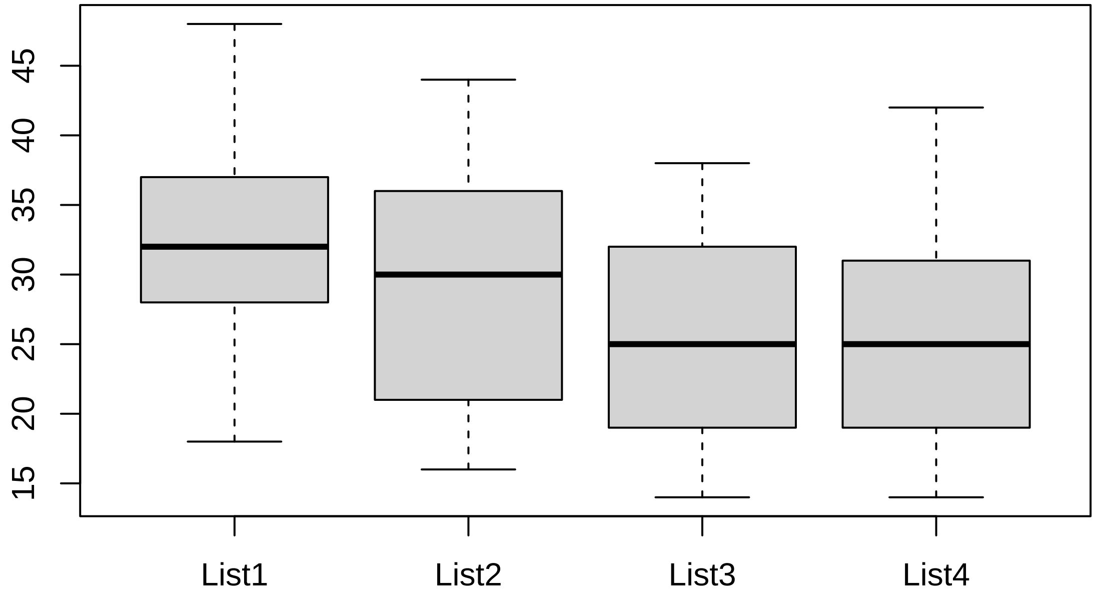
Quiz 2 Takehome

Michael Pena

2024-11-26

Question 1

```r
boxplot(hear)
```



(a)

```r
summary(hear)
```

```
##      List1           List2           List3           List4      
##  Min.   :18.00   Min.   :16.00   Min.   :14.00   Min.   :14.00  
##  1st Qu.:28.00   1st Qu.:21.50   1st Qu.:19.50   1st Qu.:19.50  
##  Median :32.00   Median :30.00   Median :25.00   Median :25.00  
##  Mean   :32.75   Mean   :29.67   Mean   :25.25   Mean   :25.58  
##  3rd Qu.:36.50   3rd Qu.:36.00   3rd Qu.:32.00   3rd Qu.:30.50  
##  Max.   :48.00   Max.   :44.00   Max.   :38.00   Max.   :42.00  
```

Right off the bat, we notice that there is a list 1 and list 2 have completely different spreads. List 1 is also noticeably higher and less spread than all the other lists. List 2 has the most spread and it looks as though list 3 and 4 are very similar. we could possibly start making our assumptions now about assuming the effect that being in certian list implies on the score.

(b)

$$
\begin{aligned}
&P(\theta, \mu, \sigma^2 | \mathbf{y}) \\
&\quad = P(\mu) P(\sigma^2) \prod_{j=1}^{J} P(\theta_j | \mu, \sigma^2) \prod_{j=1}^{J} \prod_{i=1}^{n_j} P(y_{ij} | \theta_j, \sigma^2)
\end{aligned}
$$

(c) we care about θ_j so we extract $P(\theta_j|\mu,\sigma^2)\prod_{i=1}^{n_j} P(y_{ij}|\theta_j,\sigma^2)$

$$\begin{aligned}
&P(\theta_i|\mu,\sigma^2)\prod_{i}^{n_j} P(y_{ij}|\theta_j,\sigma^2)\\
&\propto exp\Big[\frac{-1}{2\sigma^2}((\theta_j-\mu)^2+\sum_{i}^{n_j}(y_{ij}-\theta_j)^2)\Big]\\
&= exp\Big[\frac{-1}{2\sigma^2}(\theta_j^2-2\mu\theta_j+\mu^2+\sum_{i}^{n_j}(y_{ij}^2-2\theta_j y_{ij}+\theta_j^2)\Big]\\
&\propto exp\Big[\frac{-1}{2\sigma^2}(\theta_j^2-2\mu\theta_j+n_j\theta_j^2-2\theta_j\sum_{i}^{n_j} y_{ij})\Big]\\
&= exp\Big[\frac{-(n_j+1)}{2\sigma^2}(\theta_j^2-2\theta_j\frac{\mu+\sum_i^{n_j} y_{ij}}{n_j+1}+(\frac{\mu+\sum_i^{n_j} y_{ij}}{n_j+1})^2-(\frac{\mu+\sum_i^{n_j} y_{ij}}{n_j+1})^2)\Big]\\
&\propto exp\Big[\frac{-(n_j+1)}{2\sigma^2}(\theta_j^2-2\theta_j\frac{\mu+\sum_i^{n_j} y_{ij}}{n_j+1}+(\frac{\mu+\sum_i^{n_j} y_{ij}}{n_j+1})^2)\Big]\\
&= exp\Big[\frac{-(n_j+1)}{2\sigma^2}(\theta_j-\frac{\mu+\sum_i^{n_j} y_{ij}}{n_j+1})^2\Big]
\end{aligned}$$

This gives us $N(\frac{\mu+\sum_i^{n_j} y_{ij}}{n_j+1},\frac{\sigma^2}{1+n_j})$

Thus we have $P(\theta_j|\theta_{-j},\mu,\sigma^2,Y_{ij})\sim N(\frac{\mu+\sum_i^{n_j} y_{ij}}{25},\frac{\sigma^2}{25})$

(d) finding the $\mu|\theta_j,\mu,\sigma^2,\mathbf{Y}$

$$
\begin{aligned}
&P(\mu)\prod_{j}^{4}P(\theta_i|\mu,\sigma^2)\\
&\propto exp\Big[-(\mu-30)^2-\frac{1}{2\sigma^2}\sum_{i}^{4}(\theta_j-\mu)^2\Big]\\
&=exp\Big[-\Big[\mu^2-60\mu+900+\frac{1}{2\sigma^2}\sum_{j}^{4}\theta_j^2-\frac{2\mu}{2\sigma^2}\sum_{j}^{4}\theta_j+\frac{24}{2\sigma^2}\mu^2\Big]\Big]\\
&\propto exp\Big[-\Big[\mu^2-60\mu-\frac{2\mu}{2\sigma^2}\sum_{j}^{4}\theta_j+\frac{24}{2\sigma^2}\mu^2\Big]\Big]\\
&=exp\Big[-\Big[\mu^2(1+\frac{12}{\sigma^2})-\mu(60+\frac{1}{\sigma^2}\sum_{j}^{4}\theta_j)\Big]\Big]\\
&=exp\Big[-\frac{1}{1+\frac{12}{\sigma^2}}\Big[\mu^2-\mu(\frac{60+\frac{1}{\sigma^2}\sum_j^4\theta_j}{1+\frac{12}{\sigma^2}})\Big]\Big]
\end{aligned}
$$

note:

$$
\begin{aligned}
&-\frac{1}{1+\frac{12}{\sigma^2}}=\frac{-\sigma^2}{2(\frac{\sigma^2}{2}+6)}\\
&\frac{60+\frac{1}{\sigma^2}\sum_j^4\theta_j}{1+\frac{12}{\sigma^2}}=\frac{60\sigma^2+\sum_j^4\theta_j}{\sigma^2+12}\\
&=exp\Big[\frac{-\sigma^2}{2(\frac{\sigma^2}{2}+6)}\Big[\mu^2-\mu(\frac{60\sigma^2+\sum_j^4\theta_j}{\sigma^2+12})+[\frac{1}{2}(\frac{60\sigma^2+\sum_j^4\theta_j}{\sigma^2+12})]^2-[\frac{1}{2}(\frac{60\sigma^2+\sum_j^4\theta_j}{\sigma^2+12})]^2\Big]\Big]\\
&\propto exp\Big[\frac{\sigma^2}{2(\frac{\sigma^2}{2}+6)}\Big[\mu-\frac{1}{2}(\frac{60\sigma^2+\sum_j^4\theta_j}{\sigma^2+12})\Big]^2\Big]\\
&\Rightarrow \mu|\theta,\sigma^2,Y\sim Norm\left(\frac{1}{2}(\frac{60\sigma^2+\sum_j^4\theta_j}{\sigma^2+12}),\frac{\sigma^2}{\frac{\sigma^2}{2}+6}\right)
\end{aligned}
$$

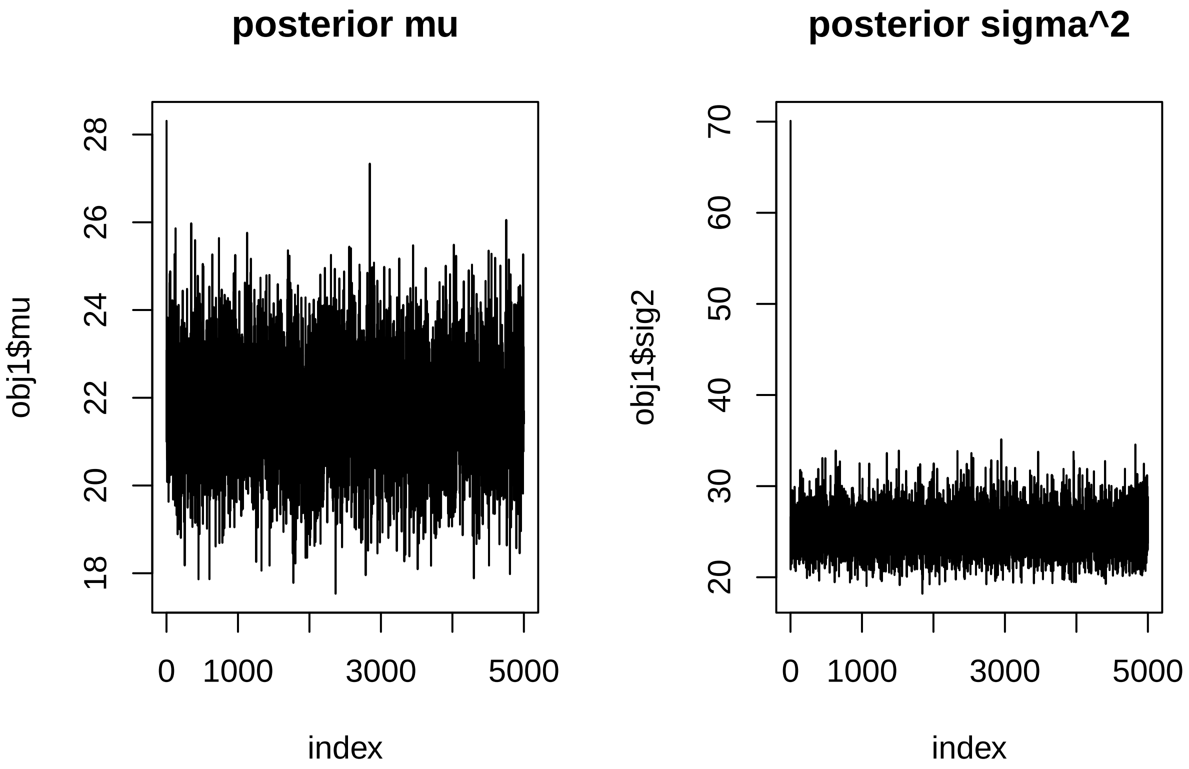
finding the distribution of $\sigma^2|\mu,\theta,\mathbf{Y}$

