STAT 4224 HW #6

Carlyle Morgan

11/29/2021

Problem 1

(a) One would expect $\operatorname{var}(y_{ij} \mid \theta_j, \sigma^2)$ is less than $\operatorname{var}(y_{ij} \mid \mu, \tau^2, \sigma^2)$, as μ and τ^2 influence y_{ij} through θ_{ij}, an added layer of uncertainty that will likely inflate the variance relatively.

(b) By definition, $y_{ij} \mid \theta_1, \ldots, \theta_J \sim \text{indep } N(\mu, \tau^2)$, so if $\operatorname{cov}(y_{i_1,j}, y_{i_2,j} \mid \theta_j, \sigma^2) \neq 0$ would be a contradiction. $\operatorname{cov}(y_{i_1,j}\, y_{i_2,j} \mid \mu, \tau^2, \sigma^2)$ should be positive, as

(c)

$$E(y_{ij}^2 \mid \mu, \tau^2, \sigma^2) = \int E(y_{ij}^2 \mid \theta_j, \sigma^2)\, p(\theta_j \mid \mu, \tau^2)\, d\theta_j$$
$$= \int (\sigma^2 + \theta^2)\, p(\theta_j \mid \mu, \tau^2)\, d\theta_j$$
$$= \sigma^2 + E(\theta_j^2 \mid \mu, \tau^2)$$

$$E(y_{ij} \mid \mu, \tau^2, \sigma^2) = \int E(y_{ij} \mid \theta_j, \sigma^2)\, p(\theta_j \mid \mu, \tau^2)\, d\theta_j$$
$$= E(\theta_j \mid \mu, \tau^2)$$

$$\operatorname{var}(y_{ij} \mid \mu, \tau^2, \sigma^2) = \sigma^2 + E(\theta_j^2 \mid \mu, \tau^2) + \left(E(\theta_j \mid \mu, \tau^2)\right)^2 = \sigma^2 + \operatorname{var}(\theta_j \mid \mu, \tau^2)$$

$$\Rightarrow \operatorname{var}(y_{ij} \mid \mu, \tau^2, \sigma^2) > \operatorname{var}(\theta_j \mid \mu, \tau^2)$$

$\operatorname{cov}(y_{i_1,j}, y_{i_2,j} \mid \theta_j, \sigma^2) = 0$ follows by definition

$$\operatorname{cov}(y_{i_1,j}, y_{i_2,j} \mid \mu, \tau^2, \sigma^2) = E(y_{i_1,j}\, y_{i_2,j} \mid \mu, \tau^2, \sigma^2) - E(y_{i_1,j} \mid \mu, \tau^2, \sigma^2)\, E(y_{i_2,j} \mid \mu, \tau^2, \sigma^2)$$
$$= \int E(y_{i_1,j}\, y_{i_2,j} \mid \theta_j, \sigma^2)\, p(\theta_j \mid \mu, \tau^2)\, d\theta_j - \int E(y_{i_1,j} \mid \theta_j, \sigma^2)\, p(\theta_j \mid \mu, \tau^2)\, d\theta_j$$
$$- \int E(y_{i_2,j} \mid \theta_j, \sigma^2)\, p(\theta_j \mid \mu, \tau^2)\, d\theta_j$$

$$(y_{i_1,j} \mid \theta_j, \sigma^2) \perp (y_{i_2,j} \mid \theta_j, \sigma^2) \Rightarrow E(y_{i_1,j} \mid \theta_j, \sigma^2) = E(y_{i_2,j} \mid \theta_j, \sigma^2) = \theta_j.$$

$\Rightarrow \operatorname{cov}(y_{i_1,j}, y_{i_2,j} \mid \nu, \sigma_j^2, \tau^2) = E(\theta_j^2 \mid \nu, \tau^2) - E(\theta_j \mid \nu, \tau))^2 = \operatorname{var}(\theta_j \mid \nu, \tau^2)$

which is greater than zero.

(d)

$$P(\nu \mid \theta, \sigma^2, \tau^2, y) = \frac{P(y \mid \nu, \theta, \sigma^2, \tau^2) p(\nu \mid \theta, \sigma^2, \tau^2)}{P(y \mid \theta, \sigma^2, \tau^2)}$$

$$= \frac{P(y \mid \theta, \sigma^2)}{P(y \mid \theta, \sigma^2)} P(\nu \mid \theta, \sigma^2, \tau^2)$$

$$= \frac{P(\sigma^2 \mid \nu, \theta, \tau^2) p(\nu \mid \theta, \tau^2)}{p(\sigma^2 \mid \theta, \tau^2)}$$

$$= \frac{P(\theta \mid \sigma^2, \nu, \tau^2) p(\sigma^2, \nu, \tau^2) p(\theta \mid \tau^2)}{P(\nu, \theta, \tau^2) p(\theta \mid \sigma^2, \tau^2) p(\sigma^2 \mid \tau^2)} p(\nu \mid \theta, \tau^2)$$

$$= \frac{P(\nu) p(\sigma^2) p(\tau^2)}{P(\nu) p(\sigma^2) p(\tau^2)} P(\nu \mid \theta, \tau^2)$$

$$= P(\nu \mid \theta, \tau^2)$$

This means that given the prior, the priors of ν, σ^2, and τ^2 are independent.

