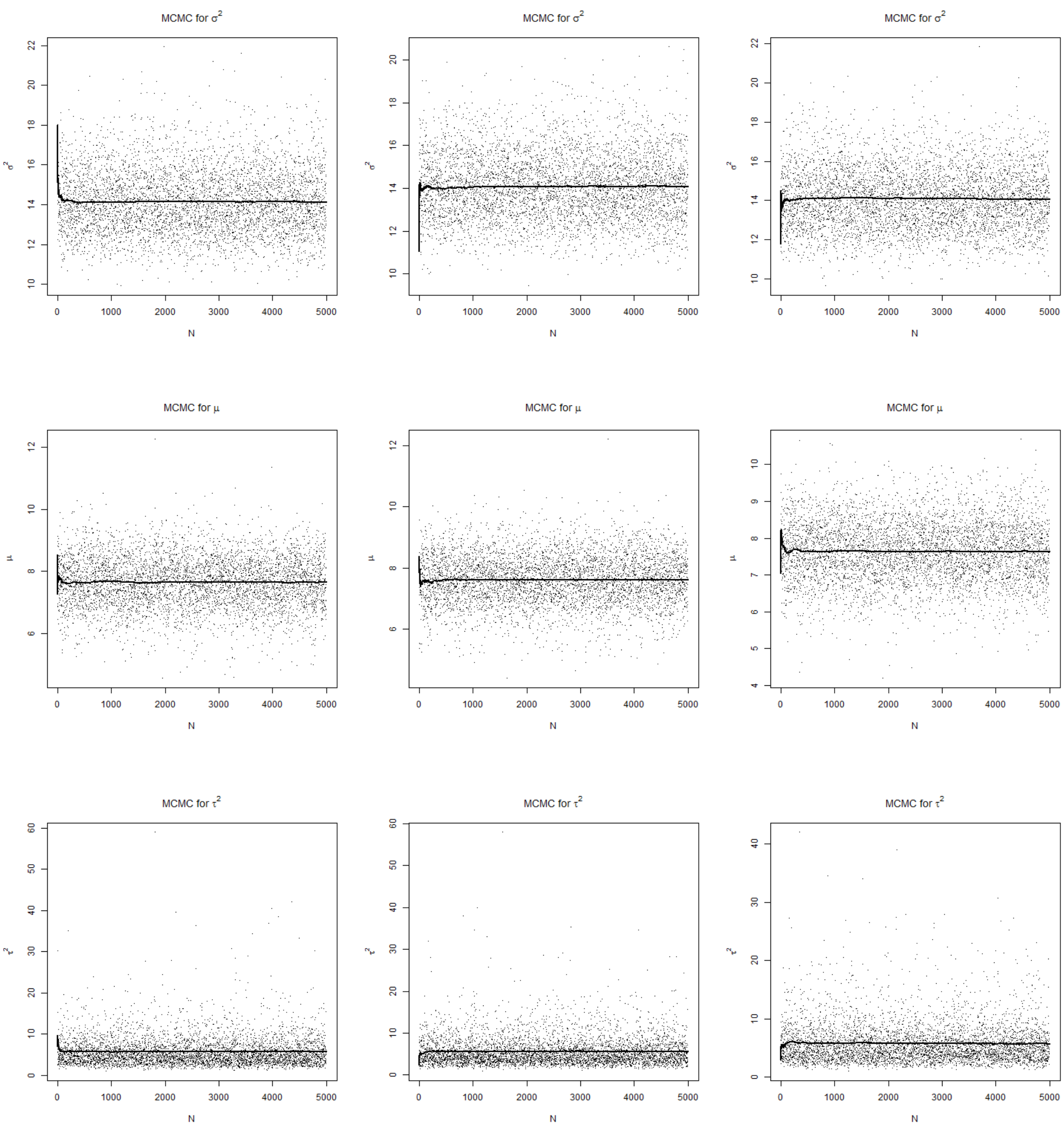
STA 360: Assignment 8

Michael Lin

March 24, 2015

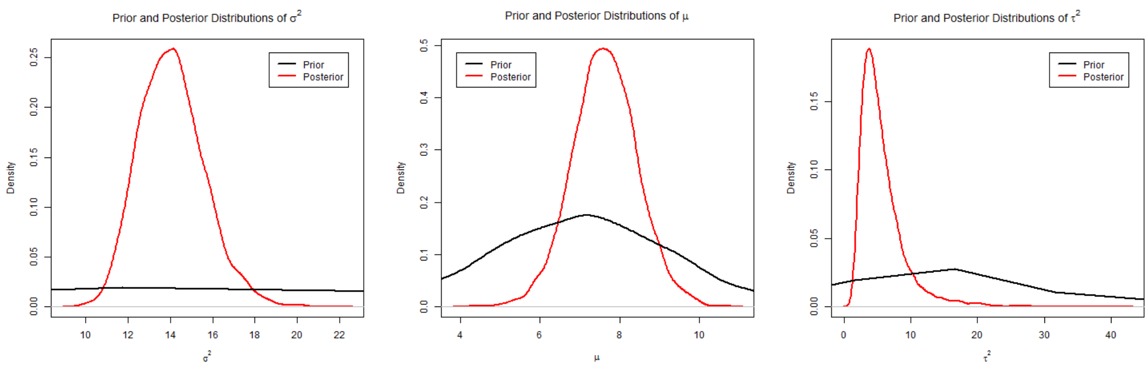
8.3 (a) Below are plots of σ^2, μ, τ^2 for 3 different runs.



(b) Posterior means and 95% confidence regions:

	Posterior Mean	95% Credible Interval
σ^2	14.0755	[11.32467, 17.49859]
μ	7.645934	[6.065964, 9.204791]
τ^2	5.640306	[1.950527, 15.128655]

Plots of prior and posterior densities are also shown below:

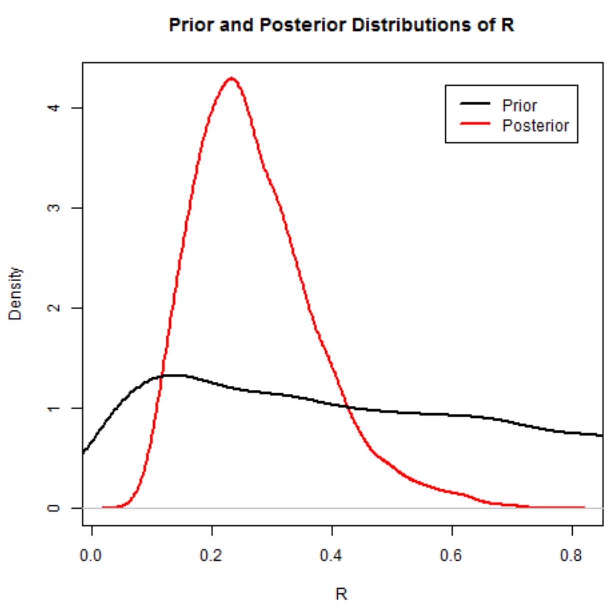


The data allowed us to obtain posterior distributions with smaller spread than the respective prior distributions. In particular, the prior distribution of σ^2 is uninformative while its posterior distribution suggests a mean of around 14. The posterior mean of μ is greater than prior mean while the posterior mean of τ^2 is slightly less than