36-463/663: Multilevel & Hierarchical Models
HW09 Solution

November 15, 2016

Quesion 1

Following the derivation given in class, when

$$L(\mu) \propto \exp\left\{\frac{n(\bar{x}-\mu)^2}{2\sigma_0^2}\right\}, \quad f(p) \propto \exp\left\{\frac{-(\mu-\mu_0)^2}{2\tau_0^2}\right\},$$

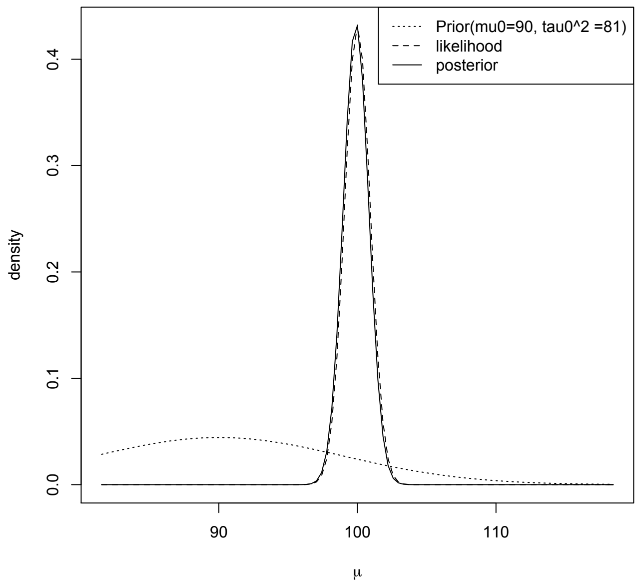
the posterior is also normally distributed with $\mu^* = \left(\frac{\tau_0^2}{\tau_0^2+\sigma_0^2/n}\right)\bar{x} + \left(\frac{\sigma_0^2/n}{\tau_0^2+\sigma_0^2/n}\right)\mu_0, \quad \sigma^{*2} = \frac{1}{n/\sigma_0^2+1/\tau_0^2}$. We will try

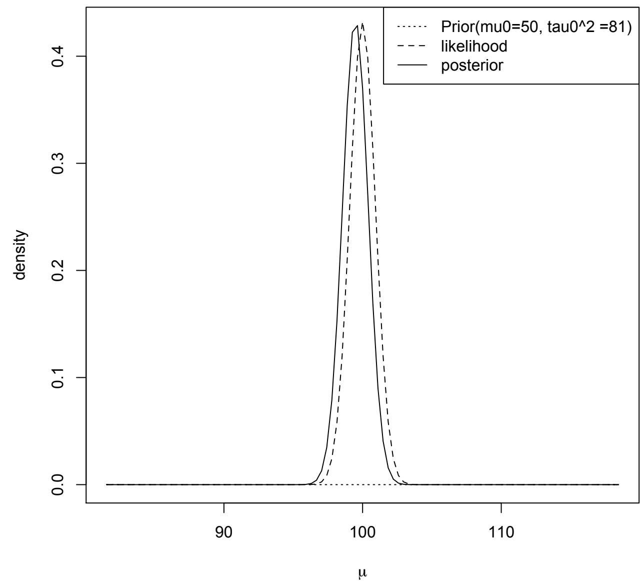
- $\mu_0 = 90, \tau_0^2 = 81$ which results in $\mu^* = 100, \sigma^{*2} = 0.8$
- $\mu_0 = 50, \tau_0^2 = 81$ which results in $\mu^* = 99.5, \sigma^{*2} = 0.8$ and
- $\mu_0 = 50, \tau_0^2 = 4$ which results in $\mu^* = 91, \sigma^{*2} = 0.7$.