2.

a.

```r
library(mcmcse)

y1 <- scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school1.dat")
y2 <- scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school2.dat")
y3<-scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school3.dat")
y4<-scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school4.dat")
y5<-scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school5.dat")
y6<-scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school6.dat")
y7<-scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school7.dat")
y8<-scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/school8.dat")
ybar<-c(mean(y1),mean(y2),mean(y3),mean(y4),mean(y5),mean(y6),mean(y7),mean(y8))
n<-c(length(y1),length(y2),length(y3),length(y4),length(y5),length(y6),length(y7),length(y8))

ylist<-list(y1,y2,y3,y4,y5,y6,y7,y8)

ssrtheta<-function(ylist, theta){
  srtheta<-0
 for (i in 1:length(ylist)){
   srtheta<-srtheta + sum((ylist[[i]]-theta[i])^2)
 }
  srtheta
}

mu.update <- function(mu0, gamma0, theta, tau, J)
{
  V.mu <- 1/((1/gamma0^2)+(J/tau^2))
 mu.hat <- V.mu * ((mu0/(gamma0^2))+((J*mean(theta))/(tau^2)))
 rnorm(1, mean=mu.hat, sd=sqrt(V.mu))
}

theta.update <- function(mu, sigma, tau, J, n, ybar)
{
 V.theta <- 1/((1/tau^2)+(n/sigma^2))
 theta.hat <- V.theta * (mu/tau^2 + n*ybar/sigma^2)
 rnorm(J, mean=theta.hat, sd=sqrt(V.theta))
}

sigma.update <- function(nu0, ylist, sigma0, theta, n)
{
  df<-nu0+sum(n)
  sigma.scale<-((nu0*sigma0^2)+ssrtheta(ylist,theta))/df
  sqrt((df*sigma.scale)/rchisq(1,df))
}

tau.update <- function(tau0, eta0, J, theta, mu)
{
 df<-eta0+J
 tau.scale<-((eta0*tau0^2)+sum((theta-mu)^2))/df
 sqrt(df*tau.scale/rchisq(1,df))
}

build.chain <- function(chain.length, J, n, ybar, ylist, theta0, mu0, gamma0, nu0, sigma0, eta0, tau0)
```

```r
{
 T <- chain.length
 theta.chain <- matrix(NA, T, J)
 mu.chain <- rep(NA, T); sigma.chain <- rep(NA, T); tau.chain <- rep(NA, T);
 theta <- theta0; mu <- mu0; sigma <- sigma0; tau <- tau0; gamma<-gamma0; eta<-eta0; nu<-nu0;
 for(t in 1:T)
 {
  mu <- mu.update(mu, gamma, theta, tau, J)
  theta <- theta.update(mu, sigma, tau, J, n, ybar)
  sigma <- sigma.update(nu, ylist, sigma, theta, n)
  tau <- tau.update(tau, eta, J, theta, mu)
  theta.chain[t,] <- theta; mu.chain[t] <- mu;
  sigma.chain[t] <- sigma; tau.chain[t] <- tau;
 }
 list(theta.chain=theta.chain, mu.chain=mu.chain,
     sigma.chain=sigma.chain, tau.chain=tau.chain)
}

theta0<-ybar
mu0<-7
gamma0<-sqrt(5)
nu0<-1
sigma0<-sqrt(15)
eta0<-1
tau0<-sqrt(10)

chain<-build.chain(10000,8,n,ybar,ylist,theta0, mu0, gamma0, nu0, sigma0, eta0, tau0)

mu.chain<-chain$mu.chain
theta.chain<-chain$theta.chain
sigma.chain<-chain$sigma.chain
tau.chain<-chain$tau.chain
rm(chain)

ess(mu.chain)
```

```
## [1] 4516.284
```

```r
ess(sigma.chain)
```

```
## [1] 8614.235
```

```r
ess(tau.chain)
```

```
## [1] 2501.835
```

b.