the corresponding prior mean. This implies that the mean of all observations is the variance between groups is larger than prior belief while the population variance is slightly smaller than prior belief.

(c) Defining R as follows:

$$R = \frac{\tau^2}{\sigma^2 + \tau^2}$$

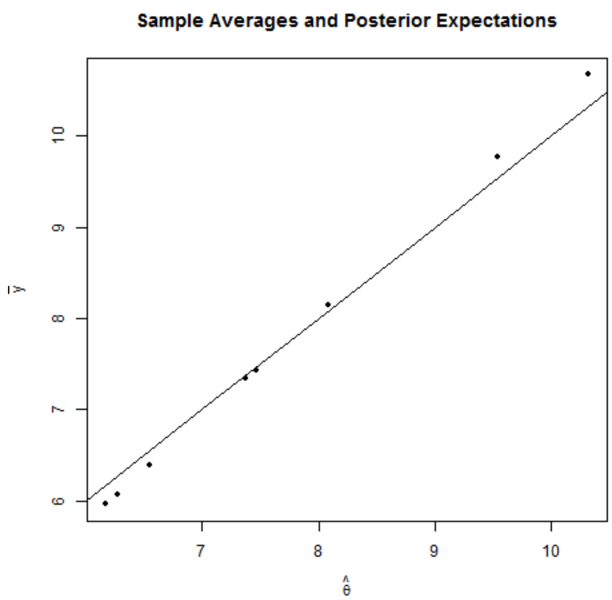
below is a plot of the prior and posterior density of R:



The prior for R is close to a uniform distribution, which suggested little prior belief that between-school variation differs from within-school variation. However, the posterior distribution has most of the mass around 0.2 and 0.4, which suggests that there is less variance between groups than within groups.

(d) The posterior probability that θ_7 is smaller than θ_6 is 0.6006. The posterior probability that θ_7 is the smallest of all the θ's is 0.339.