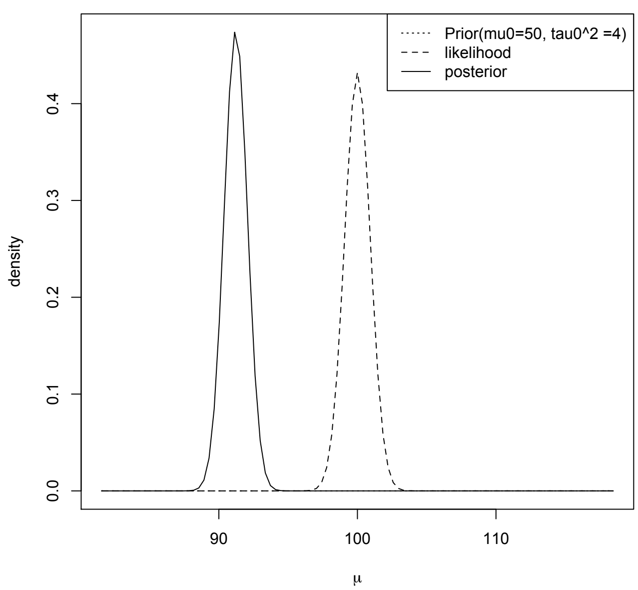
```
xbar <- 100
sigma <- 12
n <-169
lo <- xbar - 20*sigma /sqrt(n)
hi <- xbar + 20*sigma /sqrt(n)

prior <- function(x){dnorm(x,mu0,tau0)}
lik <- function(x){dnorm(x,xbar,sigma/sqrt(n))}
posterior <- function(x){
   rel <- tau0^2 /(tau0^2 + sigma^2/n)
   unrel <- 1 - rel
   mu.n <- rel*xbar + unrel*mu0
   tau.n <- sqrt(1/(n/sigma^2 + 1/tau0^2))
   dnorm(x,mu.n,tau.n)
}
mu0<- 90
tau0 <- 9
curve(posterior(x),from = lo, to = hi, xlab = expression(mu),ylab = "density",lty = 1)
curve(prior(x),add = T,lty =3)
curve(lik(x),add = T,lty =2)
legend("topright",lty = c(3,2,1), legend = c("Prior(mu0=90, tau0^2 =81)","likelihood","posterior"))

#code similar for the other two cases
```







The prior mean tends to pull the posterior mean more away from the sample mean. However the influence of the prior mean is much stronger when the prior variance is small, than when the prior variance is large.

Question 2

```
library(R2jags)
library(rube)
```

```
M2 <- "model {
for (i in 1:n){
x[i] ~ dnorm(mu[gp[i]],sig2inv)
}

for (j in 1:2){
mu[j] ~ dnorm(mu0,tau2inv)
}

mu0 ~ dnorm(0,1)

sig2inv ~dgamma(0.001,0.001)
tau2inv ~dgamma(0.001,0.001)

sig2 <- 1/sig2inv
tau2 <- 1/tau2inv

# test difference directly
mu1.minus.mu2 <- mu[1] - mu[2]

}"

data.list.2 <- list(x = c(1.81,0.52,-0.53,-1.6,-1.74,2.90,0.38,2.76,-0.13),
  gp=c(rep(1,5),rep(2,4)),n=9)

inits.2 <- function(){

list(mu0 = rnorm(1,0,1),sig2inv = rgamma(1,1,1),tau2inv = rgamma(1,1,1))
}

M2.fit <- rube(M2,data.list.2,inits.2,parameters.to.save =
c("mu0","mu","sig2","tau2","mu1.minus.mu2"),n.chains = 3)
M2.fit
p3(M2.fit)

here is the result:
Rube Results:
Run by jags at 2016-11-15 03:05 and taking 0.14 secs
                mean     sd     2.5%     25%    50%      75%  97.5% Rhat n.eff
deviance      35.069  2.305 31.51557 33.6266 34.786 36.20922 40.667 1.00  1000
mu[1]          0.140  0.630 -1.19332 -0.2628  0.195  0.58380  1.357 1.00   520
mu[2]          0.722  0.710 -0.48100  0.2617  0.681  1.07184  2.348 1.00   450
mu0            0.323  0.606 -0.89053 -0.0379  0.356  0.69715  1.588 1.00   910
mu1.minus.mu2 -0.583  0.911 -2.91294 -1.0583 -0.230  0.00696  0.665 1.00  1000
sig2           3.287  2.152  1.04595  1.9853  2.717  3.92567  8.429 1.01   450
tau2           7.932 85.926  0.00122  0.0129  0.168  1.11690 19.659 1.00  1000

DIC = 37.72881
```

Plots from the `p3()` functino begin on p. 5. Most look good, although the MCMC sample for τ^2 tends to wander off occasionally; it appears there is neither enough information in the data, nor in the prior distribution, to give the posterior a good peak for the MCMC to stay close to.