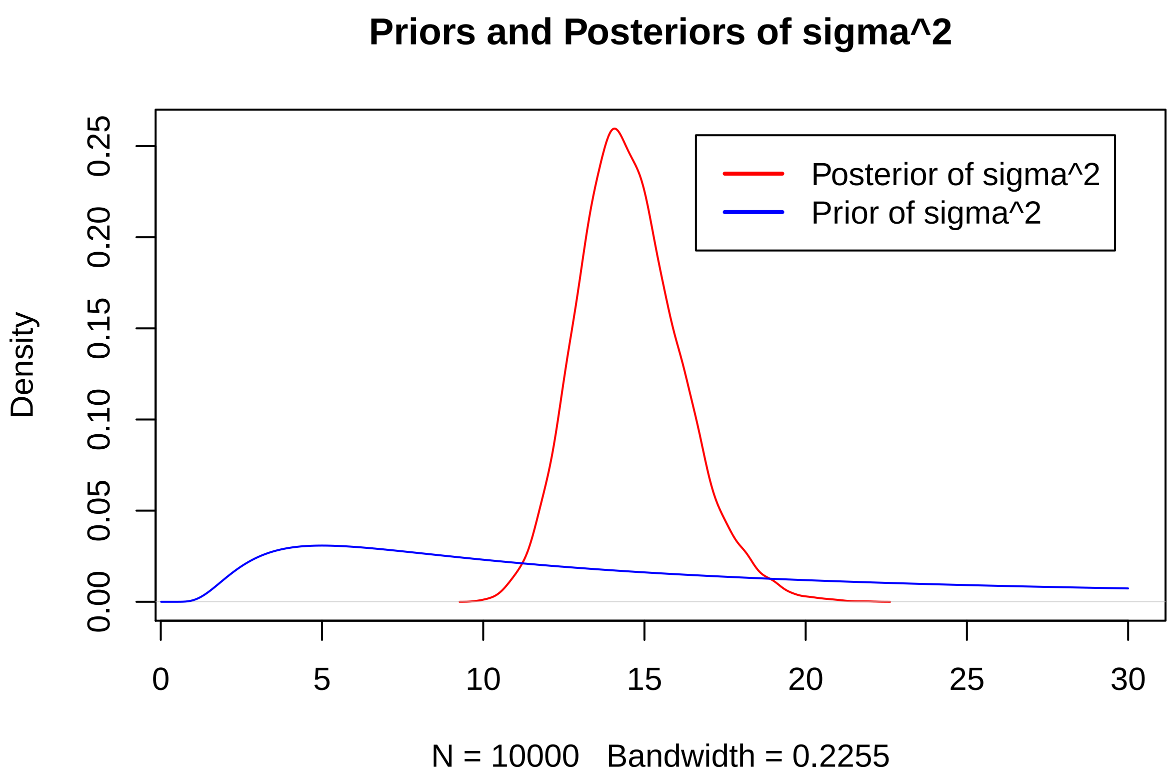
```r
quantile_table<-apply(cbind(mu.chain,sigma.chain,tau.chain)
                      ,2,function(x) quantile(x, c(0.025,0.25,0.5,0.75,0.975)))%>%
  rbind(.,apply(., 2, median))%>%
  round(.,2)
colnames(quantile_table)<-c("mu", "sigma", "tau")
rownames(quantile_table)[6] <- "Median"
quantile_table
```

```
##          mu sigma  tau
## 2.5%   6.38  3.42 0.74
```