(e) Below is a plot of the sample averages $\bar{y}_1, \ldots, \bar{y}_8$ against the posterior expectations $\theta_1, \ldots, \theta_8$:



The plot demonstrates the effect of shrinkage: smaller-than-average $\bar{y}$ values are predicted to have slightly higher predicted means, while the opposite is true for larger-than-average $\bar{y}$ values. In other words, the slope of the line passing through the data points is greater than the slope of the line shown. The sample mean of all observations is 7.766628, which is slightly larger than the posterior mean of μ which is 7.645934.

R code for 8.3:

```
## Load Data ##
library(MCMCpack)
school1 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school1.dat", se
school2 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school2.dat", se
school3 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school3.dat", se
school4 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school4.dat", se
school5 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school5.dat", se
school6 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school6.dat", se
school7 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school7.dat", se
school8 = read.table("http://www.stat.washington.edu/people/pdhoff/Book/Data/hwdata/school8.dat", se

##### Part (a) #####
## Place Data in List $$
m = 8
school = list()
for(i in 1:m){
  school[[i]] = eval(parse(text=paste("school",i,sep="")))
}


## Define parameters and priors ##
n = rep(NA, m)
for(i in 1:m){
  n[i]=eval(parse(text=paste("dim(school",i,")[1]",sep="")))
}

mu0 = 7; gam0.2 = 5;
tau0.2 = 10; eta0 = 2;
sig0.2 = 15; nu0 = 2;

## Starting values ##
ybar = rep(NA, m)
sv = rep(NA, m)
for(i in 1:m){
  ybar[i] = mean(school[[i]][,1])
  sv[i] = var(school[[i]][,1])
}
theta = ybar; sig.2 = mean(sv);
mu = mean(theta); tau.2 = var(theta)

## Setup MCMC ##
N = 5000
THETA = NULL
MU = TAU.2 = SIG.2 = rep(NA, N)

## MCMC ##
for(r in 1:N){
  mu = rnorm(1, mean = (m*mean(theta)/tau.2 + mu0/gam0.2)/(m/tau.2 + 1/gam0.2),
             sd = sqrt(1/(m/tau.2 + 1/gam0.2)))

  tau.2 = 1/rgamma(1, shape = (eta0+m)/2,
                   rate = (eta0*tau0.2 + sum((theta-mu)^2))/2)