Notice that the CI's from the M2.fit output (on page 3) for μ_1 and μ_2, respectively, are $c(-1.19, 1.36)$ and $c(-0.48, 2.35)$ respectively. Since they overlap, we cannot conclude that the treatment is effective.

Another way to see this is to make a CI for the difference $\mu_1 - \mu_2$ directly. We do this by adding the like

```
mu1.minus.mu2 <- mu[1] - mu[2]
```

to the JAGS program, getting the MCMC sample for this difference, and inspecting the CI for this difference directly. From the M2.fit output, the CI is $(-2.91, 0.665)$; and since this include zero, we cannot conclude that the difference is significantly different from zero; that is, we cannot conclude that the treatment was effective.

Question 3

```
library(R2jags)
library(rube)

M3 <- "model {
for (i in 1:n){
x[i] ~ dpois(lambda)
}

lambda ~ dgamma(1,1)
}"

data.list.3 <- list(x = c(3,7,1,7,6,1,5,4,6,2,11,3,1,6,4,4),n=16)
inits.3 <- function(){

list(lambda = rgamma(1,1,1))
}

M3.fit <- rube(M3,data.list.3,inits.3,parameters.to.save = c("lambda"),n.chains = 3)

M3.fit
p3(M3.fit)

here is the result from the simulation:
Rube Results:
Run by jags at 2016-11-15 03:33 and taking 0.1 secs
          mean    sd  2.5%   25%   50%   75% 97.5% Rhat n.eff
deviance 77.38 1.676 76.19 76.33 76.79 77.66 82.24    1  1000
lambda    4.23 0.507  3.27  3.89  4.19  4.58  5.22    1   570

DIC = 78.78523
```

our point estimate for λ is 4.23, a 95% CI is (3.27, 5.22). The `q3()` plot appears on page 9.