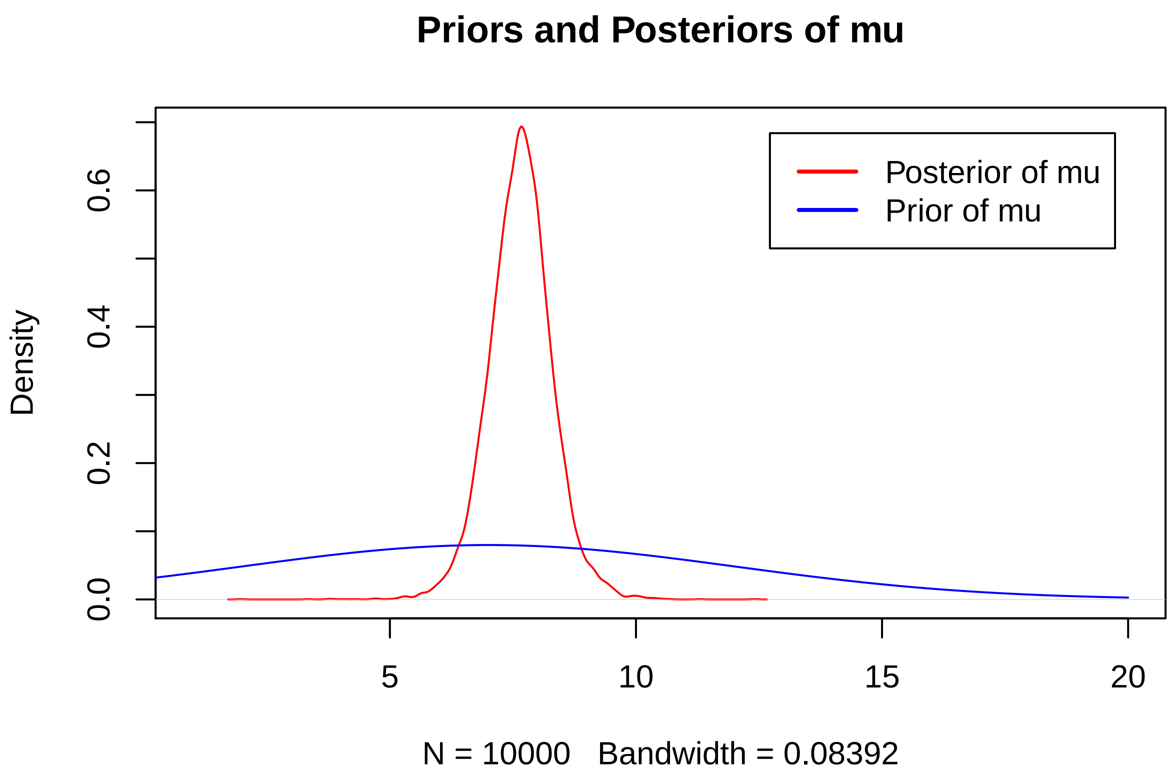
```
## 25%     7.26  3.66 1.20
## 50%     7.66  3.80 1.52
## 75%     8.05  3.94 1.93
## 97.5%   8.97  4.25 3.14
## Median 7.66  3.80 1.52
```

c.

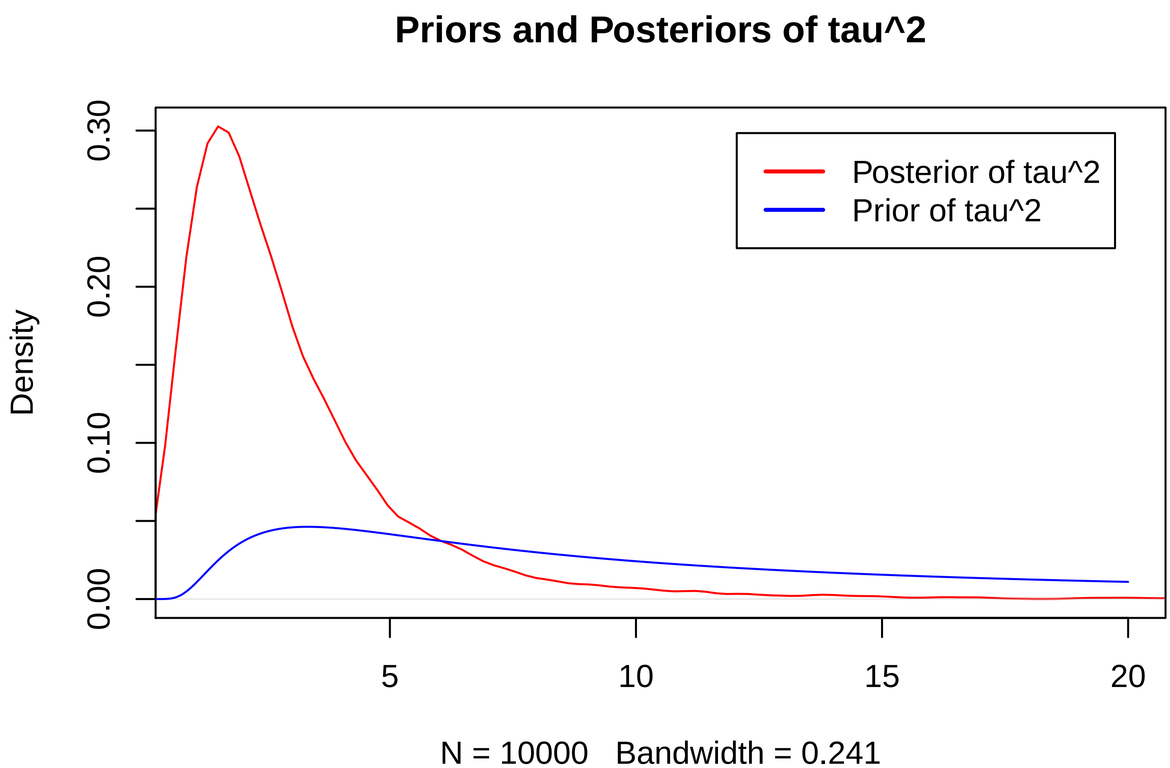
```r
sigma2.values = seq(0, 30, .01)
sigma2.prior = dchisq (1 * 15 / sigma2.values , df = 1) * 1 * 15 / sigma2.values^2
plot(density(sigma.chain^2), col = "red", xlim = c(1, 30), main = "Priors and Posteriors of sigma^2")
lines(sigma2.values , sigma2.prior , col = "blue")
legend("topright", inset =.05 , lty=1, lwd=2, col=c("red", "blue"), legend = c("Posterior of sigma^2", "Pr
```




```r
mu.values = seq(0,20,.01)
mu.prior = dnorm(mu.values, 7, 5)
plot(density(mu.chain), col = "red", xlim = c(1,20), main = "Priors and Posteriors of mu")
lines(mu.values , mu.prior , col = "blue")
legend("topright", inset =.05 , lty=1, lwd=2, col=c("red", "blue"), legend = c("Posterior of mu", "Prior o
```