$$
\begin{aligned}
&P(\sigma^2)\prod_{j}^{4}P(\theta_i|\mu,\sigma^2)\prod_{j}^{4}\prod_{i}^{24}P(y_{ij}|\theta_j,\sigma^2)\\
&=100(\frac{1}{\sigma^2})^3e^{\frac{-10}{\sigma^2}}\Big[\prod_{j}^{4}\prod_{i}^{24}\frac{1}{\sqrt{2\pi}\sigma^2}e^{\frac{-1}{2\sigma^2}(y_{ij}-\theta_j)^2}\Big]\Big[\prod_{i}^{24}\frac{1}{\sqrt{2\pi}\sigma^2}e^{\frac{-1}{2\sigma^2}(\theta_j-\mu)^2}\Big]\\
&\propto\left(\frac{1}{\sigma^2}\right)^{24(5)+3}exp\Big[\frac{-1}{\sigma^2}\Big(10+\frac{1}{2}\sum_{j}^{4}\sum_{i}^{24}(y_{ij}-\theta_j)^2+\frac{1}{2}\sum_{i}^{24}(\theta_j-\mu)^2\Big)\Big]\\
&\propto InvGamma\left(122,10+\frac{1}{2}\sum_{j}^{4}\sum_{i}^{24}(y_{ij}-\theta_j)^2+\frac{1}{2}\sum_{i}^{24}(\theta_j-\mu)^2\right)
\end{aligned}
$$

```r
set.seed(538)
# making the MCMC algorithm
GIBBS <- function(y,B = 5000){
  # estiblish initials
```

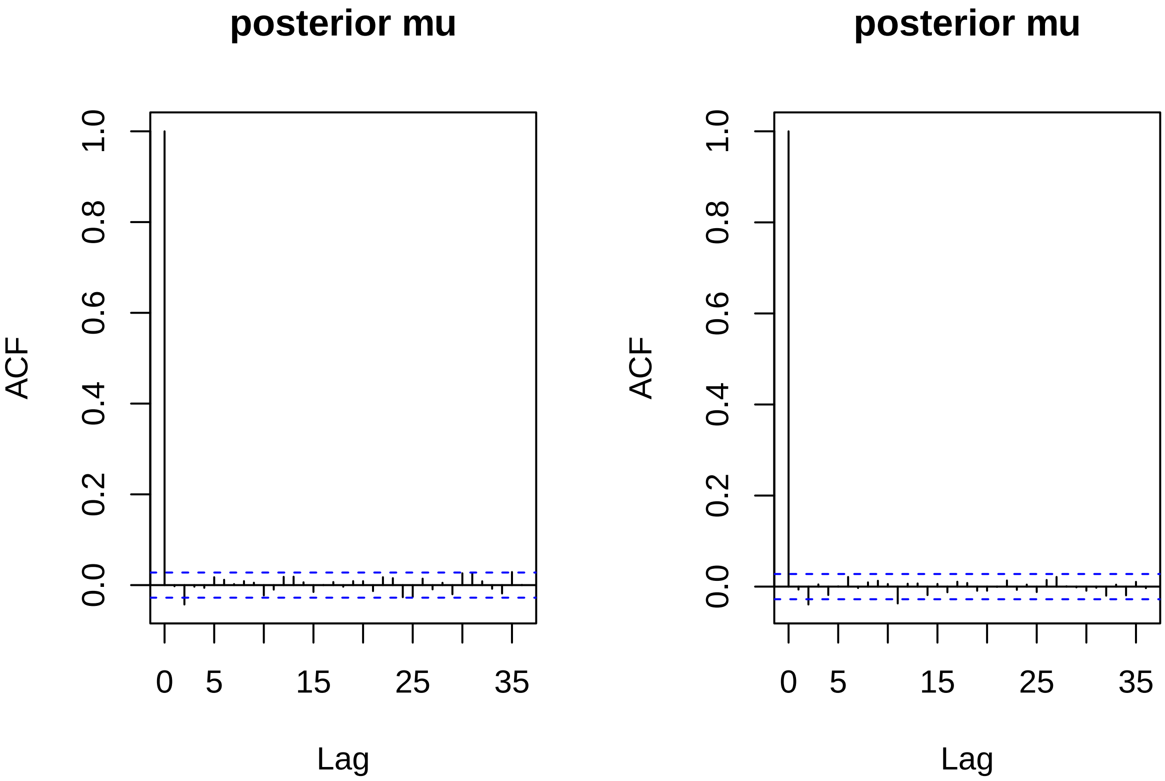
```r
   nrow(y) -> nj
   ncol(y) -> J
   theta.post <- matrix(0,B,J)
   mu.post <- sigsq.post <- numeric(length = B)
   theta.post[1,] = as.vector(colMeans(y))
   mu.post[1] = mean(theta.post[1,])
   sigsq.post[1] = var(as.vector(y))

  # MCMC loop
  for(i in 2:B){
    # set/reset
    sigsq0 <- sigsq.post[i-1]
    mu0 <-  mu.post[i-1]
    theta0 <- theta.post[i-1,]

    # mu's
    mu.mu = 0.5*(60*sigsq0 + sum(theta0))/(sigsq0 + 12)
    var.mu = sigsq0/(6 + 0.5*sigsq0)
    mu.post[i] = rnorm(1,mean = mu.mu, sd = sqrt(var.mu))

    # sigma-squareds
    beta.sigsq = 10 + 0.5*sum((t(y)-theta0)^2) + 0.5*sum((theta0 - mu0)^2)
    sigsq.post[i] = rinvgamma(1,shape = 122, scale = beta.sigsq)

    # thetas
    for(j in 1:J){
      mu.theta = (mu0 + sum(y[,j]))/25
      sd.theta = sqrt(sigsq0/25)
      theta.post[i,j] = rnorm(1,mean = mu.theta, sd = sd.theta)
    }
  }
 return(list("theta"=theta.post,"mu"=mu.post,"sig2"=sigsq.post))
}
# put in some data
as.matrix(hear) -> y

GIBBS(y) -> obj1
```

```r
# visualizations
par(mfrow = c(1,2))
plot.ts(obj1$mu,main="posterior mu", xlab ="index")
plot.ts(obj1$sig2,main="posterior sigma^2", xlab ="index")
```



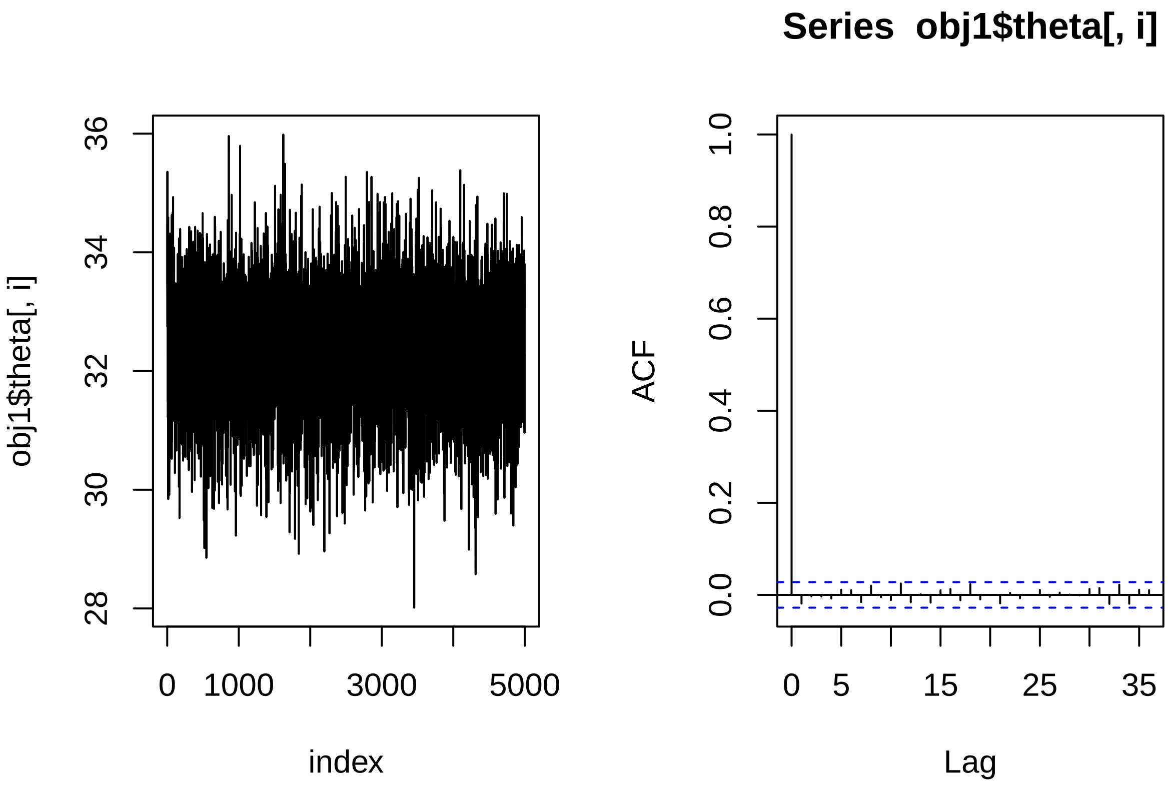
(e)

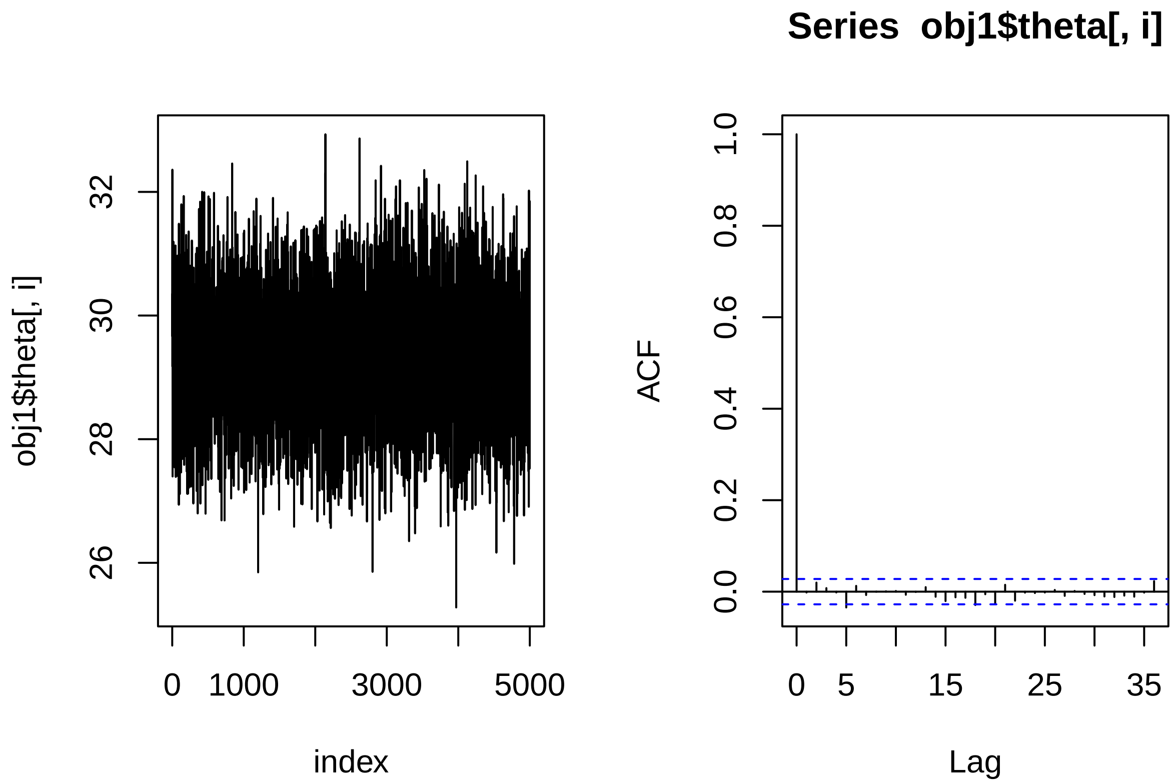
```r
acf(obj1$mu,main="posterior mu")
acf(obj1$sig2,main="posterior mu")
```