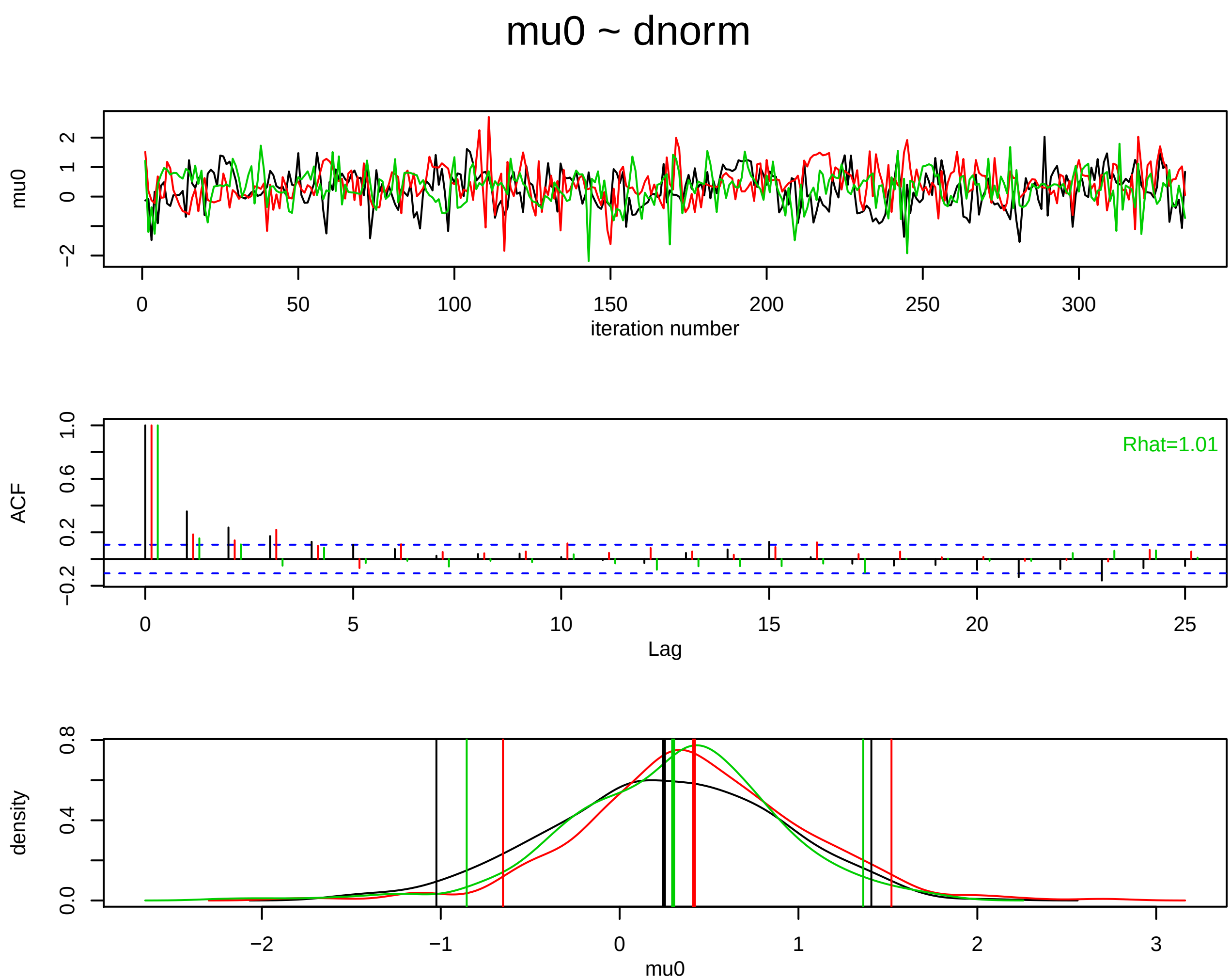


Figure 1: Question 2 plot μ_0