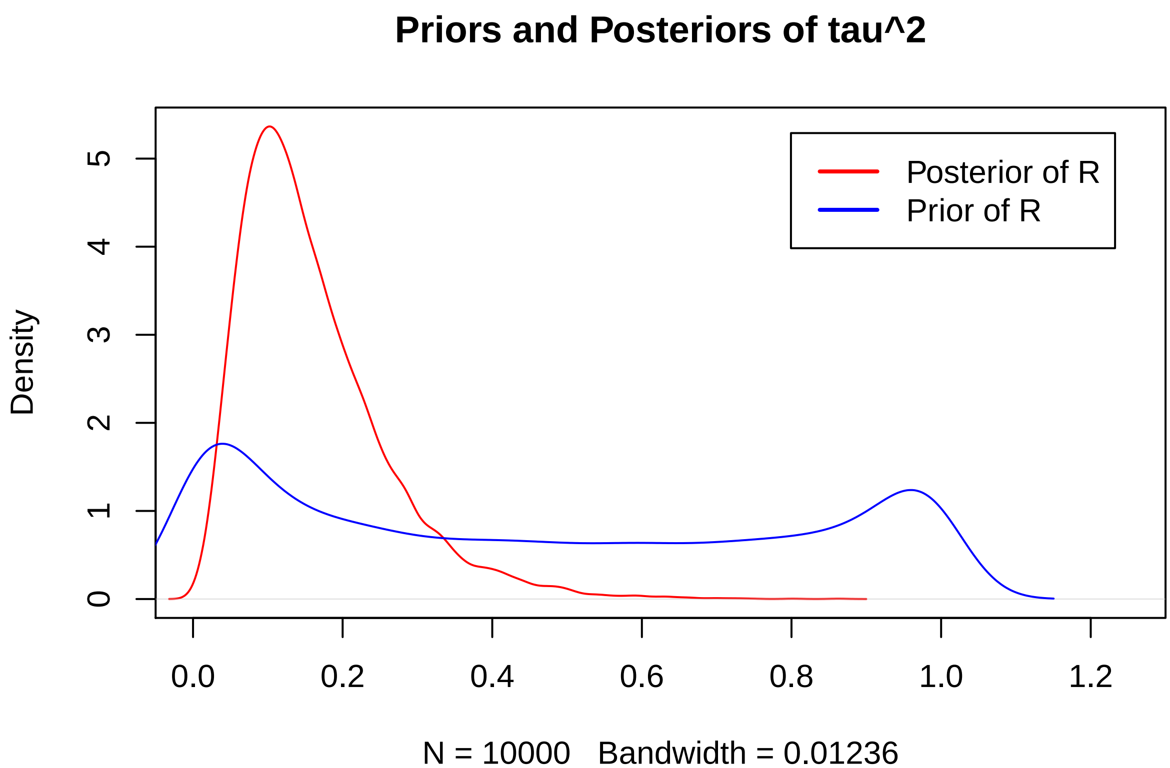

```r
tau2.values = seq(0,20,.01)
tau2.prior = dchisq (1 * 10 / tau2.values , df = 1) * 1 * 10 / tau2.values^2
plot(density(tau.chain^2), col = "red", xlim = c(1,20), main = "Priors and Posteriors of tau^2")
lines(tau2.values , tau2.prior , col = "blue")
legend("topright", inset =.05 , lty=1, lwd=2, col=c("red", "blue"), legend = c("Posterior of tau^2", "Prio
```



With more data, we get more information about where population values are likely to lie. That the prior has almost a uniform distribution for all of these parameters is indicative of our prior undecertainty.

