Assignment 7

EN553.632.02

Zhiyuan Wei
zwei26

Problem 1

1. $Y_{ij} = \mu + \alpha_i + \epsilon_{ij}$, $i=1,\dots,I$, $j=1,\dots,J$.

Then $Y_{ij} \sim N(\mu+\alpha_i, \sigma_\epsilon^2)$

$$P(Y|\alpha_1,\dots,\alpha_I,\mu,\sigma_\epsilon^2) = \prod^I \prod^J P(y_{ij}|\alpha_i,\mu,\sigma_\epsilon^2)$$

$$= \prod \prod \frac{1}{\sqrt{2\pi}\sigma_\epsilon} \exp\left(-\frac{(y_{ij}-\mu-\alpha_i)^2}{2\sigma_\epsilon^2}\right)$$

$$\propto (\sigma_\epsilon)^{-IJ} \exp\left(\frac{-\sum^I\sum^J (y_{ij}-\mu-\alpha_i)^2}{2\sigma_\epsilon^2}\right)$$

$P(\mu|\alpha_1,\dots,\alpha_I,\sigma_\epsilon^2,Y) \propto \prod^I\prod^J P(y_{ij}|\alpha_i,\mu,\sigma_\epsilon^2)$, for μ has a flat prior

$$\propto \exp\left(\frac{-\sum^I\sum^J (y_{ij}-\mu-\alpha_i)^2}{2\sigma_\epsilon^2}\right)$$

$$\propto \exp\left(-\tfrac{1}{2}(a\mu^2 - 2b\mu + c)\right)$$

$$\sim \text{Normal}\left(\mu; \frac{b}{a}, \frac{1}{a}\right).$$

for $a = \frac{IJ}{\sigma_\epsilon^2}$, $b = \frac{\sum^I\sum^J y_{ij} - J\sum^I \alpha_i}{\sigma_\epsilon^2}$

We have $\boxed{P(\mu|\alpha_1,\dots,\alpha_I,\sigma_\epsilon^2,Y) \sim N\left(\bar{y}-\bar{\alpha}, \frac{\sigma_\epsilon^2}{IJ}\right)}$

$P(\alpha_i|\mu,\sigma_\epsilon^2,Y) \propto \prod^J P(ij|\alpha_i,\mu,\sigma_\epsilon^2)\cdot P(\alpha_i)$

$$\propto \exp\left(\frac{-\sum^I\sum^J (y_{ij}-\mu-\alpha_i)^2}{2\sigma_\epsilon^2}\right)\exp\left(\frac{\alpha_i^2}{-2\sigma_\alpha^2}\right), \text{ for } \alpha_i \sim N(0,\sigma_\alpha^2)$$

$$\propto \exp\left(\frac{\sigma_\epsilon^2\alpha_i^2 + \sigma_\alpha^2\sum^J (y_{ij}-\mu-\alpha_i)^2}{2\sigma_\alpha^2\sigma_\epsilon^2}\right)$$

$$\propto \exp\left(-\tfrac{1}{2}(a\alpha_i^2 - 2b\alpha_i + c)\right)$$

$$\sim \text{Normal}\left(\alpha_i; \frac{b}{a}, \frac{1}{a}\right)$$

for $a = \frac{J}{\sigma_\epsilon^2} + \frac{1}{\sigma_\alpha^2}$, $b = \frac{\sum^J y_{ij} - J\mu}{\sigma_\epsilon^2}$

So we have $\boxed{P(\alpha_i|\mu,\sigma_\epsilon^2,Y) \sim N\left(\frac{\frac{J(\bar{y}_i-\mu)}{\sigma_\epsilon^2}}{\frac{J}{\sigma_\epsilon^2}+\frac{1}{\sigma_\alpha^2}}, \frac{1}{\frac{J}{\sigma_\epsilon^2}+\frac{1}{\sigma_\alpha^2}}\right)}$

$$P(\sigma_\epsilon^2 \mid \alpha_1, \dots \alpha_I, \mu, Y) \propto \prod^I \prod^J P(y_{ij} \mid \alpha_i, \mu, \sigma_\epsilon^2) P(\sigma_\epsilon^2)$$

$$\propto (\sigma_\epsilon^2)^{-IJ} \exp\left(-\frac{\sum^I \sum^J (y_{ij}-\mu-\alpha_i)^2}{2\sigma_\epsilon^2}\right) b^a \frac{1}{\Gamma(a)} (\sigma_\epsilon^2)^{-a+1} \exp\left(-\frac{b}{\sigma_\epsilon^2}\right)$$