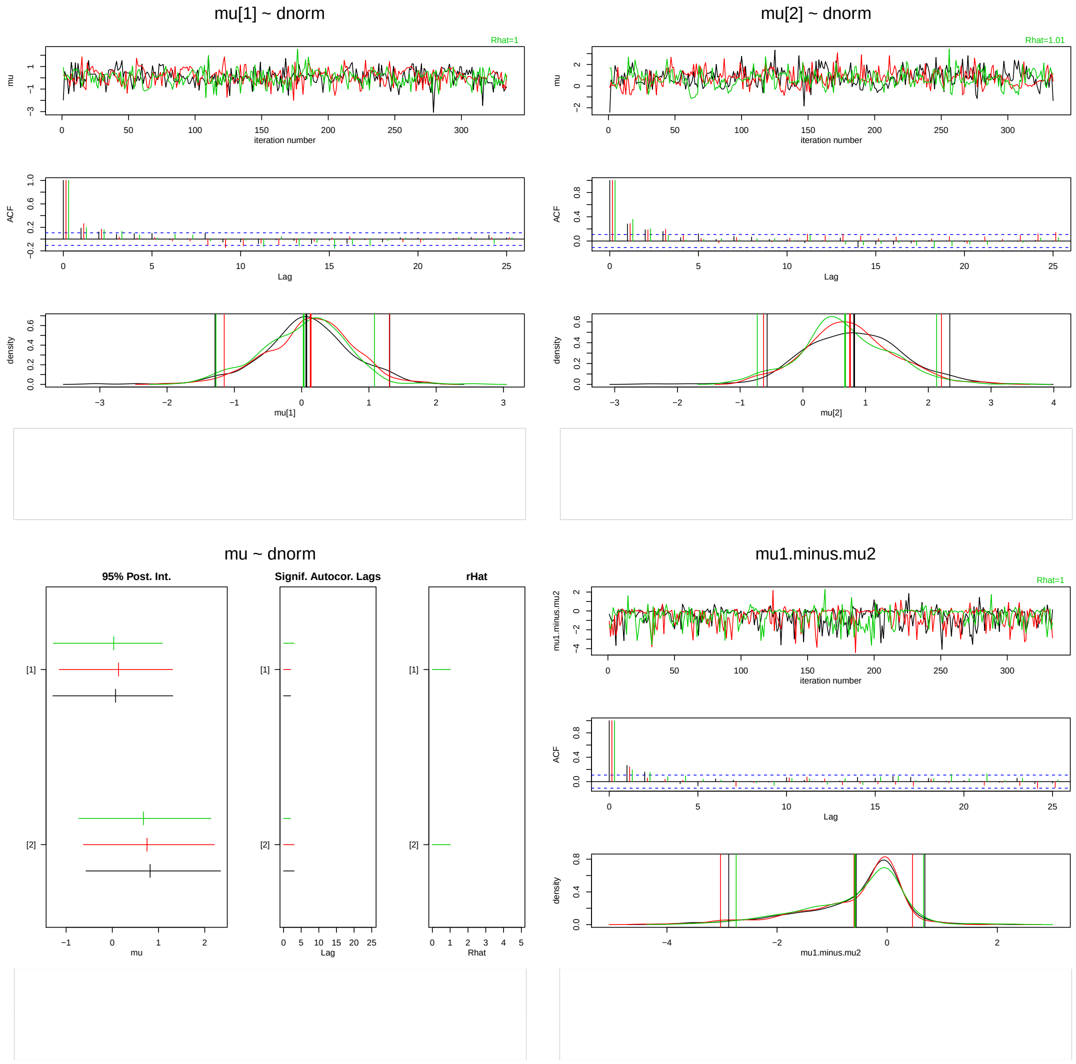


Figure 2: Question 2 plot μ

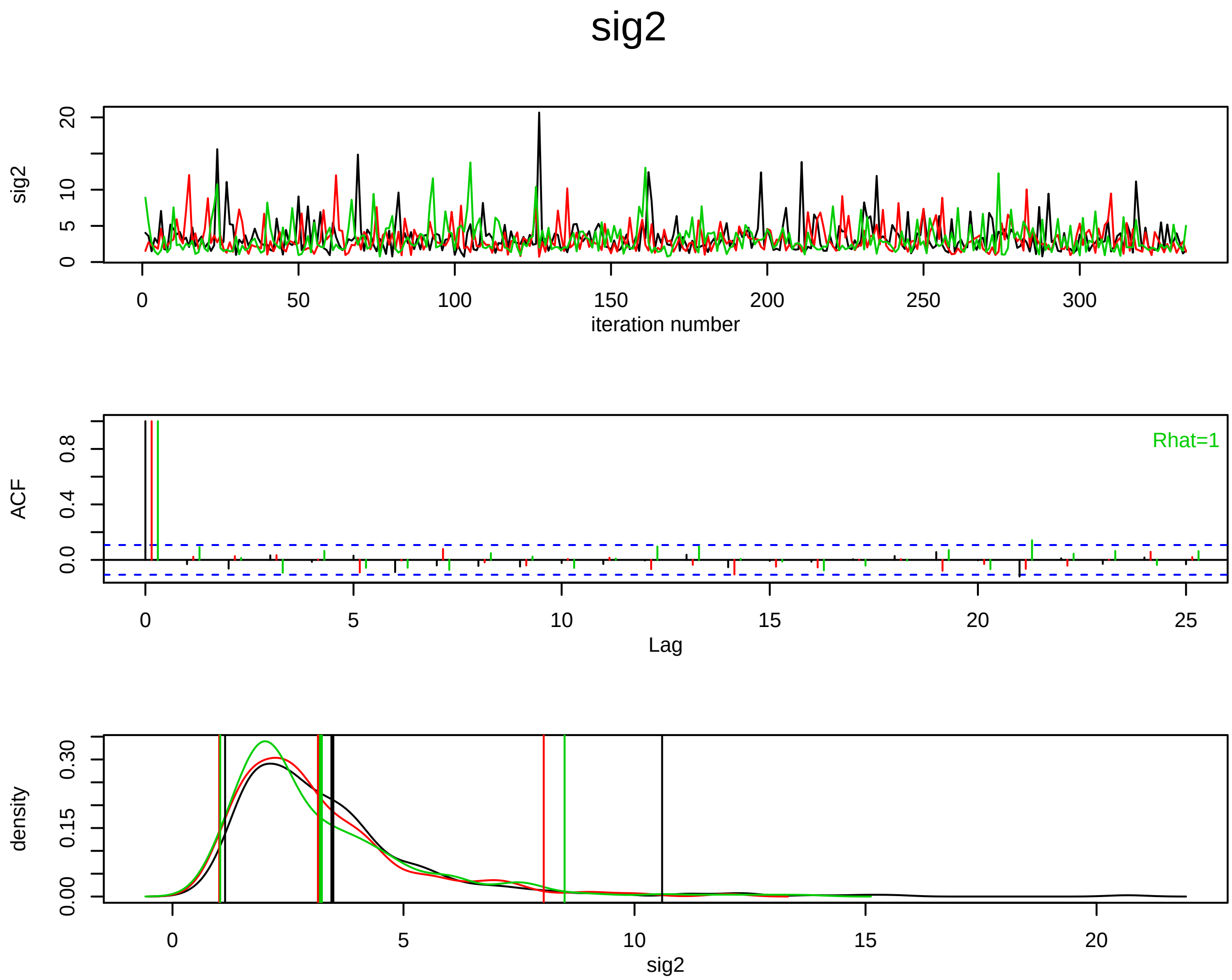