d.

```
tau2.prior.values = 10 / rchisq (10000 , 1)
sigma2.prior.values = 15 / rchisq (10000 , 1)
R.prior = tau2.prior.values / (tau2.prior.values + sigma2.prior.values)
R.post = tau.chain ^2 / (tau.chain ^2 + sigma.chain ^2)
plot(density(R.post), col = "red", xlim = c(0,1.25), main = "Priors and Posteriors of tau^2")
lines(density(R.prior), col = "blue")
legend("topright", inset =.05 , lty=1, lwd=2, col=c("red", "blue"), legend = c("Posterior of R", "Prior of
```



The relative flatness of the prior distribution indicates our uncertainty about the magnitude between-school variation (associated with tau^2) before observing the data. The peaks in the posterior of R indicate how much more relatively certain we have become that R is going to lie between $0.0 - 0.4$ instead of seeing it as likely as any of the other values between $0.0 - 1.2$.

e.

$Pr(\theta_7 < \theta_6) = 0.5$ and $Pr(\theta_7 = min(\theta_i)) = \frac{1}{8} = 0.125$

```
mean(theta.chain[, 7] < theta.chain[, 6])
```

```
## [1] 0.5286
```

```
mean(theta.chain [,7] == pmin(theta.chain))
```

```
## [1] 0.125
```

$Pr(\theta_7 < \theta_6|y) = 0.5286$ and $Pr(\theta_7 = min(\theta_i)|y) = 0.125$

f.

```
ytilde<-apply(theta.chain,2, function(x) rnorm(10000,x,sigma.chain))

mean(ytilde[, 7] < ytilde[, 6])
```

```
## [1] 0.5068
```

```
mean(ytilde[,7] == pmin(ytilde))
```

```
## [1] 0.125
```

$Pr(\tilde{y}_7 < \tilde{y}_6|y) = 0.5$ means that if we were to randomly select a new student from schools 6 and 7, the probability that the student from school seven would have studied less hours is predicted to be 0.5. $Pr(\tilde{y}_7 = min(\tilde{y}_i)|y) = 0.125$