, for $\sigma_\epsilon^2 \sim IG(a, b)$

$$\propto (\sigma_\epsilon^2)^{-(a+\frac{IJ}{2})+1} \exp\left(-\left(b+\frac{\sum^I\sum^J (y_{ij}-\mu-\alpha_i)^2}{2}\right)\frac{1}{\sigma_\epsilon^2}\right)$$

Thus, $\boxed{P(\sigma_\epsilon^2 \mid \alpha_1, \dots, \alpha_I, \mu, Y) \sim \text{InverseGamma}\left(a+\frac{IJ}{2}, b+\frac{\sum^I\sum^J (y_{ij}-\mu-\alpha_i)^2}{2}\right)}$

2. Convergence diagnosis is to check whether the MCMC algorithm converged at certain iteration T so that the output can be the ~~true~~ simulation of the true distribution for all $t > T$. Otherwise, the simulation will be baised.

We can use traceplot or autocorrelation plot to check for convergence.

If the sampler is suffering from slow convergence, we can either discard some parameters that ~~are~~ have high computation cost or design a better model to reduce the autocorrelations.

3. Conditions that might weaken the identifiability of the parameter can be a biased prior of μ and low group size J that ~~cause not enou~~ also cause limited information.

4. For two parametrizations:

We have (i) (μ, α): $\mu | \text{rest} \sim N\left(\bar{y} - \bar{\alpha}, \frac{\sigma_\epsilon^2}{IJ}\right)$

$$\alpha_i | \text{rest} \sim N\left(\frac{\frac{J(\bar{y}_i - \mu)}{\sigma_\epsilon^2}}{\frac{J}{\sigma_\epsilon^2} + \frac{1}{\sigma_\alpha^2}}, \frac{1}{\frac{J}{\sigma_\epsilon^2} + \frac{1}{\sigma_\alpha^2}}\right).$$

(ii) (μ, η): $\mu | \text{rest} \sim N\left(\bar{\eta}, \frac{\sigma_\alpha^2}{I}\right)$

$$\alpha_i | \text{rest} \sim N\left(\frac{\frac{J\bar{y}_i}{\sigma_\epsilon^2} + \frac{\mu}{\sigma_\alpha^2}}{\frac{J}{\sigma_\epsilon^2} + \frac{1}{\sigma_\alpha^2}}, \frac{1}{\frac{J}{\sigma_\epsilon^2} + \frac{1}{\sigma_\alpha^2}}\right)$$

We use the following above to implement Gibbs samplers, We have ~~that~~ both ~~parametrizat~~ parametrizations, and ~~perform~~ the result of autocorrelation plots show that the performances for two parameter spaces are similar

5. We implemented the Gibbs samplers above again, but with $\sigma_\alpha^2 = 10$. ~~instea~~
The result shows that with (μ, η), we have a better performance than (μ, α)
And this suggests that with large value of σ_α^2, hierarchical centering reparametrizations are prefered.

Assignment 7 (2)