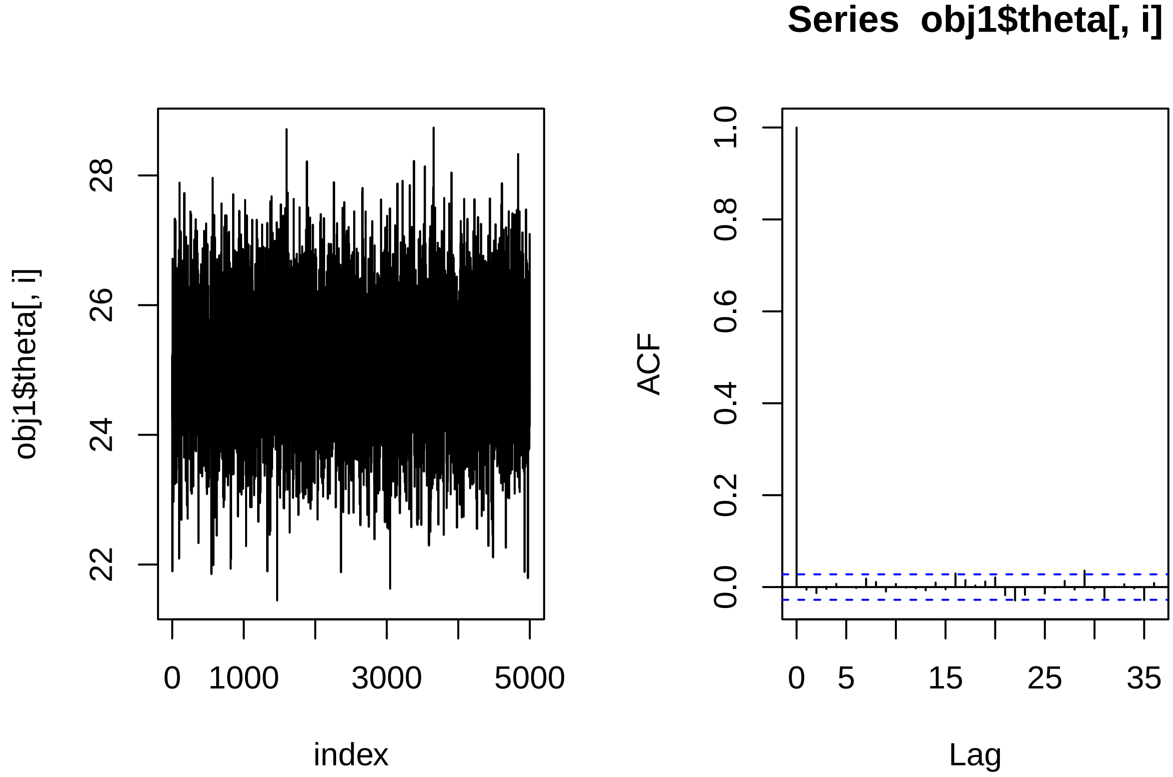


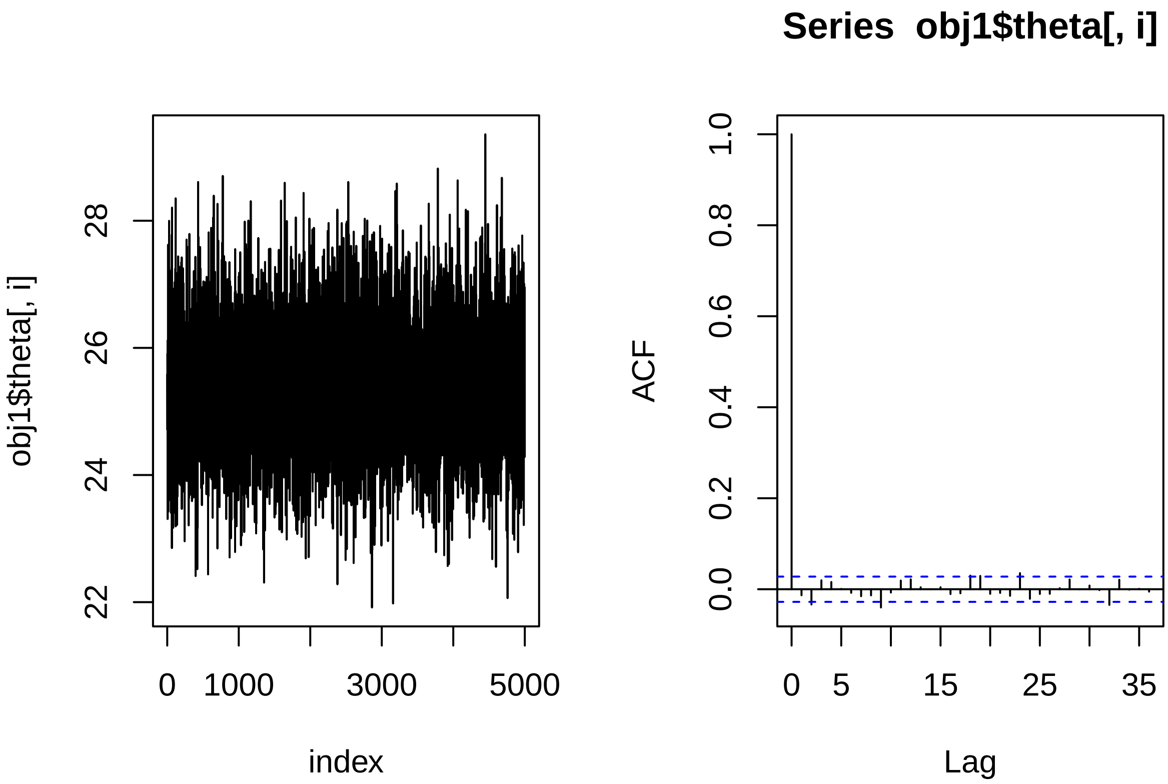
no thinning needed for these parameters

```r
#visualizing the four thetas
par(mfrow = c(1,2))
for(i in 1:4){
  plot.ts(obj1$theta[,i],xlab = "index")
  acf(obj1$theta[,i])
}
```




obj1$theta[, i]
26
28
30
32
0
1000
3000
5000
index
Series obj1$theta[, i]
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
15
25
35
Lag

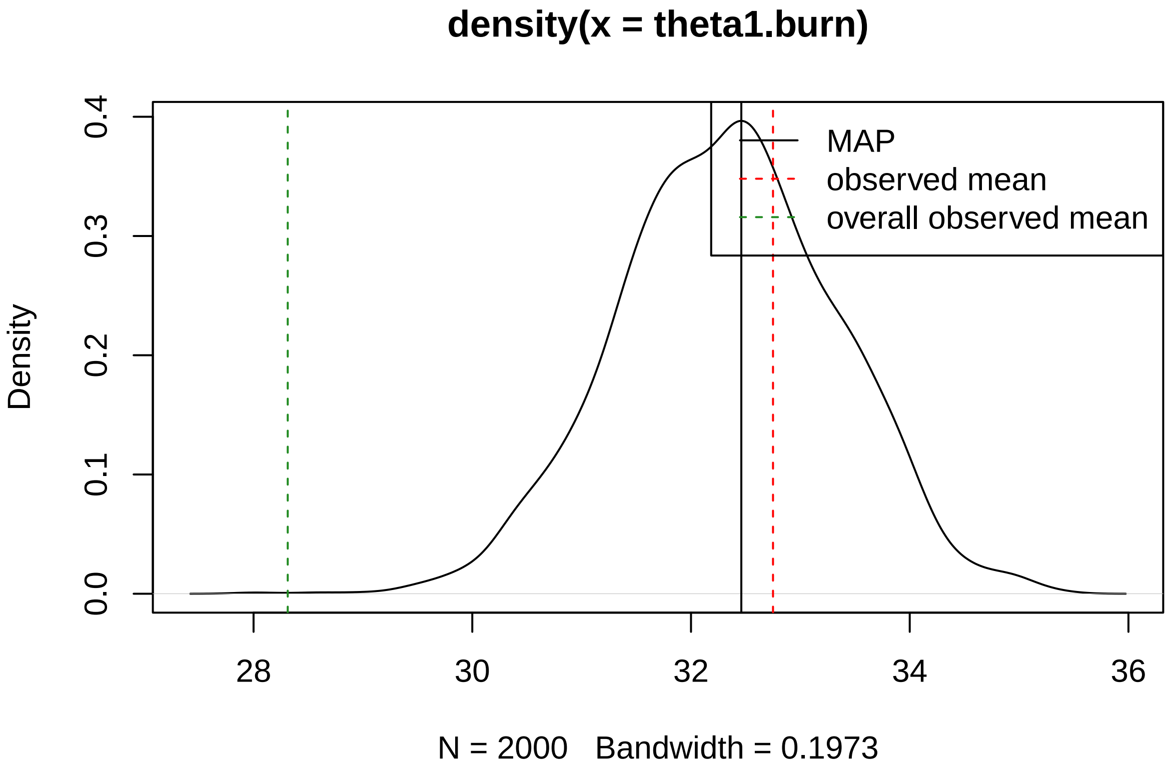

obj1$theta[, i]
22
24
26
28
0
1000
3000
5000
index
Series obj1$theta[, i]
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
15
25
35
Lag