```

```
  for(i in 1:m){
    theta[i] = rnorm(1, mean = (n[i]*ybar[i]/sig.2 + mu/tau.2)/(n[i]/sig.2 + 1/tau.2),
                     sd = sqrt(1/(n[i]/sig.2 + 1/tau.2)))
  }

  dsum = 0;
  for(j in 1:m){
    for(i in 1:n[j]){
      dsum = dsum + (school[[j]][i,1] - theta[j])^2
    }
  }

  sig.2 = 1/rgamma(1, shape = (nu0 + sum(n))/2, rate = 0.5*(nu0*sig0.2 + dsum))

  ## Save Update ##
  MU[r] = mu
  TAU.2[r] = tau.2
  THETA = cbind(THETA, theta)
  SIG.2[r] = sig.2
}

## Running Sum and Average ##
MU.sum = TAU.2.sum = SIG.2.sum = rep(NA, N)
MU.avg = TAU.2.avg = SIG.2.avg = rep(NA, N)
THETA.sum = THETA.avg = matrix(nrow = 8, ncol = N)
MU.sum[1] = MU.avg[1] = MU[1]
TAU.2.sum[1] = TAU.2.avg[1] = TAU.2[1]
SIG.2.sum[1] = SIG.2.avg[1] = SIG.2[1]
THETA.sum[,1] = THETA[,1]


for(i in 2:N){
  MU.sum[i] = MU[i] + MU.sum[i-1]
  MU.avg[i] = MU.sum[i]/i

  TAU.2.sum[i] = TAU.2[i] + TAU.2.sum[i-1]
  TAU.2.avg[i] = TAU.2.sum[i]/i

  SIG.2.sum[i] = SIG.2[i] + SIG.2.sum[i-1]
  SIG.2.avg[i] = SIG.2.sum[i]/i

  THETA.sum[,i] = THETA[,i] + THETA.sum[,i-1]
  THETA.avg[,i] = THETA.sum[,i]/i
}

## Traceplot and Running Averages ##
x = 1:N
plot(x, MU, pch = '.', xlab = 'N', ylab = expression(mu),
     main = expression("MCMC for"~mu))
lines(x, MU.avg, lwd = 2)

title.vec = c("mu","sigma^{2}","tau^{2}")
datatitle.vec = c("MU","SIG.2","TAU.2")
avgtitle.vec = c("MU.avg","SIG.2.avg","TAU.2.avg")
```

```
filetitle.vec = c("mu","sig","tau")

titles = NULL
titles = cbind(titles, datatitle.vec, title.vec, avgtitle.vec, filetitle.vec)


for(i in 1:length(title.vec)){
  temp = parse(text=paste("expression(’MCMC for’~",titles[i,2],")", sep=""))
  png(paste(titles[i,4],"-b.png",sep=""))
  plot(x,eval(parse(text=titles[i,1])), pch = ".", xlab = ’N’,
       ylab = parse(text=titles[i,2]), main = eval(temp))
  lines(x,eval(parse(text=titles[i,3])), lwd = 2)
  dev.off()
}


##### Part (b) #####
## Posterior mean ##
mu.postmean = mean(MU)
sig.2.postmean = mean(SIG.2)
tau.2.postmean = mean(TAU.2)

## 95% credible interval ##
print(quantile(MU, c(0.025, 0.975)))
print(quantile(SIG.2, c(0.025, 0.975)))
print(quantile(TAU.2, c(0.025, 0.975)))

## Obtain prior and posterior ##
x.mu = seq(4,10,0.01)
x.sig = seq(0,40,0.01)
x.tau = seq(0,40,0.01)

## Sample from priors and posterior ##
mu.prior = rnorm(1000, mu0, sqrt(gam0.2))
mu.post = MU
sig.2.prior = rinvgamma(1000, shape = nu0/2, scale = nu0*sig0.2/2)
sig.2.post = SIG.2
tau.2.prior = rinvgamma(1000, shape = eta0/2, scale = eta0*tau0.2/2)
tau.2.post = TAU.2

## Plot prior and posterior ##
png("mu-density.png")
plot(density(mu.post), col = "red", lwd = 2,
     xlab = expression(mu), ylab = "Density",
     main = expression("Prior and Posterior Distributions of"~mu))
lines(density(mu.prior), lwd = 2)
legend("topleft",lty=c(1,1),lwd=c(2,2),c("Prior","Posterior"),
       col=c("black","red"),inset=0.05)
dev.off()

png("sig-density.png")
plot(density(sig.2.post), col = "red", lwd = 2,
     xlab = expression(sigma^{2}), ylab = "Density",
     main = expression("Prior and Posterior Distributions of"~sigma^{2}))
```

```
lines(density(sig.2.prior), lwd = 2)
legend("topright",lty=c(1,1),lwd=c(2,2),c("Prior","Posterior"),
       col=c("black","red"),inset=0.05)
dev.off()

png("tau-density.png")
plot(density(tau.2.post), col = "red", lwd = 2,
     xlab = expression(tau^{2}), ylab = "Density",
     main = expression("Prior and Posterior Distributions of"~tau^{2}))
lines(density(tau.2.prior), lwd = 2)
legend("topright",lty=c(1,1),lwd=c(2,2),c("Prior","Posterior"),
       col=c("black","red"),inset=0.05)
dev.off()


##### Part (c) #####
## Posterior Density R ##
R.prior = tau.2.prior/(tau.2.prior + sig.2.prior)
R.post = tau.2.post/(tau.2.post + sig.2.post)

png("rplot.png")
plot(density(R.post), type = "l", col = "red", lwd = 2,
     xlab = "R", ylab = "Density",
     main = "Prior and Posterior Distributions of R")
lines(density(R.prior), lwd = 2)
legend("topright",lty=c(1,1),lwd=c(2,2),c("Prior","Posterior"),
       col=c("black","red"),inset=0.05)
dev.off()


##### Part (d) #####
print(mean(THETA[7,] < THETA[6,]))
print(mean(THETA[7,] < THETA[1,] & THETA[7,] < THETA[2,]
           & THETA[7,] < THETA[3,] & THETA[7,] < THETA[4,]
           & THETA[7,] < THETA[5,] & THETA[7,] < THETA[6,]
           & THETA[7,] < THETA[8,]))

##### Part (e) #####
## Plot samp avg vs posterior expectation ##
theta.postmean = rep(NA,m)
for(i in 1:m){
  theta.postmean[i] = mean(THETA[i,])
}

png("groupmean.png")
plot(theta.postmean, ybar, pch = 20,
     xlab=expression(hat(theta)), ylab=expression(bar(y)),
     main="Sample Averages and Posterior Expectations")
lines(6:11,6:11)
dev.off()

## All observation mean ##
print(mu.postmean)
samp.avg = sum(ybar*n)/sum(n)
```