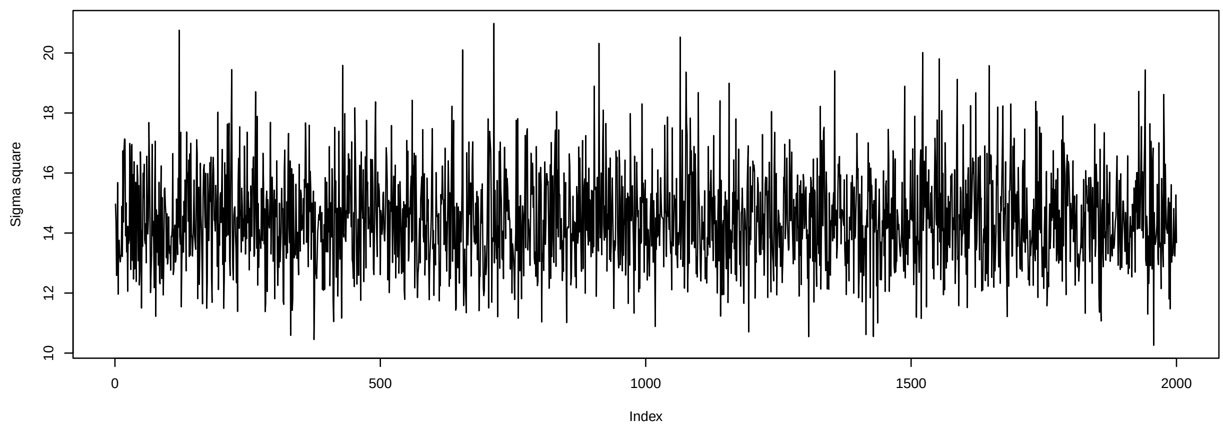
8.3

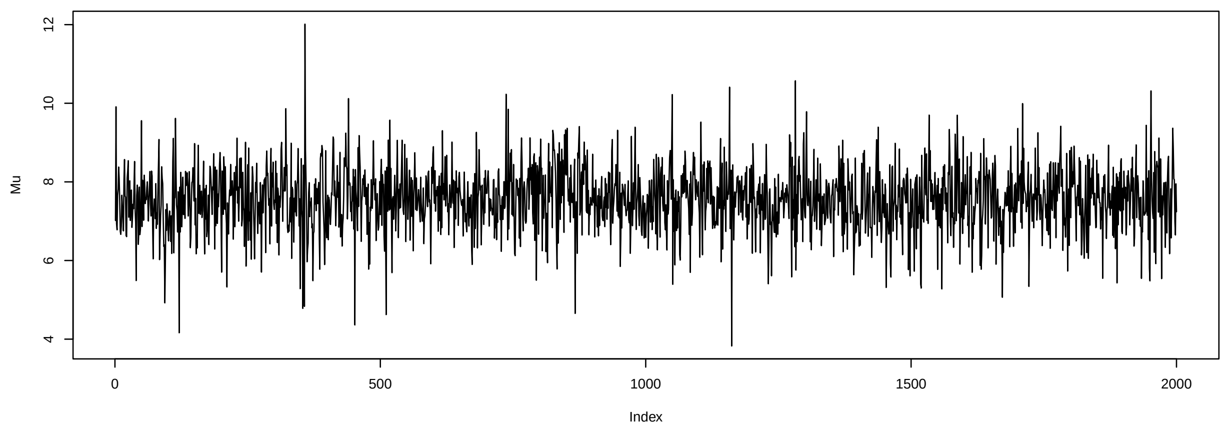
```
library(dplyr)
library(tidyr)
library(MCMCpack)
library(coda)
library(MASS)
schools.list = lapply(1:8, function(i) {
  f = paste("school",i,".dat",sep="")
  w = read.table(f)

  data.frame(
    school = i,
    hours = w[, 1] %>% as.numeric
  )
})
Y = do.call(rbind, schools.list)
```