no thinning needed for *theta* either. Let's burn off the first 3000

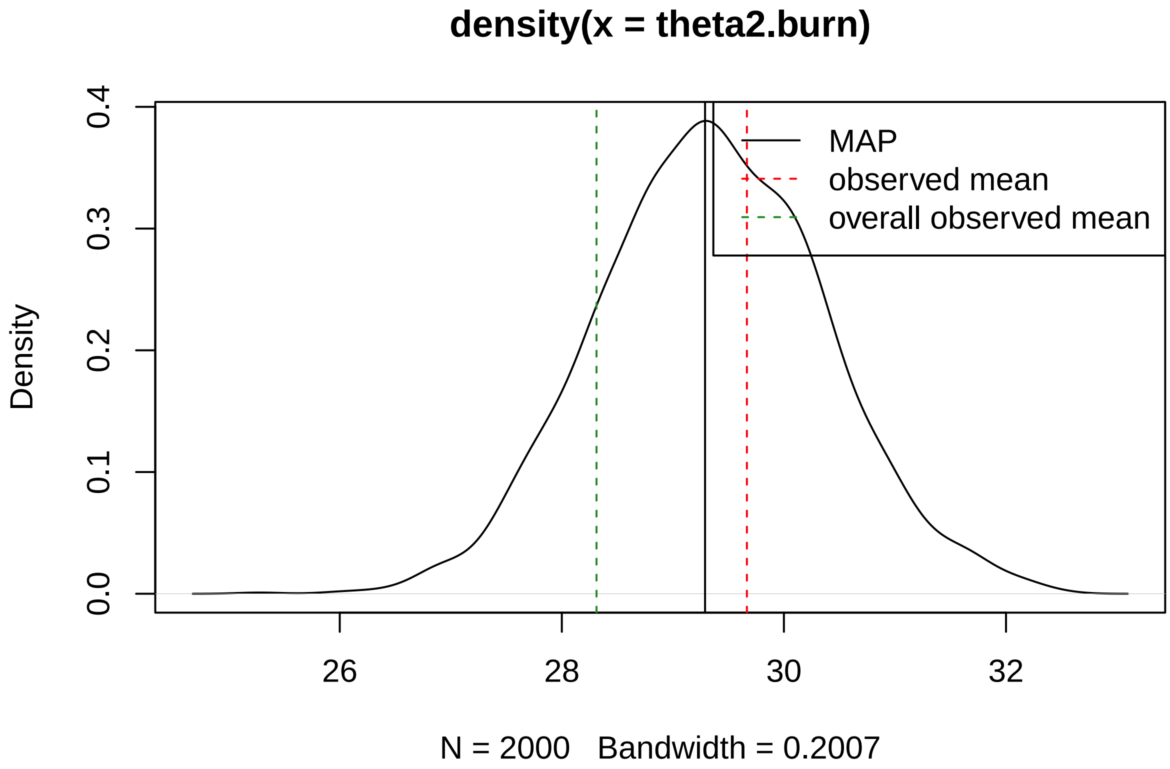
```r
# burning
k = 3001:5000
mu.burn = obj1$mu[k]
sig2.burn = obj1$sig2[k]
theta1.burn = obj1$theta[k,1]
theta2.burn = obj1$theta[k,2]
theta3.burn = obj1$theta[k,3]
theta4.burn = obj1$theta[k,4]
```

```r

#par(mfrow=c(2,1))
MAP <- function(X){density(X)$x[density(X)$y == max(density(X)$y)]}
#theta1
dens = density(theta1.burn)
plot(dens)
abline(v = MAP(theta1.burn))
abline(v = mean(y[,1]),lty = 2, col = 'red')
abline(v = mean(y), lty = 2, col="forestgreen")
legend("topright",
       c("MAP","observed mean","overall observed mean"),
       lty = c(1,2,2),
       col = c("black",'red','forestgreen'))
```

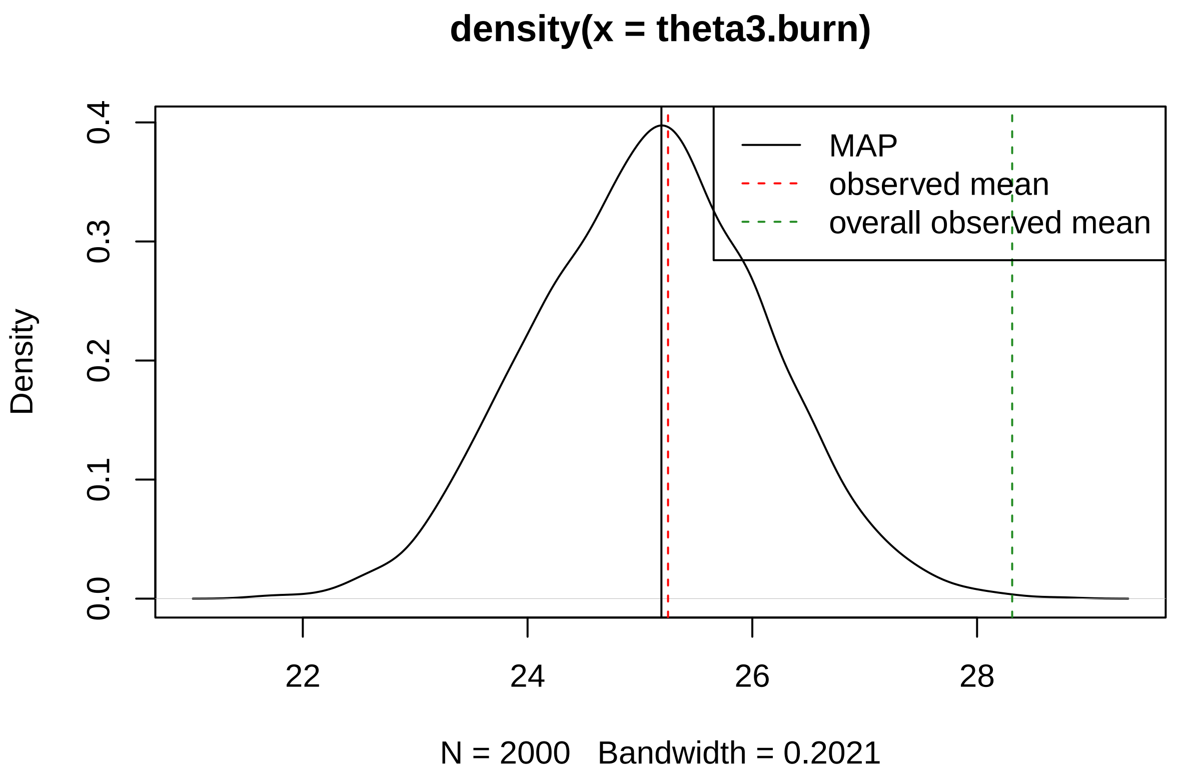


(f)

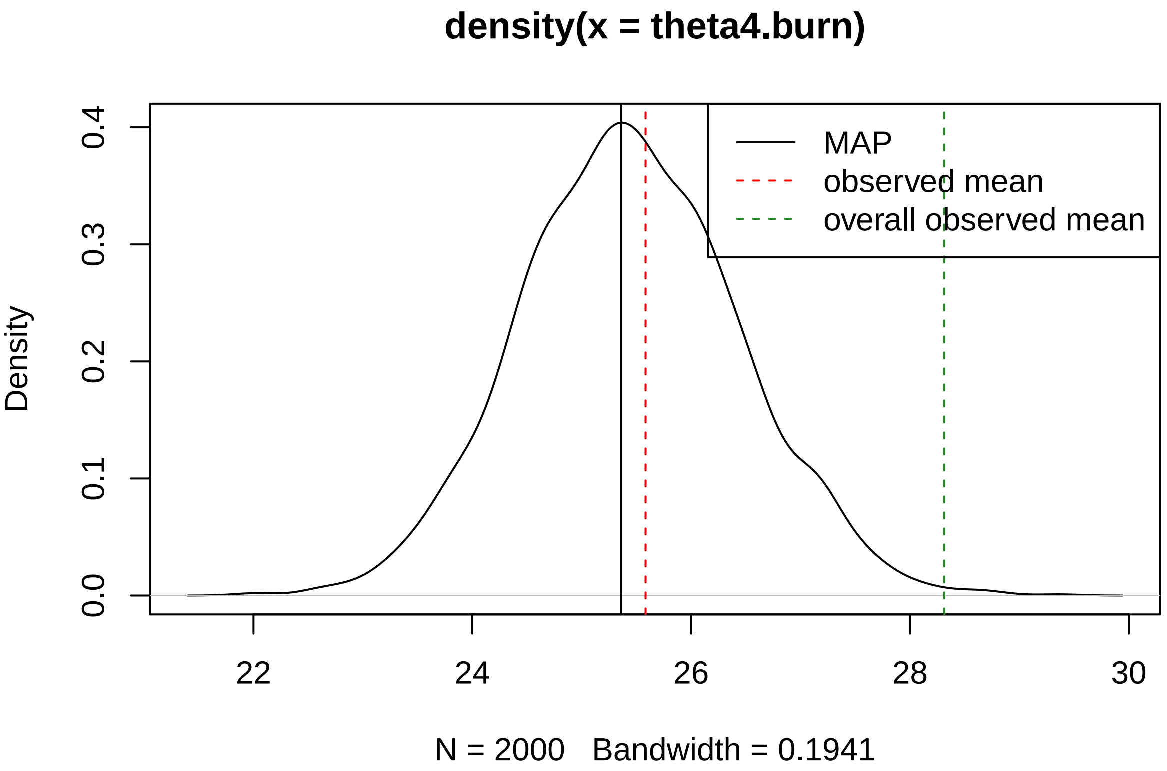
```r
#theta2
dens = density(theta2.burn)
plot(dens)
abline(v = MAP(theta2.burn))
abline(v = mean(y[,2]),lty = 2, col = 'red')
abline(v = mean(y), lty = 2, col="forestgreen")
legend("topright",
       c("MAP","observed mean","overall observed mean"),
       lty = c(1,2,2),
       col = c("black",'red','forestgreen'))
```



```r
#theta3
dens = density(theta3.burn)
plot(dens)
abline(v = MAP(theta3.burn))
abline(v = mean(y[,3]),lty = 2, col = 'red')
abline(v = mean(y), lty = 2, col="forestgreen")
legend("topright",
       c("MAP","observed mean","overall observed mean"),
       lty = c(1,2,2),
       col = c("black",'red','forestgreen'))
```



```
#theta4
dens = density(theta4.burn)
plot(dens)
abline(v = MAP(theta4.burn))
abline(v = mean(y[,4]),lty = 2, col = 'red')
abline(v = mean(y), lty = 2, col="forestgreen")
legend("topright",
       c("MAP","observed mean","overall observed mean"),
       lty = c(1,2,2),
       col = c("black",'red','forestgreen'))
```



```
# credible intervals and point estimates
THET <- cbind(theta1.burn,theta2.burn,theta3.burn,theta4.burn)
row = matrix(0,4,3)
for(r in 1:4){
  row[r,]= c(quantile(THET[,r],0.025),MAP(THET[,r]),quantile(THET[,r],.975))
}
colnames(row) <- c("Lower","MAP","Upper")
rownames(row) <- c("theta_1","theta_2","theta_3","theta_4")
kable(row, caption = "95% credible interval and MAP")
```

Table 1: 95% credible interval and MAP

	Lower	MAP	Upper
theta_1	30.35318	32.46017	34.17283
theta_2	27.38634	29.28986	31.38136
theta_3	23.11685	25.19115	27.07401
theta_4	23.55233	25.36050	27.41182

It seems that the overall observed mean is always farther from the point estimate than the overall mean so there is clearly effect that being in a different list has; being in a different list can imply that there is there is a different level of difficulty that each list will experience.

