統算HW10

統碩一 109225017 林心玹

# 1

N=1000, to get posterior samples:

Algorithm:

Step1: Generate

Step2: Given initial

Step3: Generate ,

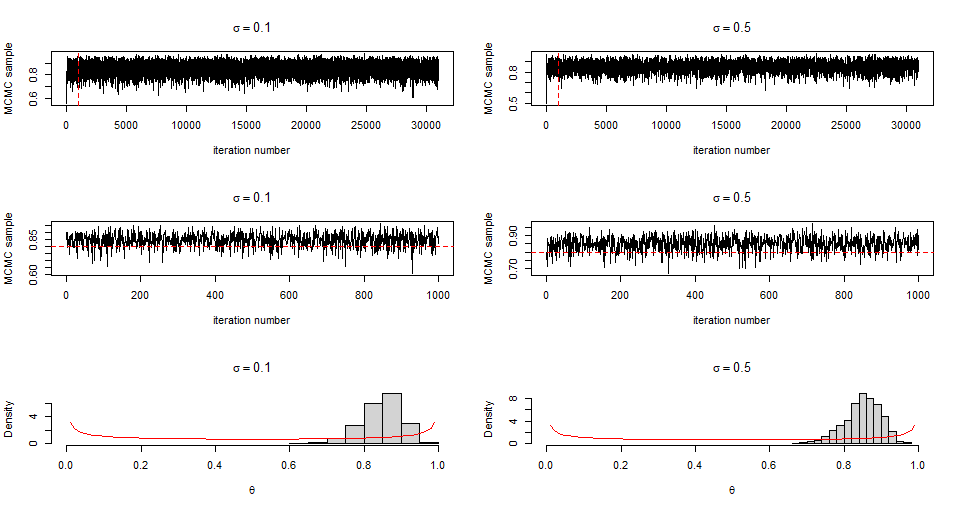
Step4: Generate

Step5:

If ,

Step6: n=n+1, , return to Step2 times

posterior <- function(theta,y){(theta^(y-0.5))\*((1-theta)^(50-y-0.5))}  
f <- function(m=1000,N=1000,k=30,theta0=0.5,sigma){  
 a <-c();theta <- c()  
 y0 <- rbinom(1,50,0.8)  
 u <- runif(m+N\*k)  
 for(i in 1:(m+N\*k)){  
 repeat{  
 z <- rnorm(1,theta0,sigma)  
 if(z>0 & z<1){break}  
 }  
 a[i] <- alpha <- min(posterior(z,y0)/posterior(theta0,y0),1)  
 if(alpha>u[i]){  
 theta[i] <- z  
 }else{  
 theta[i] <- theta0  
 }  
 theta0 <- theta[i]  
 }  
 return(list(a,u,theta))  
}  
set.seed(109225017)  
result1 <- f(sigma=0.1)  
m=1000;N=1000;k=30  
result1.theta <- result1[[3]][seq(m+k,m+N\*k,k)]  
table <- matrix(c(mean(result1.theta),var(result1.theta),sum(result1[[1]]>result1[[2]])/(m+N\*k)),nrow=1)  
set.seed(109225017)  
result2 <- f(sigma=0.5)  
result2.theta <- result2[[3]][seq(m+k,m+N\*k,k)]  
table <- rbind(table,c(mean(result2.theta),var(result2.theta),sum(result2[[1]]>result2[[2]])/(m+N\*k)))  
colnames(table) <- c("E(theta|y)","Var(theta|y)","AR")  
rownames(table) <- c("sigma=0.1","sigma=0.5")  
table <- as.data.frame(table)  
par(mfrow=c(3,2))  
plot(1:(m+N\*k),result1[[3]],type="l",xlab = "iteration number",ylab="MCMC sample",main=expression  
 (sigma==0.1))  
abline(v=m,col="red",lty=2)  
  
plot(1:(m+N\*k),result2[[3]],type="l",xlab = "iteration number",ylab="MCMC sample",main=expression  
 (sigma==0.5))  
abline(v=m,col="red",lty=2)  
  
plot(1:N,result1.theta,type="l",xlab = "iteration number",ylab="MCMC sample",main=expression(sigma==0.1))  
abline(h=0.8,col="red",lty=2)  
  
plot(1:N,result2.theta,type="l",xlab = "iteration number",ylab="MCMC sample",main=expression(sigma==0.5))  
abline(h=0.8,col="red",lty=2)  
  
hist(result1.theta,freq = F,xlim=c(0,1),xlab=expression(theta),main=expression(sigma==0.1))  
curve(dbeta(x,0.5,0.5),add=T,col="red")  
  
hist(result2.theta,freq = F,xlim=c(0,1),xlab=expression(theta),main=expression(sigma==0.5))  
curve(dbeta(x,0.5,0.5),add=T,col="red")



|  |  |  |  |
| --- | --- | --- | --- |
|  | E(theta|y) | Var(theta|y) | AR |
| sigma=0.1 | 0.8480974 | 0.0027426 | 0.5382903 |
| sigma=0.5 | 0.8502182 | 0.0022889 | 0.2091613 |

由上表可知，𝜎=0.1和𝜎=0.5的E(theta|y) 都接近0.8，但當𝜎=0.1時，AR明顯高於𝜎=0.5， 因為後驗分配是在有樣本下對先驗分配的修正,但先驗分配假設過於主觀(變異數很小，theta會收斂至其期望值)造成與後驗分配差異大。

# 2

Generate a sample of size m=1000 from

m=1000=N;k=30

Algorithm:

Step1: Specify

Step2: Generate

Step3: n=1

Step4:

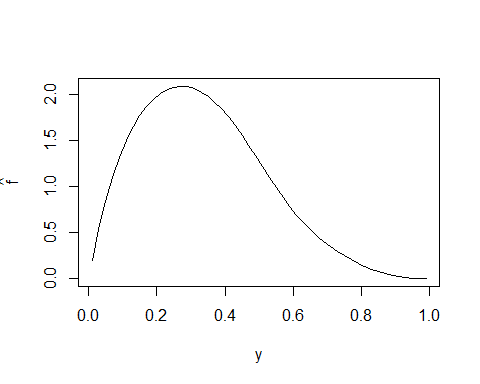
Step5:

Step6: n=n+1, go to Step4 after m+N\*k itrations

Step7: Set .

E(Y), Var(Y),

#2  
y0 <- 0.5  
N = m = 1000;k <- 30  
x <-c(); y <- c()  
set.seed(109225017)  
for(i in 1:(m+N\*k)){  
 x[i] <- rbinom(1,20,y0)  
 y[i] <- rbeta(1,x[i]+2,20-x[i]+4)  
 y0 <- y[i]  
}  
result <- y[seq(m+k,m+N\*k,k)]  
xaxis <- seq(0.01,0.99,0.02)  
fyhat <- sapply(xaxis, function(y){  
 mean(dbeta(y,x[seq(m+k,m+N\*k,k)]+2,20-x[seq(m+k,m+N\*k,k)]+4))  
})  
plot(xaxis,fyhat,type = "l",xlab = "y",ylab =expression(hat(f)))



mean(result);var(result)

## [1] 0.3358946

## [1] 0.03205318