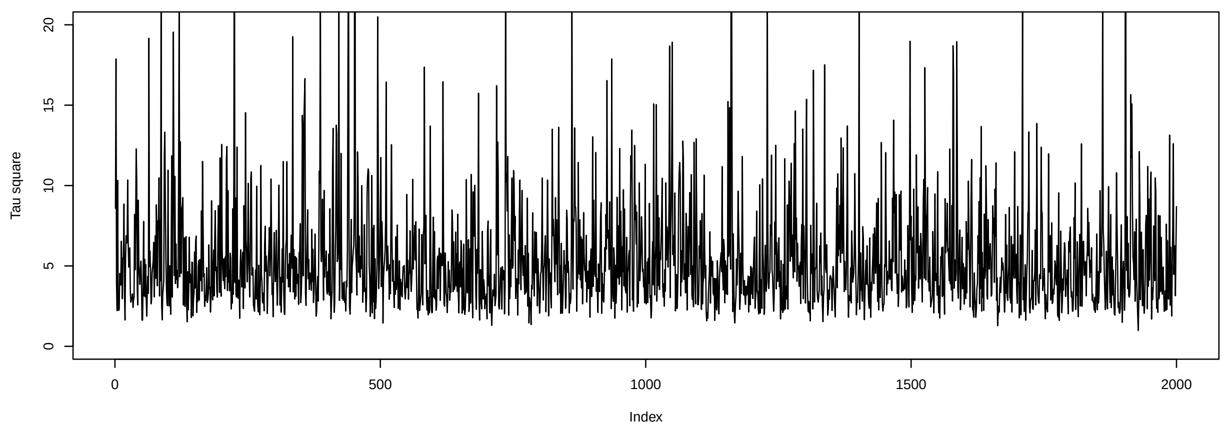
(a)

```
# Prior
mu0 = 7
g0_square = 5
tau0_square = 10
eta0 = 2
sigma0_square = 15
nu0 = 2
m=8
# Starting values
n = sample_var = ybar = rep(NA, m)
for (i in 1:m) {
  Y_i = Y[Y[, 1] == i, 2]
  ybar[i] = mean(Y_i)
  sample_var[i] = var(Y_i)
  n[i] = length(Y_i)
}
theta = ybar
sigma2 = mean(sample_var)
mu = mean(theta)
tau2 = var(theta)
#Gibbs
S = 2000
THETA = matrix(nrow = S, ncol = m)
SMT = matrix(nrow = S, ncol = 3)
colnames(SMT) = c('sigma2', 'mu', 'tau2')
for (s in 1:S) {
```

```r
  # Sample theta
  for (j in 1:m) {
    vtheta = 1 / (n[j] / sigma2 + 1 / tau2)
    etheta = vtheta * (ybar[j] * n[j] / sigma2 + mu / tau2)
    theta[j] = rnorm(1, etheta, sqrt(vtheta))
  }
  # Sample sigma square
  nun = nu0 + sum(n)
  ss = nu0 * sigma0_square
  for (j in 1:m) {
    ss = ss + sum((Y[Y[, 1] == j, 2] - theta[j])^2)
  }
  sigma2 = 1 / rgamma(1, nun / 2, ss / 2)
  # Sample mu
  vmu = 1 / (m / tau2 + 1 /g0_square)
  emu = vmu * (m * mean(theta) / tau2 + mu0 / g0_square)
  mu = rnorm(1, emu, sqrt(vmu))
  # Sample tau square
  etam = eta0 + m
  ss = eta0 * tau0_square + sum((theta - mu)^2)
  tau2 = 1 / rgamma(1, etam / 2, ss / 2)

  THETA[s, ] = theta
  SMT[s, ] = c(sigma2, mu, tau2)
}
par(mfrow = c(3,1))
plot(SMT[,1],type = "l",ylab = "Sigma square")
plot(SMT[,2],type = "l",ylab = "Mu")
plot(SMT[,3],type = "l",ylab = "Tau square",ylim = c(0,20))
```








```r
effectiveSize(SMT[, 1])
```

```
## var1
## 2000
```

```r
effectiveSize(SMT[, 2])
```

```
##    var1
## 1688.83
```

```r
effectiveSize(SMT[, 3])
```

```
##     var1
## 1397.734
```

We can see from the trace plot that our Markov Chain is stationary and converge to certain value instead of bouncing up and down. And we ran the chain long enough since we can see that the effective sizes for σ^2, μ, and τ^2 are all above 1000.