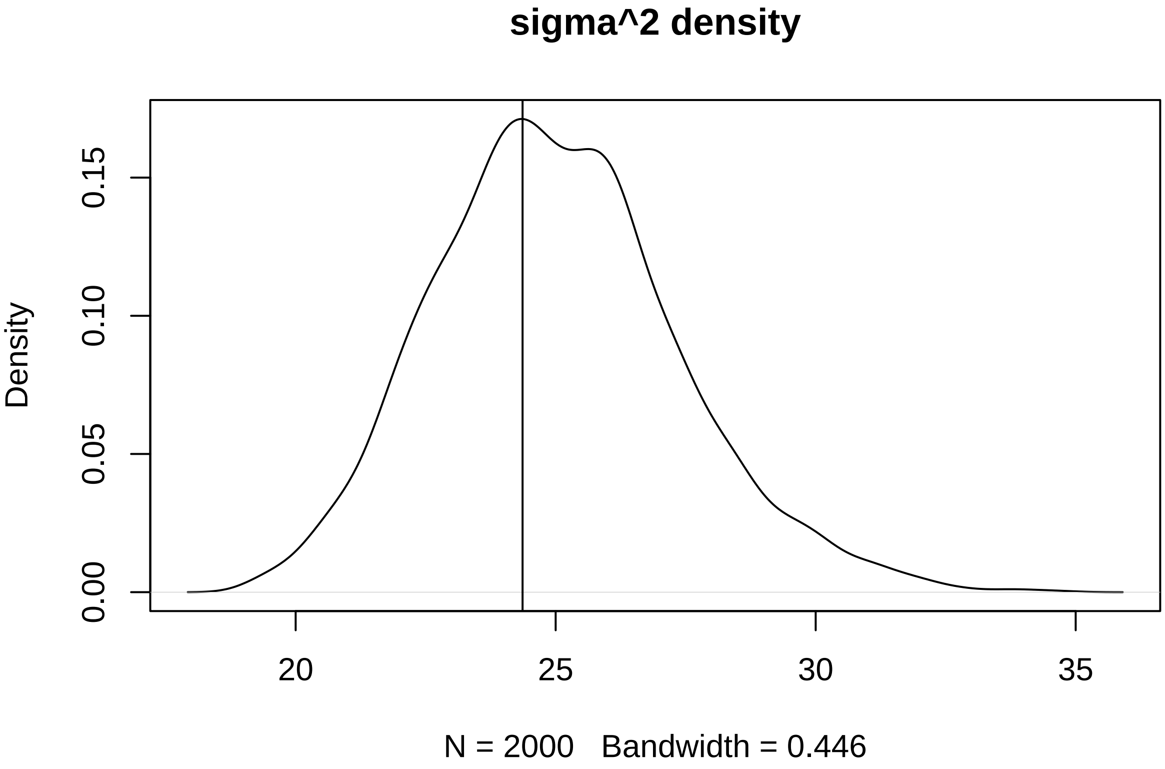
```
# cred int
cred.map = rbind(c(quantile(sig2.burn,.025),MAP(sig2.burn),quantile(sig2.burn,.975)),
      c(quantile(mu.burn,.025),MAP(mu.burn),quantile(mu.burn,.975)))
rownames(cred.map) = c("sigma.squared", "mu")
colnames(cred.map) = c("Lower","MAP","Upper")
kable(cred.map, caption = "95% cred. interval and MAP")
```

(g)

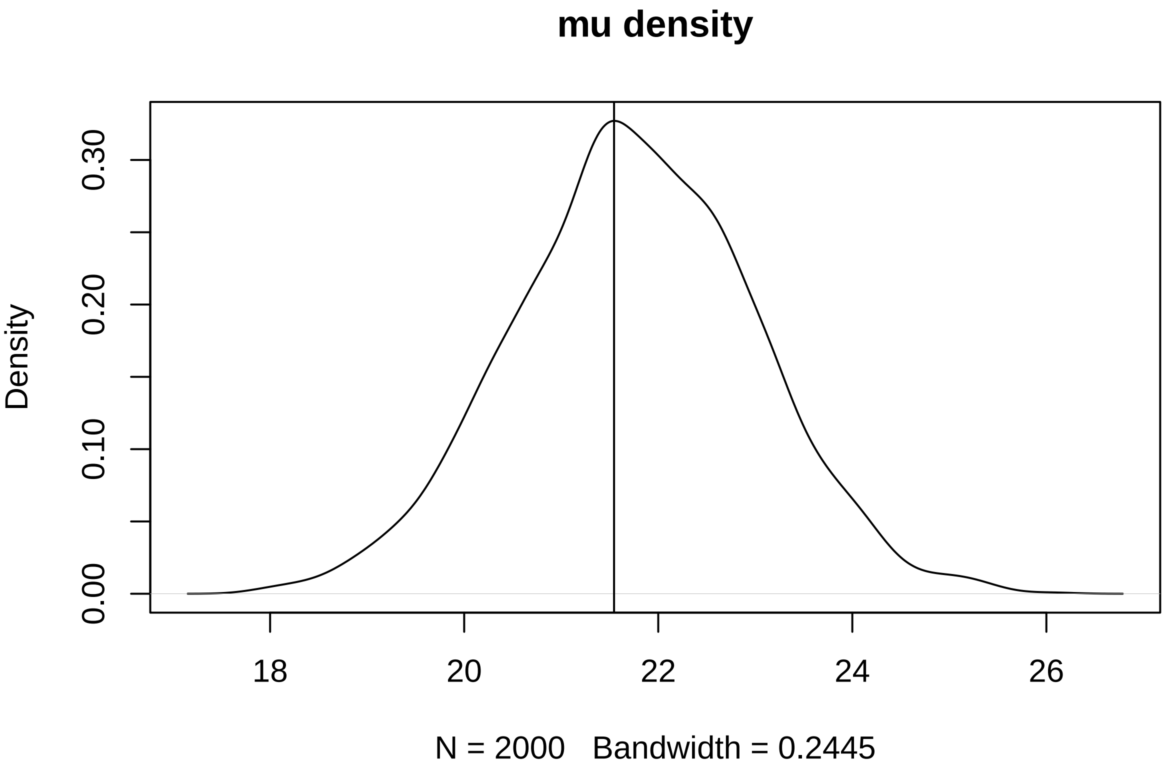
Table 2: 95% cred. interval and MAP

	Lower	MAP	Upper
sigma.squared	20.77128	24.36342	30.11671
mu	19.30405	21.54458	24.17287

```
# density maps
#sigma-squard
dens = density(sig2.burn)
plot(dens, main = "sigma^2 density")
abline(v = MAP(sig2.burn))
```



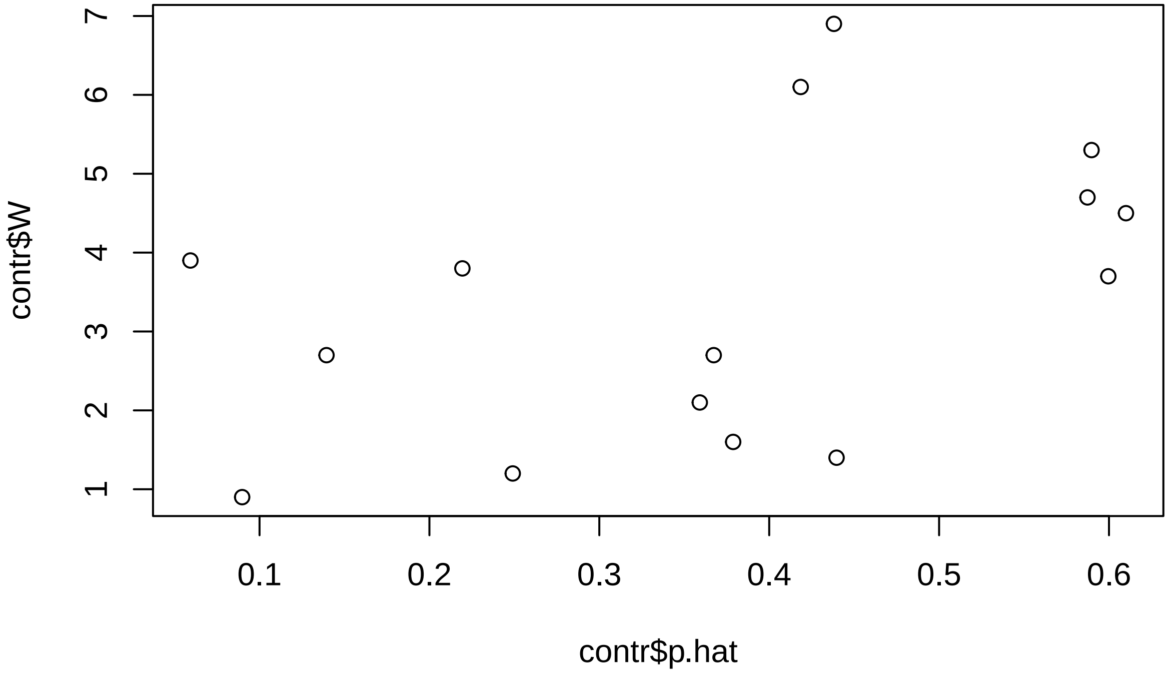
```
#mu
dens = density(mu.burn)
plot(dens, main = "mu density")
abline(v = MAP(mu.burn))
```



Standard deviation for Y is around 5 for and the overall population mean of Y of the scores is around 22 which seems to make sense for list 3 and 4.