Figure 3: Question 2 plot σ_2

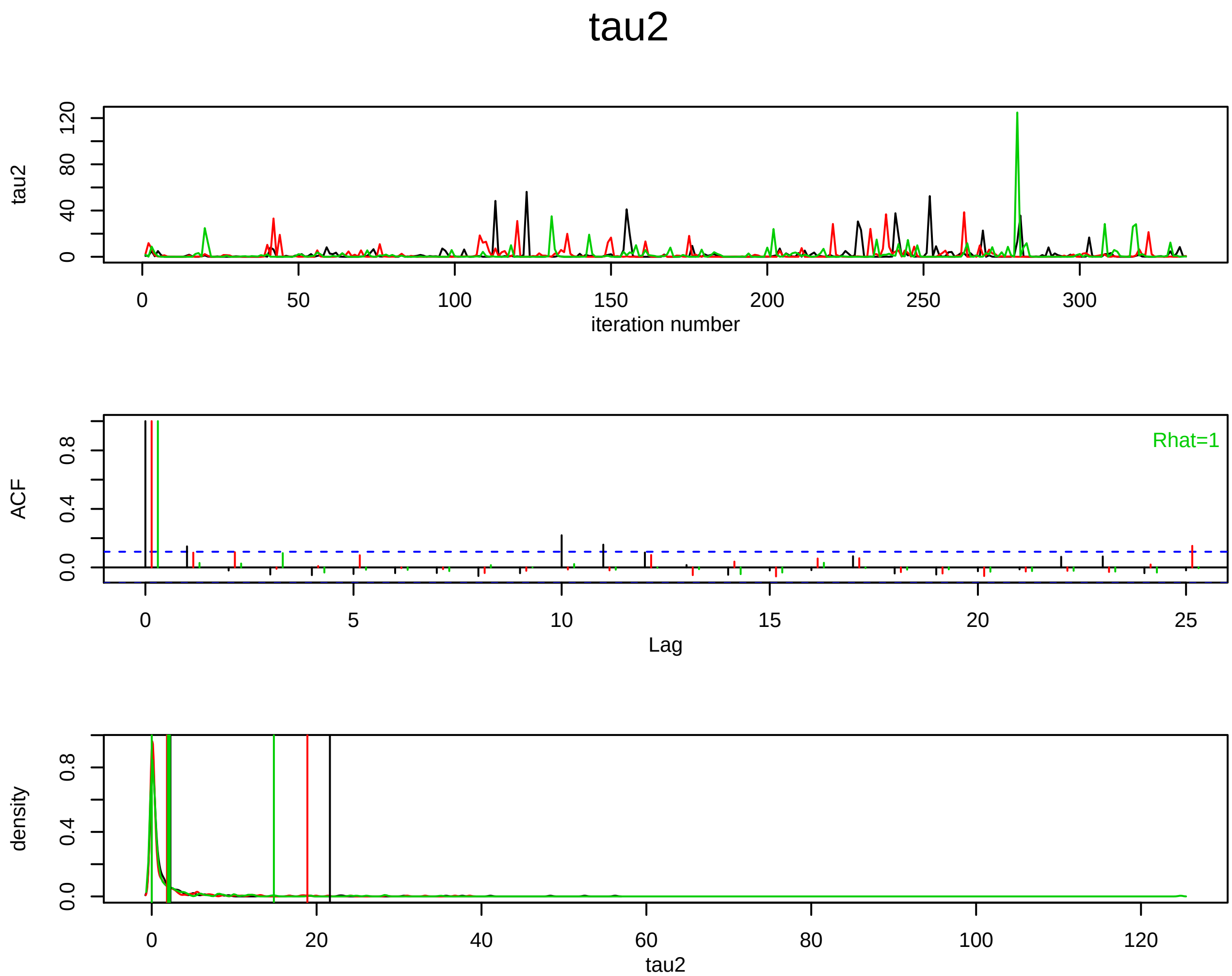