(b)

```r
#posterior means of sigma square
mean(SMT[,1])
```

```
## [1] 14.45891
```

```r
#95% confidence region for sigma square
quantile(SMT[,1],prob=c(0.025,0.975))
```

```
##     2.5%    97.5%
## 11.68156 17.80508
```

```r
#posterior means of mu
mean(SMT[,2])
```

```
## [1] 7.55811
```

```r
#95% confidence region for mu
quantile(SMT[,2],prob=c(0.025,0.975))
```

```
##     2.5%    97.5%
## 5.895801 9.093516
```

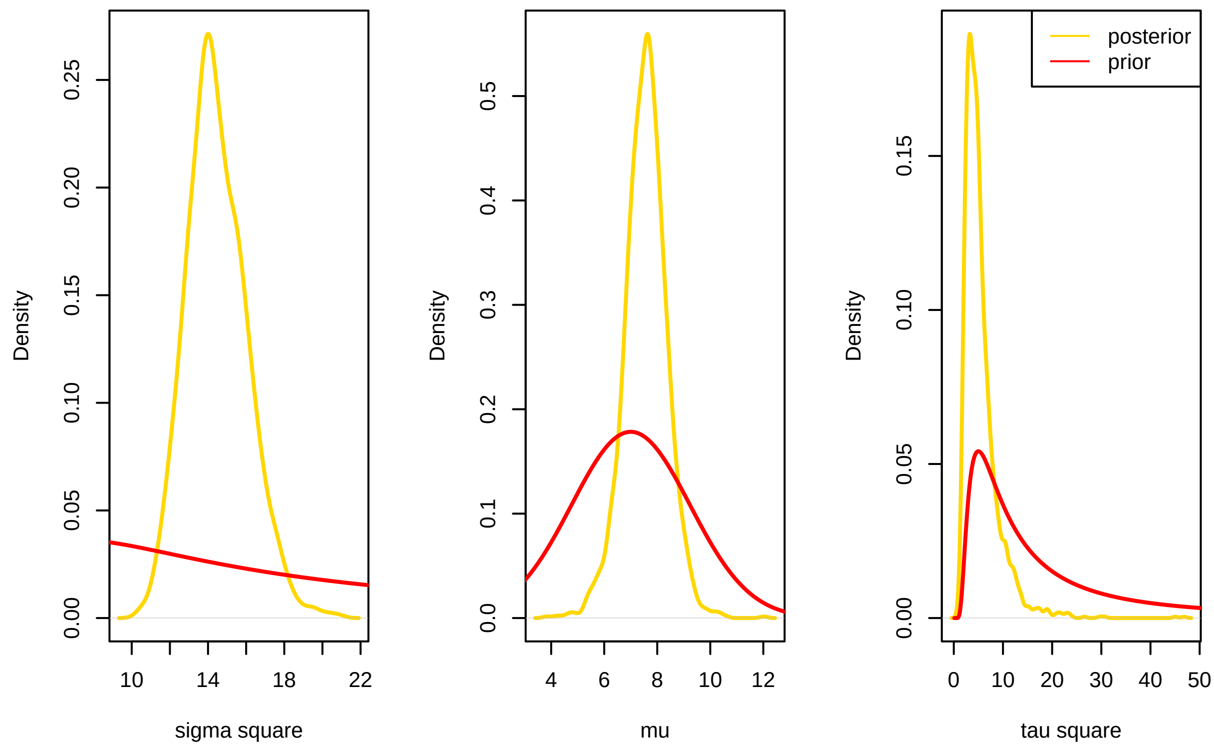
```r
#posterior means of tau square
mean(SMT[,3])
```

```
## [1] 5.477331
```

```r
#95% confidence region for tau square
quantile(SMT[,3],prob=c(0.025,0.975))
```

```
##      2.5%     97.5%
##  1.885764 14.077872
```