Problem 2

```r
contr$p.hat <- contr$y/contr$N
plot(contr$p.hat,contr$W)
```

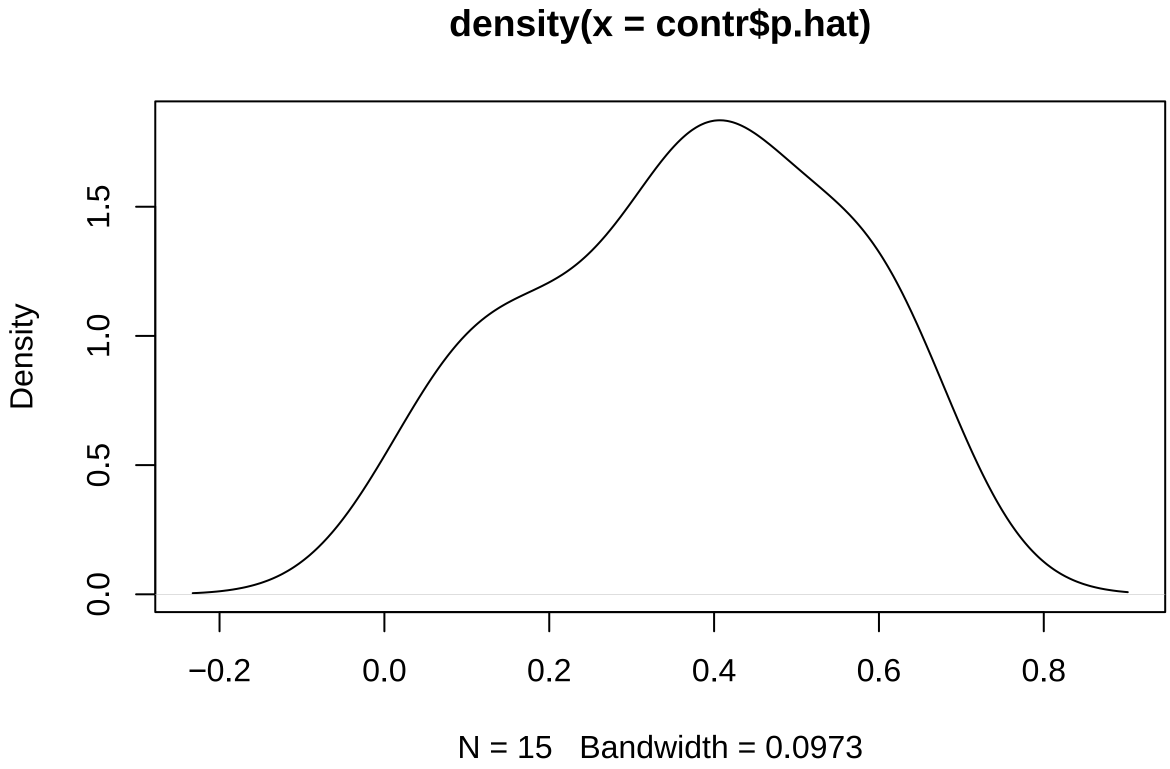


```
cor(contr$W,contr$p.hat)
```

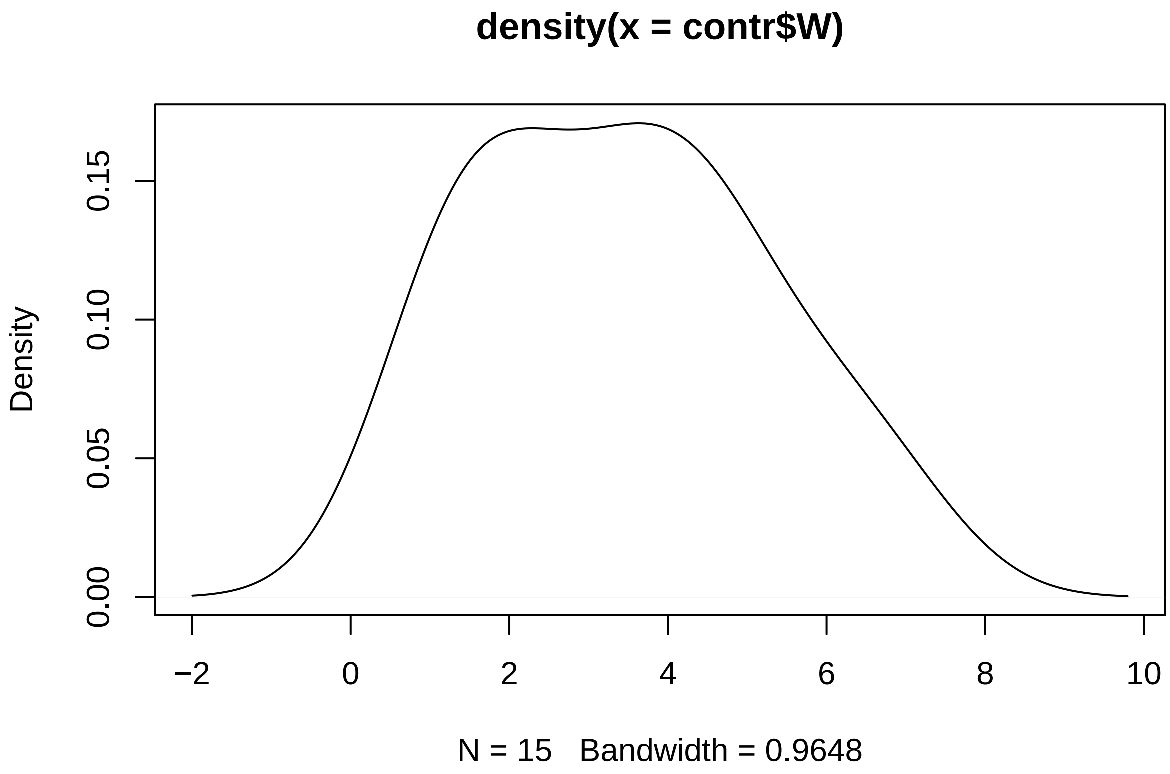
```
## [1] 0.4539023
```

There seems to be a slight positive correlation between proportion of women who use contraception and the education level but it's not too impressive (~0.4539). For simplicity sake, I have added a $\hat{p}$ column.

```
plot(density(contr$p.hat))
```



```
plot(density(contr$W))
```



```
# define bayesians funcs
Prior <-  function(alpha,beta){
    (alpha + beta)^(-5/2)
}

LLh <- function(theta,alpha,beta){
    sum(log(dbeta(theta, alpha, beta)))
}

Proposal <- function(alpha,beta){
    1/(alpha*beta)
}

rProposal <- function(n,mean,cov){
    rmvnorm(n,mean,cov)
}

# The Algorithm
Algo <- function(y,N,B,S){
  set.seed(538)
    # initials
    n = length(y)
    accept = 0
    alpha.post <- beta.post <- numeric()
    theta.post2 = matrix(0,ncol = B, nrow = n)

    # intial variables
    alpha0 = 4.1112
    beta0 = 7
    theta0 = numeric(length = n)

    # LOOP
    for(b in 1:B){
        # Gibbss step for thetas
        for(j in 1:n){
            shp1 = alpha0 + y[j]
            shp2 = beta0 + N[j] - y[j]
            theta0[j] = rbeta(1,shp1,shp2)
        }
        # MH step for alpha/beta
        phi.star = rProposal(1, c(log(alpha0),log(beta0)), 1*S)
        alpha.star = exp(phi.star[1])
        beta.star = exp(phi.star[2])

        r = exp(
        LLh(theta0, alpha.star, beta.star)
        + log(Prior(alpha.star, beta.star))
        + log(Proposal(alpha0, beta0))
        - LLh(theta0, alpha0, beta0)
        - log(Prior(alpha0, beta0))
        - log(Proposal(alpha.star, beta.star))
        )

        # logic selection
        if(runif(1) < min(1,r)){
            alpha0 = alpha.star
            beta0 = beta.star
            accept = accept + 1
        }
        # drop off
        alpha.post[b] = alpha0
        beta.post[b] = beta0
        theta.post2[,b] = theta0
    }

    # output
    return(list("alpha"=alpha.post,"beta"=beta.post,"theta"=theta.post2,"AR"=accept/B))
}
```

```
# run the algorithm
Stune = diag(2)
contr$y -> yin
contr$N -> Nin
Algo(yin,Nin,50000,Stune) -> obj2
obj2$AR
```

(b)

```
## [1] 0.11308
```

```
# retune with posterior
a11  = var(obj2$alpha)
```

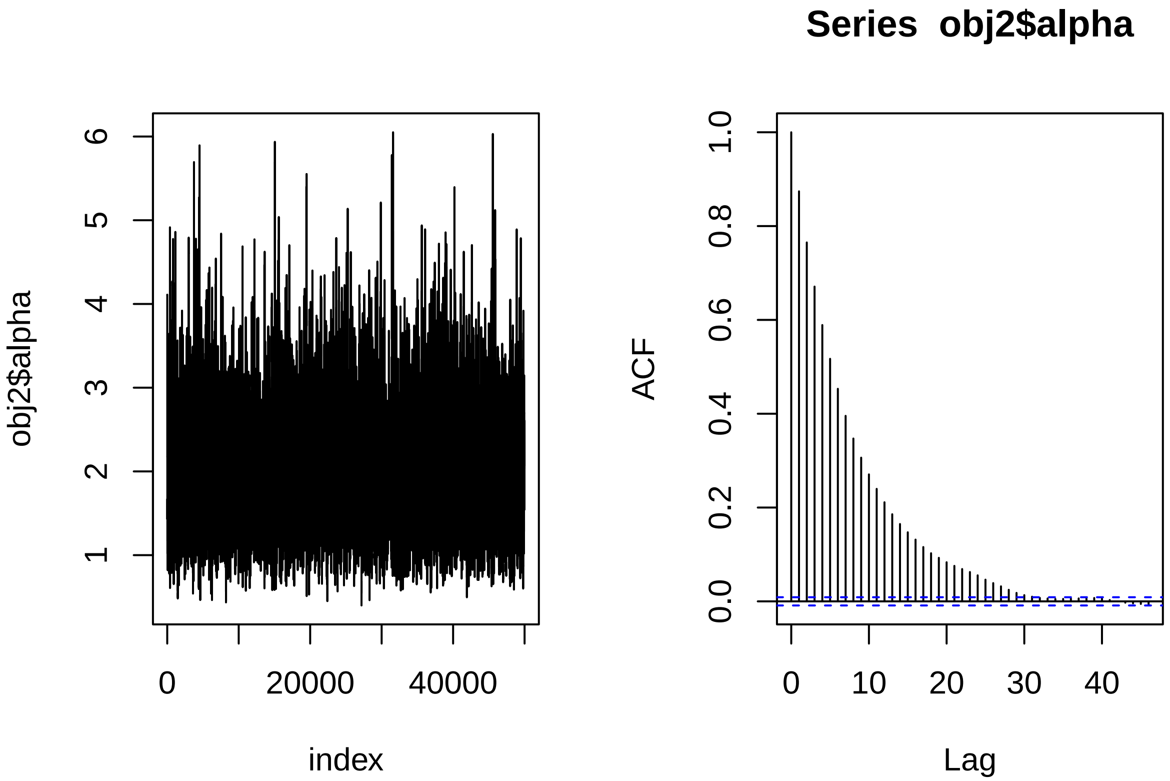
```
a12 <- cov(obj2$alpha,obj2$beta)
a22 <- var(obj2$beta)
Stune = matrix(c(a11,a12,a12,a22),2,2, byrow = T)
Stune
```

```
##           [,1]     [,2]
## [1,] 0.5585132 0.850026
## [2,] 0.8500260 1.821991
```

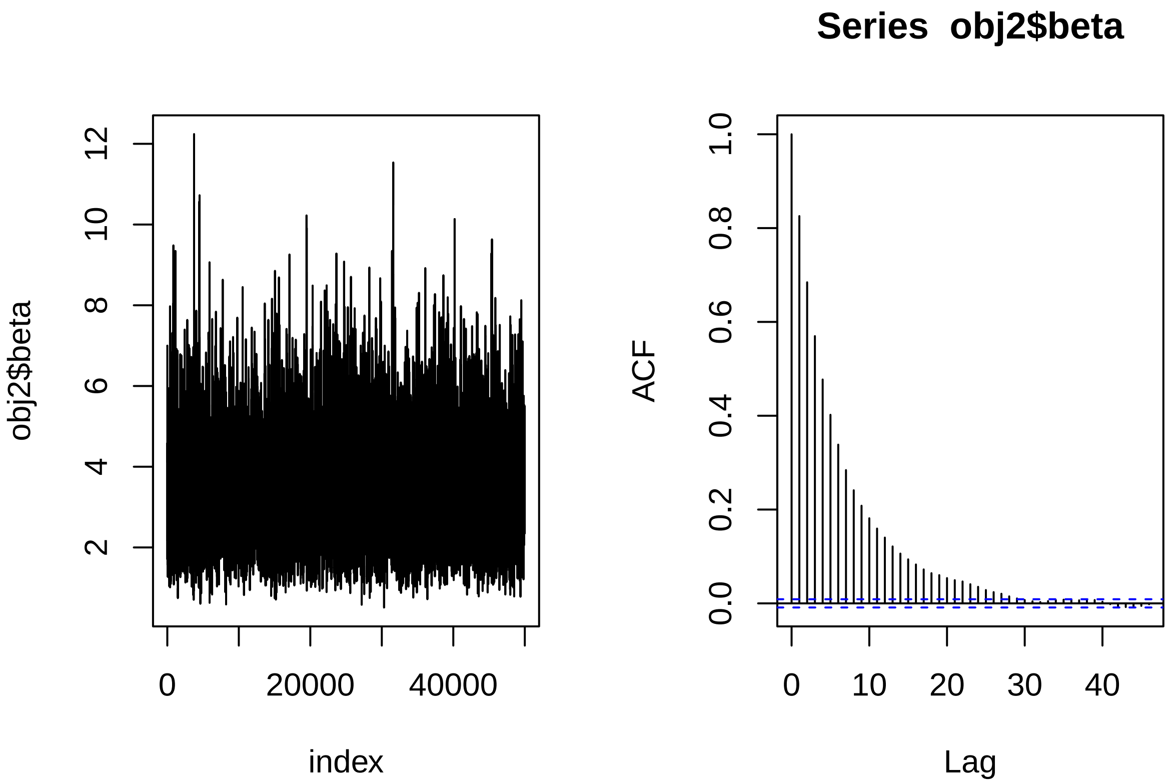
```
# rerun algorithm
Algo(yin,Nin,50000,Stune) -> obj2
obj2$AR
```

```
## [1] 0.18498
```

```
par(mfrow = c(1,2))
# visualizations
plot.ts(obj2$alpha,xlab = "index")
acf(obj2$alpha)
```