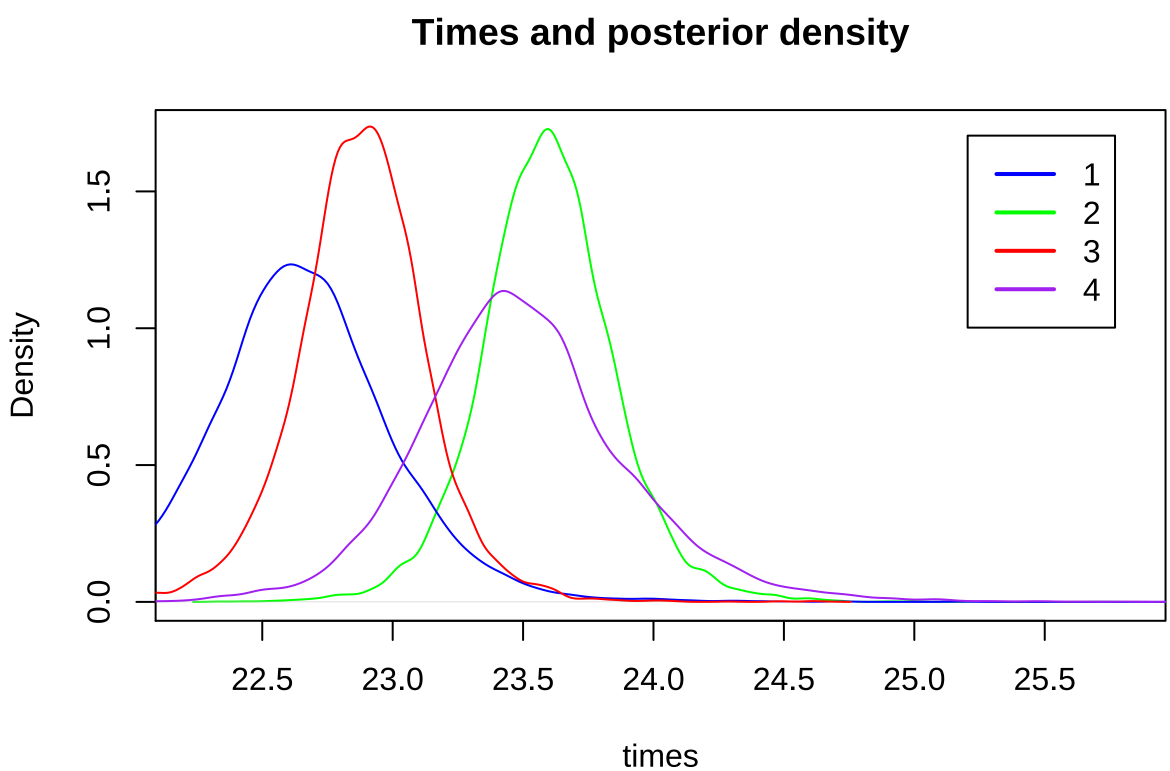
3.

```
library(MASS)
swim<-scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/swim.dat")

swimmer1<-data.frame("week" = seq(0,10,by=2), "times" = swim[1:6])
swimmer2<-data.frame("week" = seq(0,10,by=2), "times" = swim[7:12])
swimmer3<-data.frame("week" = seq(0,10,by=2), "times" = swim[13:18])
swimmer4<-data.frame("week" = seq(0,10,by=2), "times" = swim[19:24])

X<-cbind(1,swimmer1$week)


## Use B1,B2~N((23,0), (0.25,0,0.25,0))




lin.reg.gibbs<-function(Sim.Size, beta.0, Sigma.0, X, y, sigma2.0, nu.0){
  sigma2 <- sigma2.0
  T<-Sim.Size
  n<-length(y)
  Sigma0.Inv <- solve(Sigma.0)
  beta.chain <- matrix(NA, T, dim(X)[2])
  sigma2.chain <- rep(NA, T)
  for(t in 1:T)
    {
      V.beta <- solve( Sigma0.Inv + t(X) %*% X / sigma2 )
       m.beta <- V.beta %*%
      ( Sigma0.Inv %*% beta.0 + t(X) %*% y / sigma2 )
       beta <- mvrnorm(1, m.beta, V.beta)
      RSS <- sum( (y - X%*%beta)^2 )
      sigma2 <- (nu.0*sigma2.0 + RSS) / rchisq(1, nu.0+n)
      beta.chain[t,] <- beta;  sigma2.chain[t] <- sigma2
  }
   list(beta.chain=beta.chain, sigma2.chain=sigma2.chain)
}

chain1<-lin.reg.gibbs(10000,c(23,0),matrix(c(0.25,0,0,0.25),nrow = 2), X, swimmer1$times,0.25/rchisq(1, 1)
beta.chain1<-chain1$beta.chain
sigma.chain1<-chain1$sigma2.chain
chain2<-lin.reg.gibbs(10000,c(23,0),matrix(c(0.25,0,0,0.25),nrow = 2), X, swimmer2$times,0.25/rchisq(1, 1)
beta.chain2<-chain2$beta.chain
sigma.chain2<-chain2$sigma2.chain
chain3<-lin.reg.gibbs(10000,c(23,0),matrix(c(0.25,0,0,0.25),nrow = 2), X, swimmer3$times,0.25/rchisq(1, 1)
beta.chain3<-chain3$beta.chain
sigma.chain3<-chain3$sigma2.chain
chain4<-lin.reg.gibbs(10000,c(23,0),matrix(c(0.25,0,0,0.25),nrow = 2), X, swimmer4$times,0.25/rchisq(1, 1)
beta.chain4<-chain4$beta.chain
sigma.chain4<-chain4$sigma2.chain
rm(chain1,chain2,chain3,chain4)
```

```r
X.tilde<-c(1,12)
y.tilde<-matrix(NA,10000,4)
y.tilde[,1]<-rnorm(10000,beta.chain1%*%X.tilde, sigma.chain1)
y.tilde[,2]<-rnorm(10000,beta.chain2%*%X.tilde, sigma.chain2)
y.tilde[,3]<-rnorm(10000,beta.chain3%*%X.tilde, sigma.chain3)
y.tilde[,4]<-rnorm(10000,beta.chain4%*%X.tilde, sigma.chain4)
```

