,iS)))
u<-runif(1)
if (u<acceptance){
  th<-thnew
  accepted<-accepted+1
}
  xs<-c(xs,th)
}
xs<-matrix(xs,(M+1),2,byrow=T)
# some results
> accepted
[1] 601
> mean(xs[100:1000,1])
[1] 2.058023
> mean(xs[100:1000,2])
[1] 1.041193
> quantile(xs[1:1000,1],probs=c(.25,.75))
      25%      75%
1.305534 2.713041
> qnorm(c(.25,.75),mean=m[1],sd=sqrt(S[1,1]))
[1] 1.325510 2.674490

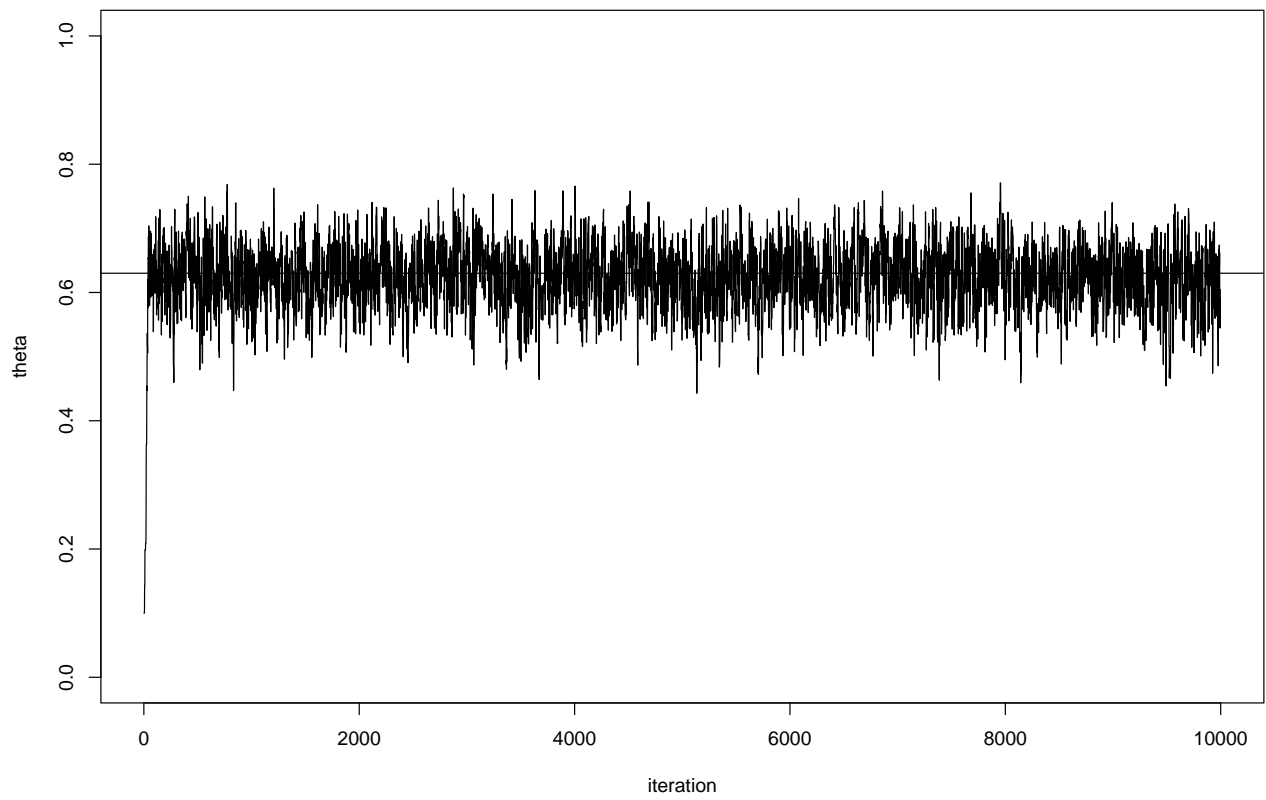
```

Metropolis-Hastings for Genetic Linkeage Example

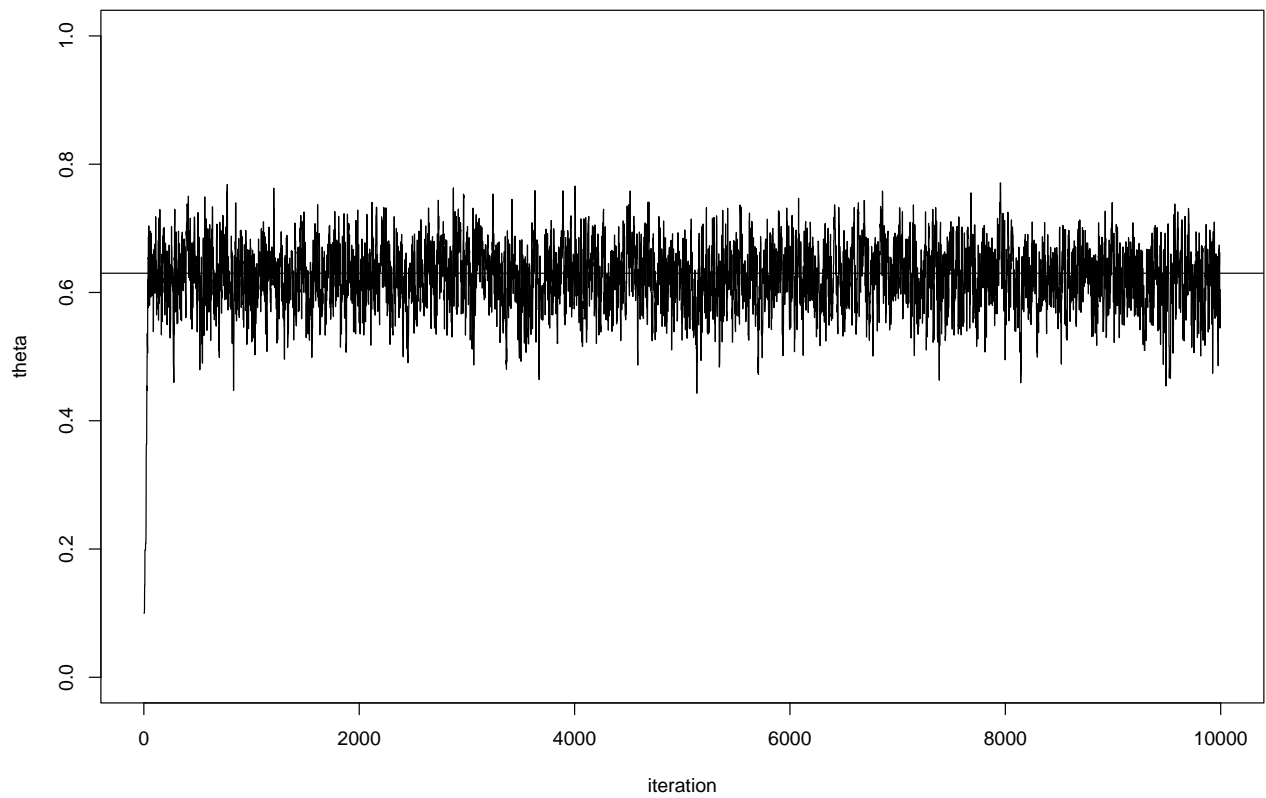
- These are 10000 draws from the M-H algorithm using a value of $\sigma = 0.001$ for the proposal density. Vertical line is located at the posterior mean.



- Now we have 10000 draws from the M-H algorithm using a value of $\sigma = 0.005$



- These are 10000 draws from the M-H algorithm using a value of $\sigma = 0.12$. The acceptance rate in this case is 0.45.



Code for the M-H. Genetic linkage example

```
# Metropolis-Hastings Example for genetic Linkeage
# Random Walk proposal
post <- function(th,y)
{
  pst <- ( (2+th)^y[1])*( (1-th)^{y[2]+y[3]})*(th^y[4])
  return(pst)
}
y <- c(125,18,20,34)
accept <- 0
M <- 10000
th <- 0.1
xs <- th
for (i in 1:M)
{
  thnew<-th+rnorm(1,mean=0,sd=0.12)
  acceptance<-min(c(1,post(thnew,y)/post(th,y)))
  u<-runif(1)
  if (u<acceptance){
    th<-thnew
    accept <-accept+1
  }
  xs<-c(xs,th)
}
```