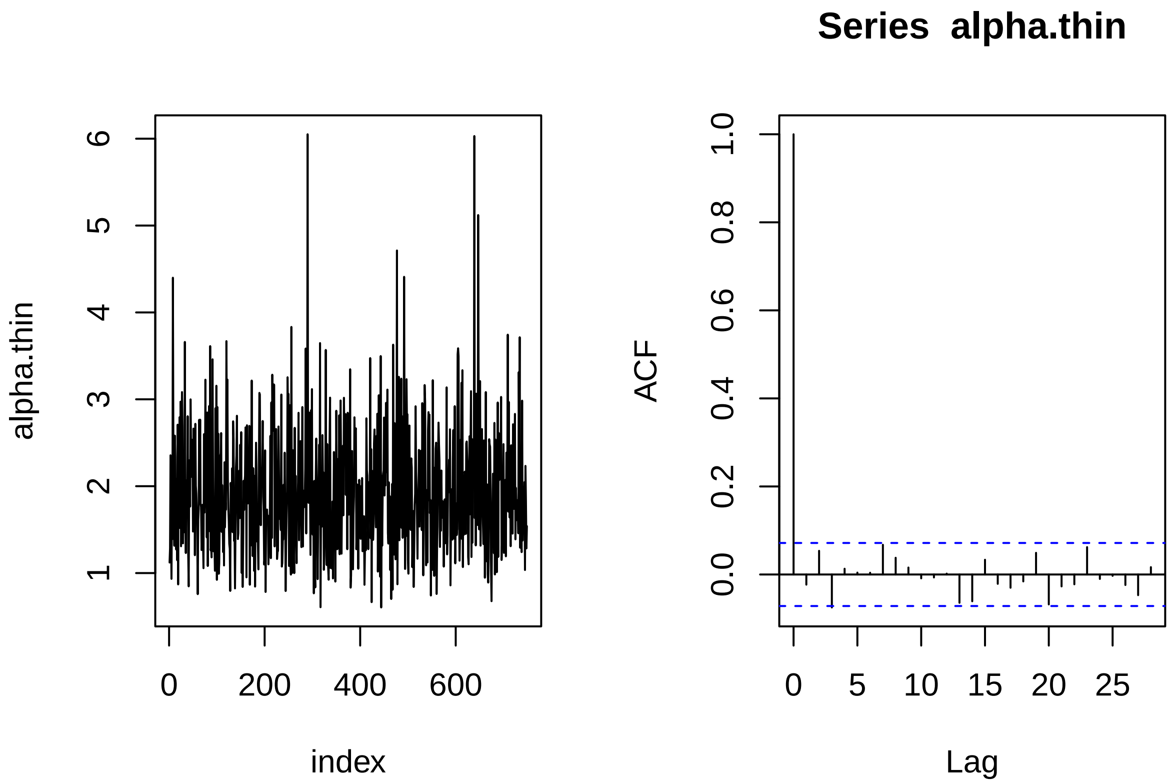


```
plot.ts(obj2$beta,xlab = "index")
acf(obj2$beta)
```

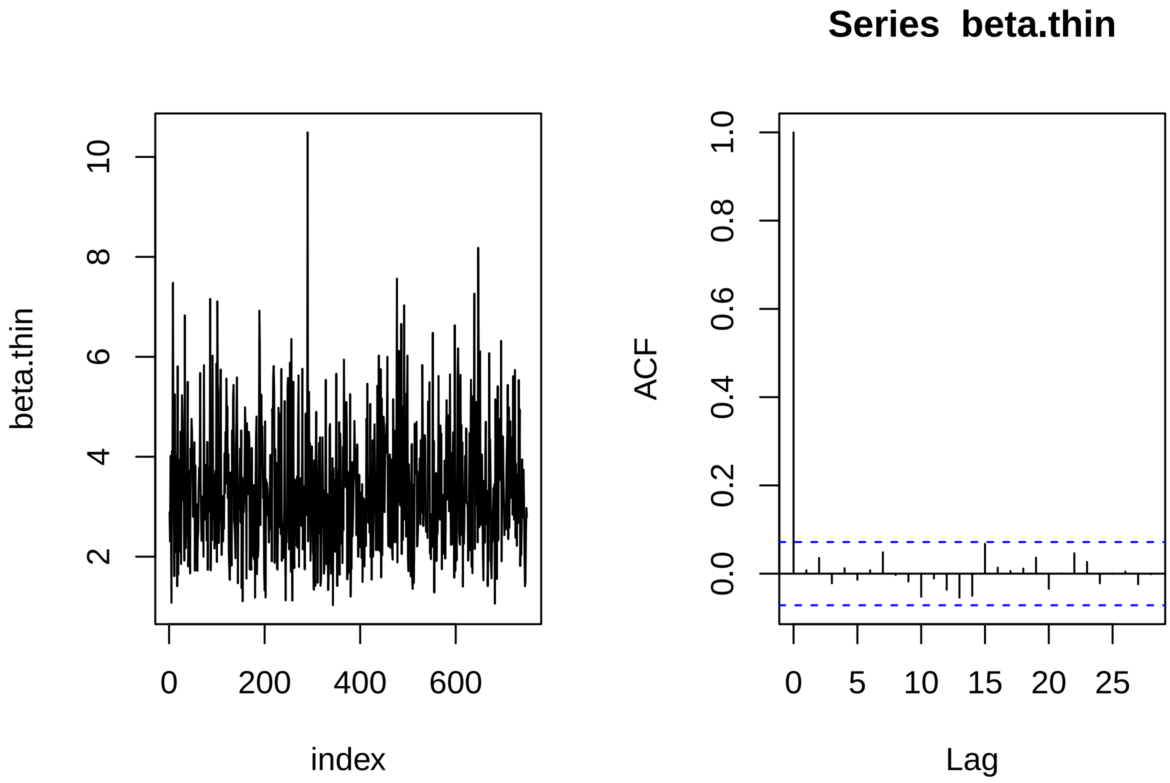


```
B = length(obj2$alpha)
```

```
burn = 20000
1:B*40 -> k
k = k[burn < k & k< B]
alpha.thin = obj2$alpha[k]
beta.thin = obj2$beta[k]
# burned/thinned visualizations
par(mfrow = c(1,2))
plot.ts(alpha.thin,xlab = "index")
acf(alpha.thin)
```

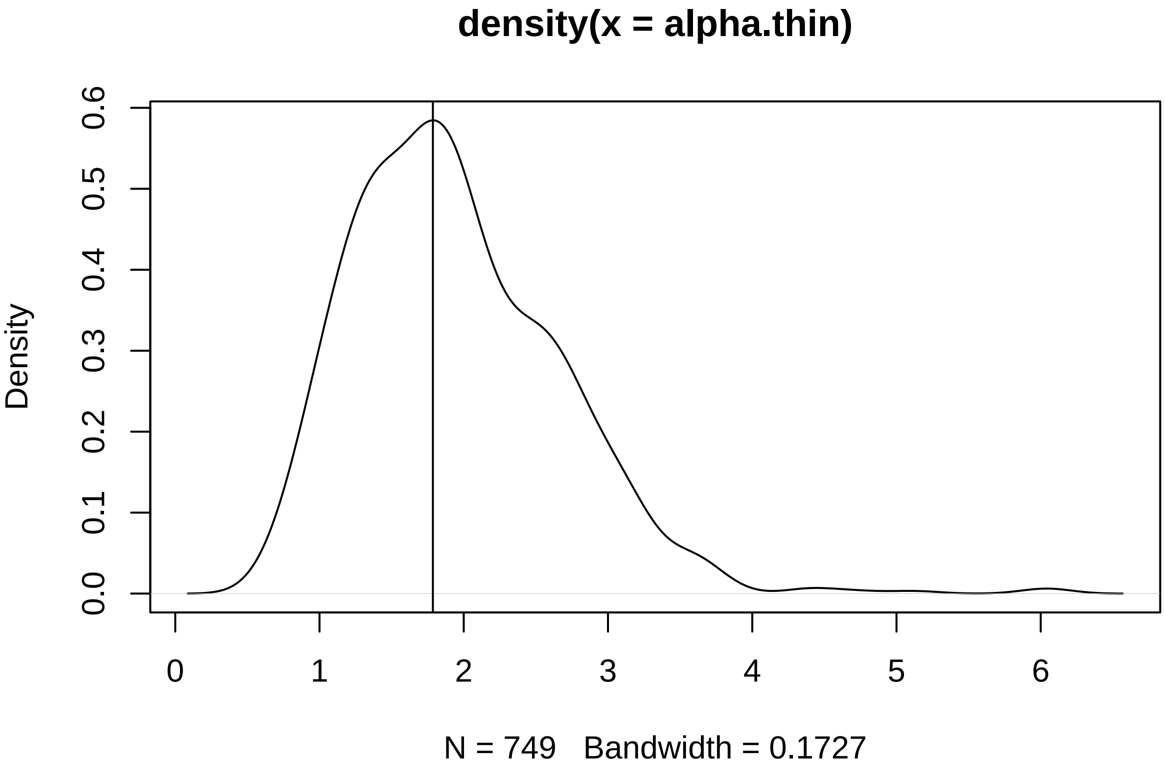


```r
plot.ts(beta.thin,xlab = "index")
acf(beta.thin)
```