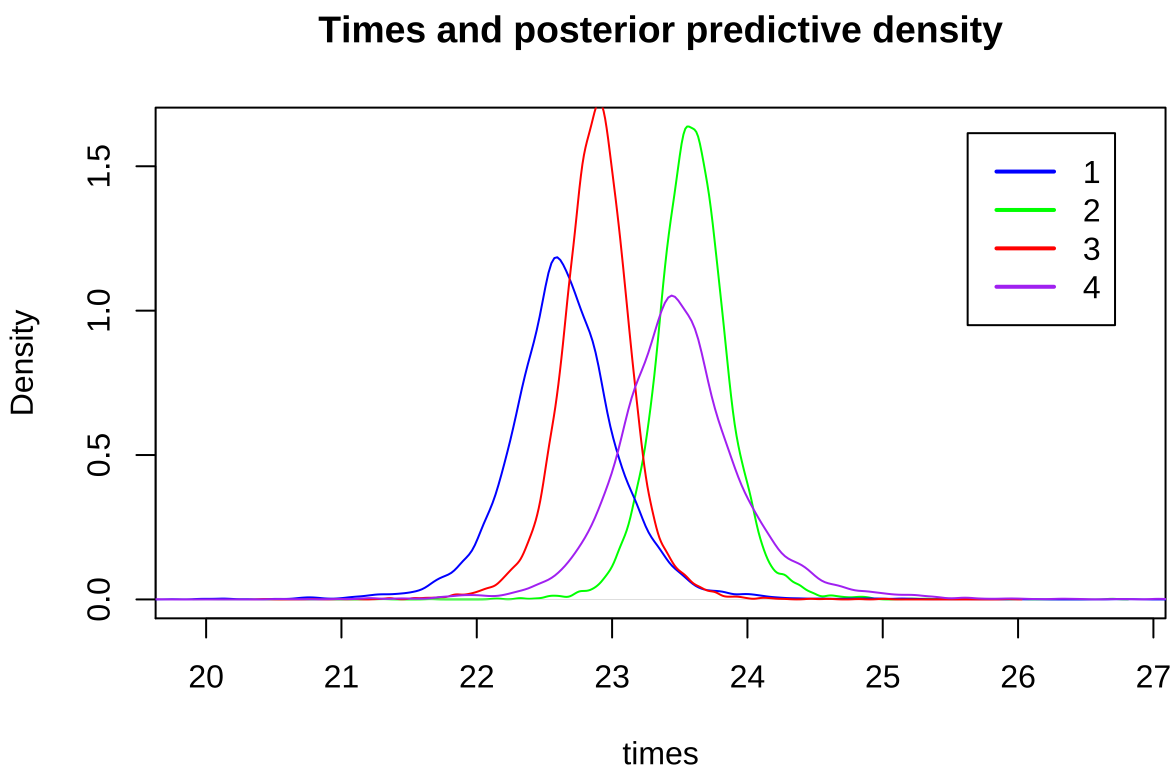
a.

```r
plot(density(beta.chain2[,1]*X.tilde[1]+beta.chain2[,2]*X.tilde[2]), col ="green", xlab = "times", main =
lines(density(beta.chain1[,1]*X.tilde[1]+beta.chain1[,2]*X.tilde[2]), col ="blue")
lines(density(beta.chain3[,1]*X.tilde[1]+beta.chain3[,2]*X.tilde[2]), col ="red")
lines(density(beta.chain4[,1]*X.tilde[1]+beta.chain4[,2]*X.tilde[2]), col ="purple")
legend("topright", inset=.05, lwd=2, lty=1,
   col=c("blue", "green", "red", "purple"),
   legend=c("1","2","3","4"))
```



b.

```r
plot(density(y.tilde[,2]), col ="green", xlab = "times", main = "Times and posterior predictive density")
lines(density(y.tilde[,1]), col ="blue")
lines(density(y.tilde[,3]), col ="red")
lines(density(y.tilde[,4]), col ="purple")
legend("topright", inset=.05, lwd=2, lty=1,
   col=c("blue", "green", "red", "purple"),
   legend=c("1","2","3","4"))
```



c.

```
library(stringr)
permprobs<-data.frame(Permutation=seq(1,24,1), Probability=seq(1,24,1))
perms<-c()
probs<-c()
y.tilde<-as.data.frame(y.tilde)
library(combinat)
```

```
##
## Attaching package: 'combinat'

## The following object is masked from 'package:utils':
##
##     combn
```

```
for (i in 1:24){
  perm<-permn(4)[i][[1]]
  orderedytilde<-data.frame(pi1 = y.tilde[,perm[1]], pi2 = y.tilde[,perm[2]], pi3 = y.tilde[,perm[3]],pi4
  relevant<-orderedytilde%>%
    filter(pi1<=pi2)%>%
    filter(pi2<=pi3)%>%
    filter(pi3<=pi4)
  perms<-c(perms,str_c(permn(4)[i][1], collapse=","))
  probs<-append(probs, nrow(relevant)/10000)
}
permprobs$Permutation<-perms
permprobs$Probability<-probs
head(permprobs[order(permprobs$Probability, decreasing = T),],5)
```

```
##      Permutation Probability
## 7  c(1, 3, 4, 2)      0.3318
```

```
## 8  c(1, 3, 2, 4)       0.2629
## 10 c(3, 1, 4, 2)       0.1251
## 9  c(3, 1, 2, 4)       0.1008
## 6  c(1, 4, 3, 2)       0.0615
```

The most likely permutation is that $\tilde{y}_1 \leq \tilde{y}_3 \leq \tilde{y}_4 \leq \tilde{y}_2$

i.

So if we were to send a single swimmer, swimmer 1 seems to be the best choice.

ii.

If we were to send two swimmers, 1 and 3 would be the best choices.

iii.

If we were to send three swimmers, 1, 3, and 4 would be the best choices.