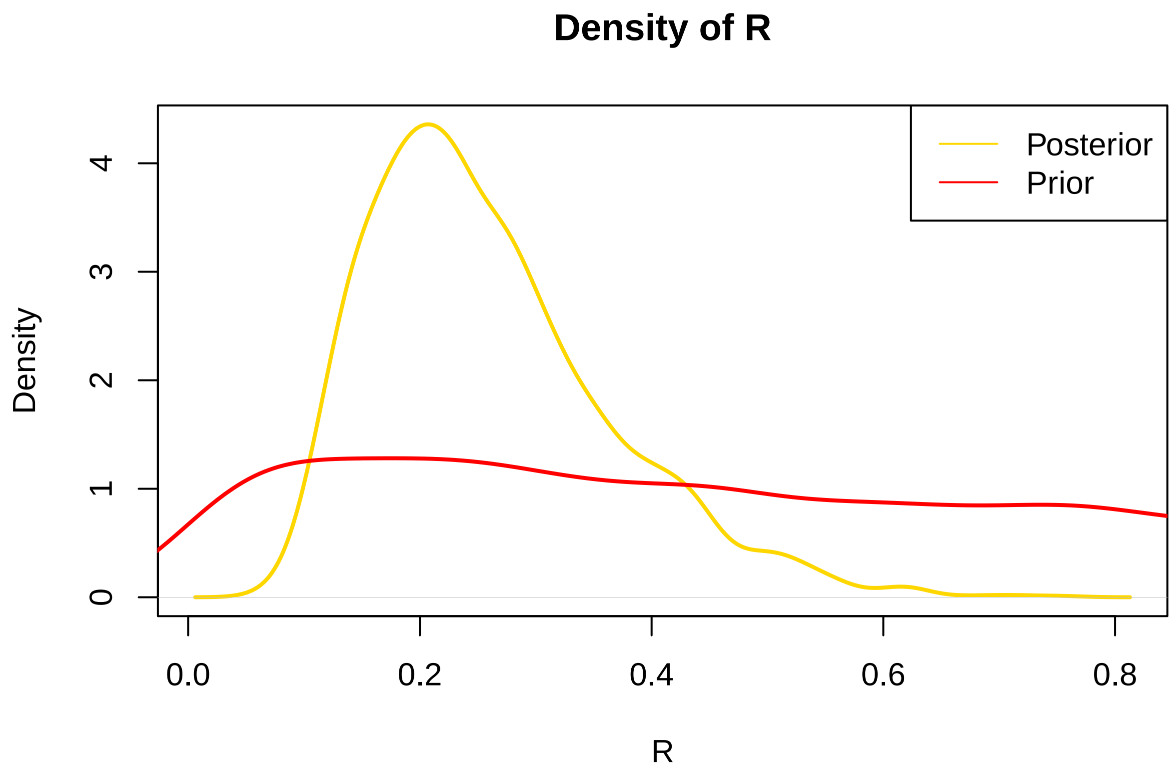
```r
par(mfrow = c(1,3))
seq1 = seq(0.1, 100, by = 0.1)
plot(density(SMT[,1]),main = "",xlab="sigma square",col="gold",lwd=2)
lines(seq1, dinvgamma(seq1, nu0/2, nu0*sigma0_square/2),col="red",lwd=2)

plot(density(SMT[,2]),main = "",xlab="mu",col="gold",lwd=2)
lines(seq1, dnorm(seq1, mu0, sqrt(g0_square)),col="red",lwd=2)

plot(density(SMT[,3]),main = "",xlab="tau square",col="gold",lwd=2)
lines(seq1, dinvgamma(seq1, eta0/2, eta0*tau0_square/2),col="red",lwd=2)
legend('topright', lty = 1, legend = c('posterior', 'prior'),col=c("gold","red"))
```



The posterior means and 95% confidence regions for σ^2, μ, and τ^2 are above. We can also see from the plots that our prior beliefs for σ^2, μ, and τ^2 are more widespread, and the densities for posterior show a more certain update on our beliefs. All three parameters changed a lot visually, while the distribution for σ^2 is the farthest from out prior belief.

(C)

```
r_post = SMT[,3]/(SMT[,1]+SMT[,3])
tau0_square_sample = 1/rgamma(S, eta0/2, eta0*tau0_square/2)
sigma0_square_sample = 1/rgamma(S, nu0/2, nu0*sigma0_square/2)
r_prior = tau0_square_sample/(sigma0_square_sample+tau0_square_sample)
plot(density(r_post), col = "gold", xlab = "R",main = 'Density of R',lwd=2)
lines(density(r_prior), col = "red",lwd=2)
legend('topright', lty = 1, col = c("gold", "red"), legend = c('Posterior', 'Prior'))
```




```
mean(r_post)
```

```
## [1] 0.2589224
```

R is the variation between schools over total variation. And we found out that around 26% of the variation is between-school variation.