Figure 4: Question 2 plot τ_2

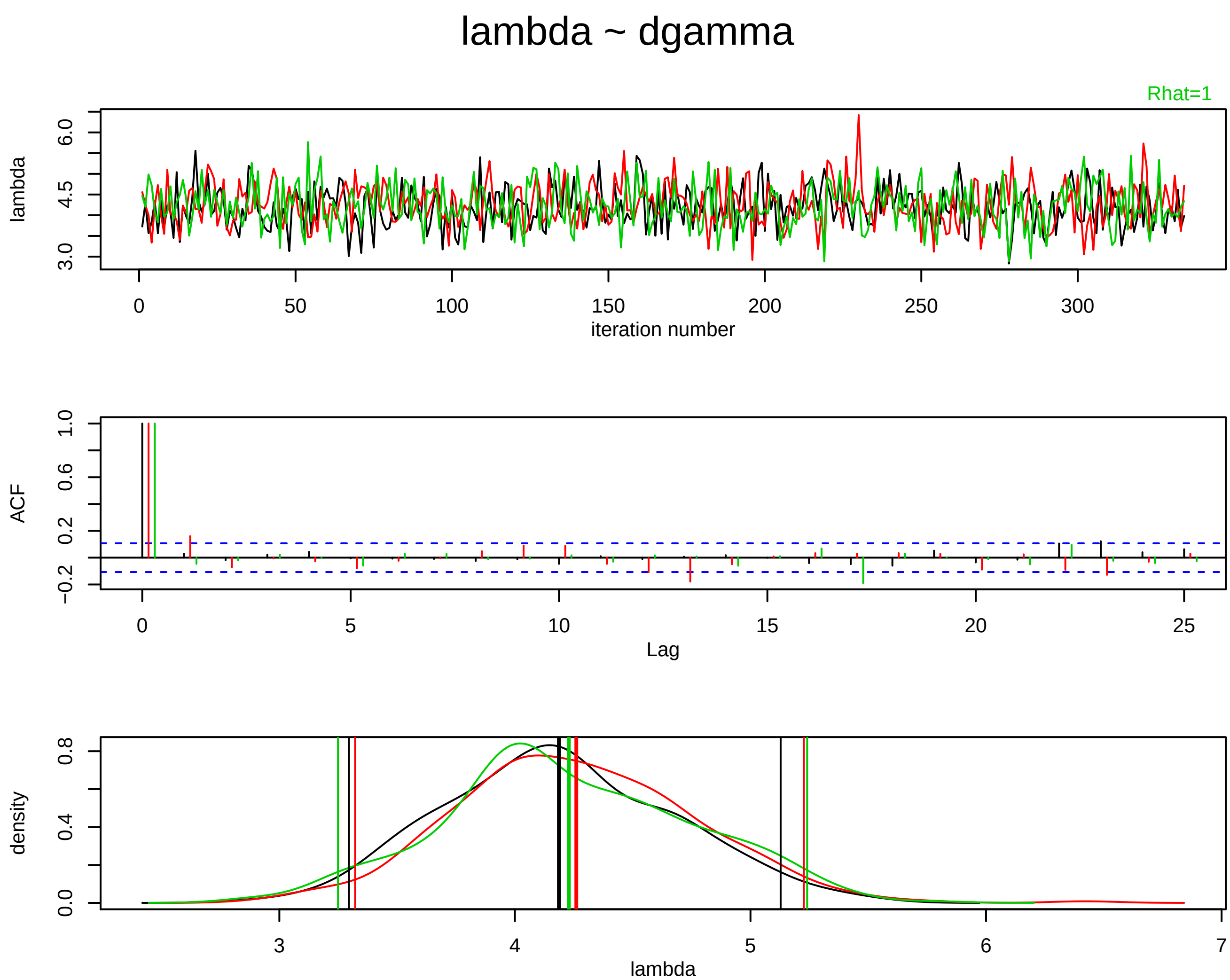


Figure 5: Question 3 plot