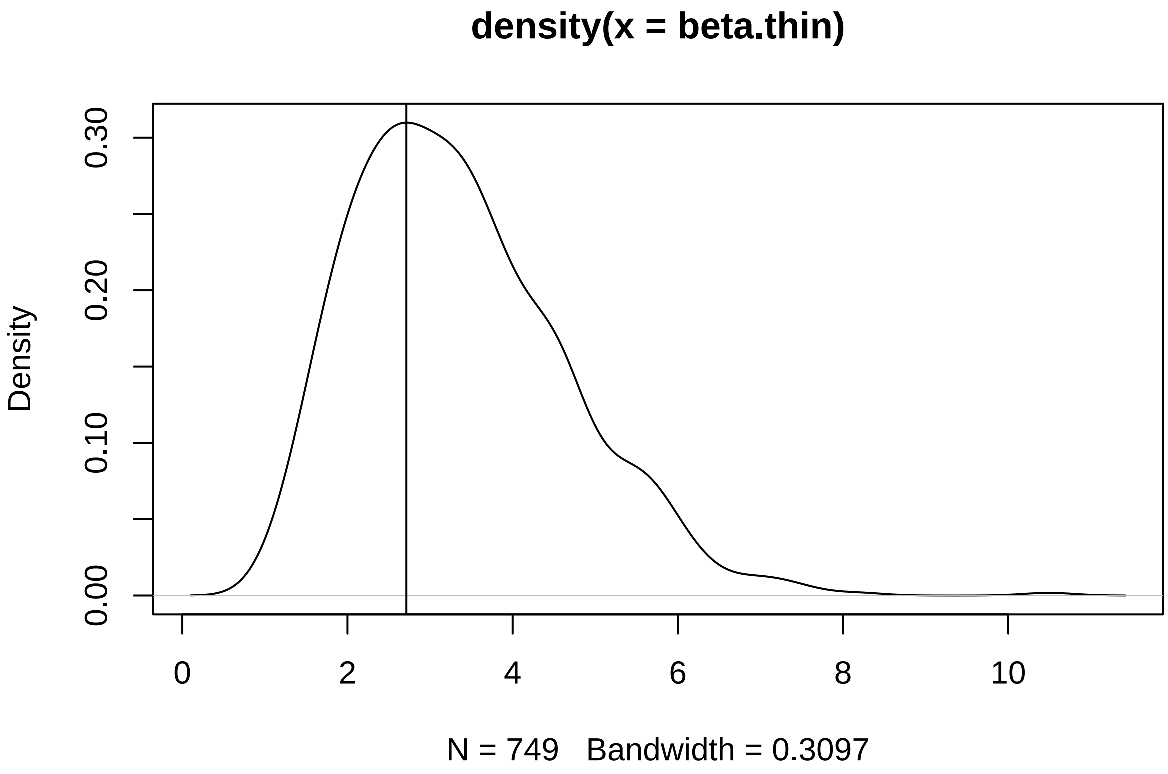


```r
# visuals
dens = density(alpha.thin)
```

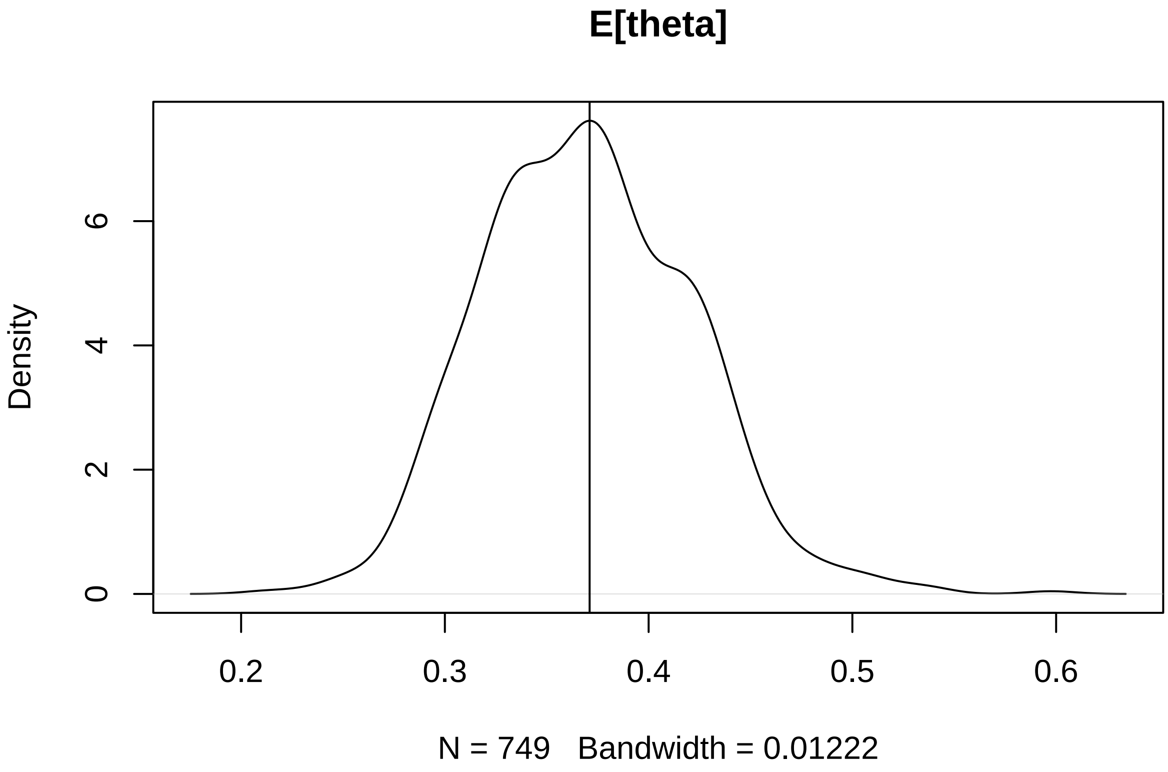
```
plot(dens)
abline(v = MAP(alpha.thin))
```



```
dens = density(beta.thin)
plot(dens)
abline(v = MAP(beta.thin))
```



```
ev = alpha.thin/(alpha.thin+beta.thin)
dens = density(ev)
plot(dens, main = "E[theta]")
abline(v = MAP(ev))
```



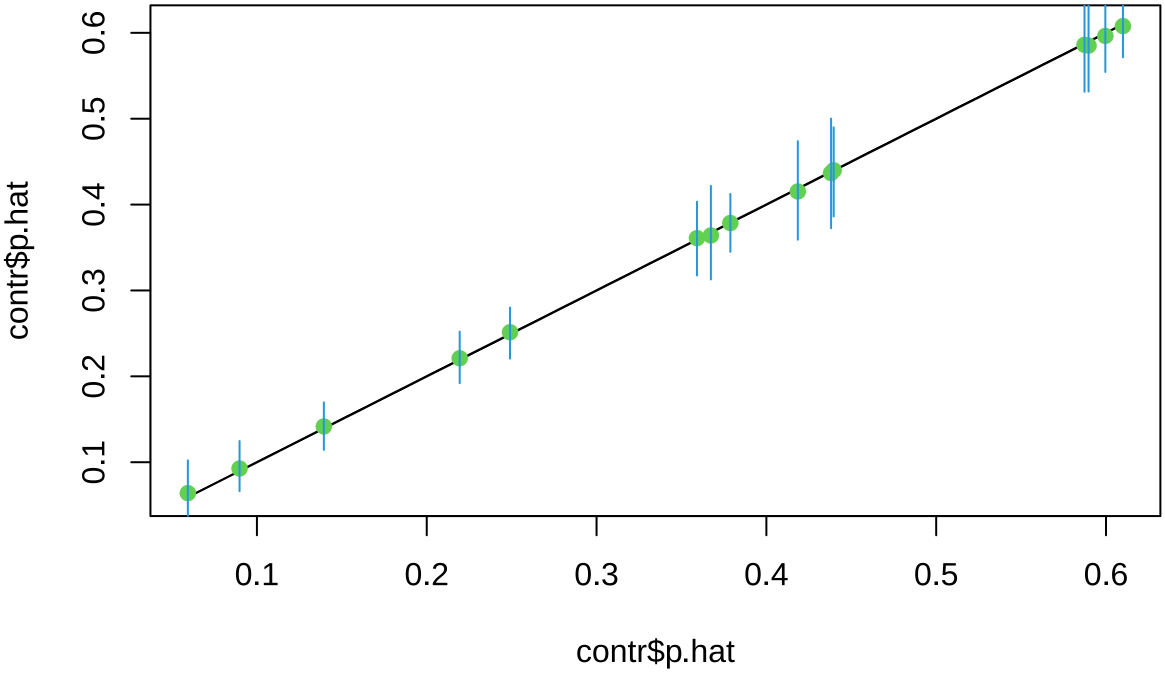
```
# table
cred.map = rbind(
  c(quantile(alpha.thin,0.025),MAP(alpha.thin),quantile(alpha.thin,.975)),
  c(quantile(alpha.thin,0.025),MAP(beta.thin),quantile(beta.thin,.975)),
  c(quantile(ev,0.025),MAP(ev),quantile(ev,.975))
)
rownames(cred.map) = c("alpha", "beta", "E[theta]")
colnames(cred.map) = c("Lower","MAP","Upper")
kable(cred.map, caption = "95% cred. interval and MAP")
```

Table 3: 95% cred. interval and MAP

	Lower	MAP	Upper
alpha	0.8465403	1.7861238	3.4966208
beta	0.8465403	2.7132091	6.1132207
E[theta]	0.2796714	0.3710374	0.4729117

I think we may be able to conclude that there is justification in using this model as our credible interval for $E[\theta]$ contains the $\bar{\theta}$ value of 0.4539023.

```
theta.thin = obj2$theta[,k]
#------------------ Displaying the theta's
CI = matrix(0, ncol = 3, nrow = 15)
for (j in 1:15) {
CI[j, ] = quantile(theta.thin[j, ], probs = c(0.025,0.5, 0.975))}
plot(contr$p.hat, contr$p.hat, type = "l")
points(contr$p.hat, CI[, 2], pch = 19, col = 3)
for (j in 1:15) {
points(c(contr$p.hat[j], contr$p.hat[j]), c(CI[j, 1], CI[j,3]), type = "l", col = 4)}
```



```r
cred.map = matrix(0,15,3)
for(i in 1:15){
  v = as.vector(quantile(theta.thin[i,],probs = c(0.025,.5,.975)))
  cred.map[i,] =v
}
colnames(cred.map) = c("Lower", "MAP", "Upper")
kable(cred.map)
```

Lower	MAP	Upper
0.2206381	0.2513666	0.2800969
0.1920471	0.2210838	0.2520631
0.3449612	0.3785759	0.4124509
0.5714656	0.6078265	0.6445202
0.1143959	0.1416608	0.1697049
0.5544570	0.5963938	0.6383091
0.3174360	0.3609464	0.4034365
0.0662946	0.0926787	0.1246903
0.3861322	0.4399413	0.4900302
0.5313950	0.5860343	0.6353225
0.3590568	0.4152829	0.4738853
0.5315867	0.5851860	0.6386907
0.3128118	0.3640594	0.4217566
0.0370283	0.0640338	0.1022062
0.3724190	0.4367750	0.5002366

```r
kable(contr$W)
```

x
1.2
3.8
1.6
4.5
2.7
3.7
2.1
0.9
1.4
4.7
6.1
5.3
2.7
3.9
6.9

Looking at the graph above, we can conclude that our posterior θ's imply that we need to account for cluster differences. Thus the hierarchical model is appropriate.

analyzing the literature

I believe that the main objective of the authors was to show that the model on medicare.gov was it rather misleading in predicting their mortality rate (of heart attack victims); it was important for them to show there's a more efficient way of estimating a mortality rate given the various attributes of the medical centers at hand. The issues that the statisticians ran into considering the hierarchical model being used for "hospital compare": the model was inaccurately following the data when considering lower volume medical centers. Apparently the previous model was making assumptions about spread and expected value that, not accounting for attributes such as size, were not fitting to the data. Smaller volume hospitals will always have less data than bigger facilities; so spread, or variance, is going to be naturally different when throwing them into a pool where data sets from larger hospital swim. The mortality rate for these lower volume hospitals was "shrinking" down to the national average. Imagine this situation person at risk for heart attack comparing hospitals whose mortality rates are being underestimated. Overall, the authors found that considering more hospital attributes (as well as patient risk factors and their relation to the hospital) is what makes the better model. Personally, I think this could be detrimental to public policy as policy makers who are unaware of the nuances of the topic at hand can be misled to make poor decisions because of a faulty statistical model; we must be sure our assumptions align with our observations. Three small improve their mortality rate they received more funding but miss out on it because of politicians who are lead by the numbers of "hospital compare."