(d)

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

```
## [1] 0.531
```

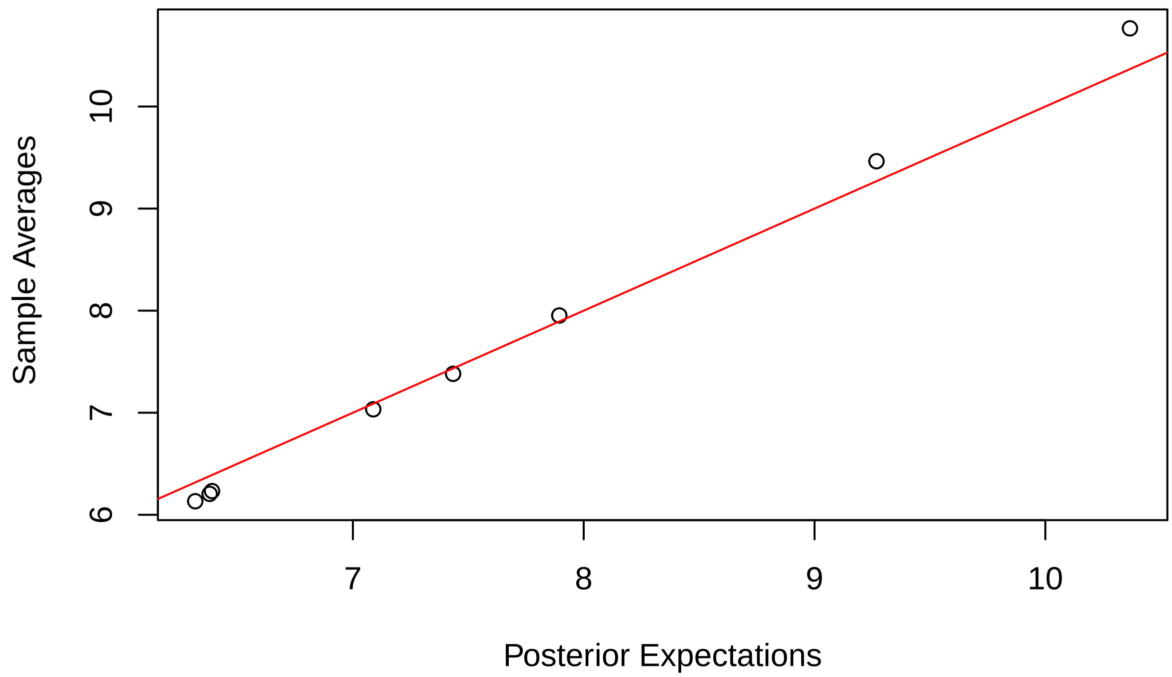
```
mean(apply(THETA, 1, which.min) == 7)
```

```
## [1] 0.322
```

We can see that $P(\theta_7 < \theta_6|rest) \approx 0.52$ and $P(\theta_7 = min(\theta)|rest) \approx 0.33$.

(e)

```
post_expect = colMeans(THETA)
plot(post_expect,ybar,pch=1,xlab = "Posterior Expectations",ylab = " Sample Averages")
abline(a = 0, b = 1,col="red")
```




```r
#sample mean of all observations
mean(Y[, 2])
```

```
## [1] 7.691278
```

```r
#posterior mean of mu
mean(SMT[, 2])
```

```
## [1] 7.55811
```

Our sample mean of all observations is around 7.691 and posterior mean of μ is around 7.546. From the plot, we can see that there is a strong relationship between the sample averages and the posterior expectations. Also, schools with high or low sample averages tend to pull away from the posterior expectations while schools with sample averages close to the global mean have a less difference between its group posterior expectations and group sample average.