HW07

Zach White

10/28/2016

Exercise 7.3

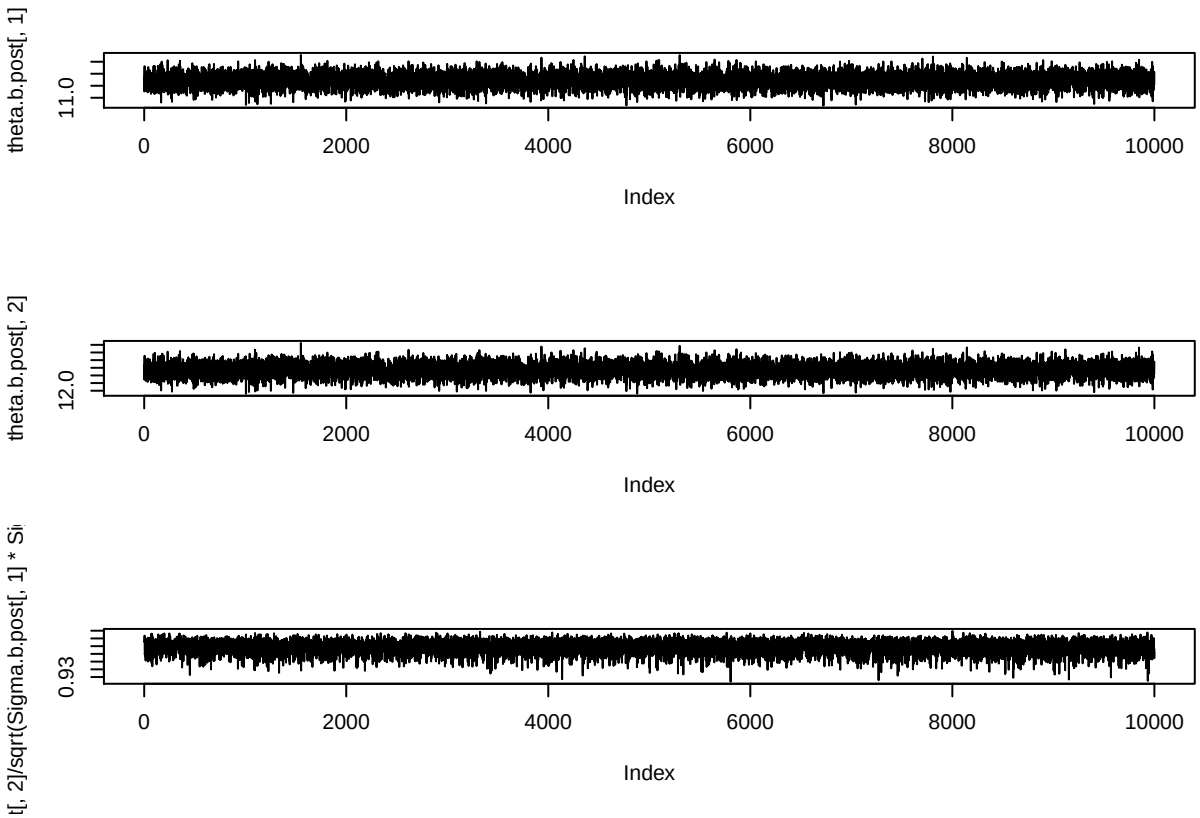
Part A

```
blue = read.table("/home/grad/zmw5/Fall 2016/STA 601/sta601/HW/bluecrab.dat")
orange = read.table("/home/grad/zmw5/Fall 2016/STA 601/sta601/HW/orangecrab.dat")
L.b.0 = S.b.0 = cov(blue)
L.o.0 = S.o.0 = cov(orange)
mu.b.0 = ybar.b = apply(blue,2,mean)
mu.o.0 = ybar.o = apply(orange,2,mean)

L.b.0.inv = S.b.0.inv = solve(L.b.0)
L.o.0.inv = S.o.0.inv = solve(L.o.0)

ybar.b = apply(blue,2,mean)
ybar.o = apply(orange,2,mean)
Sigma.b = cov(blue)
Sigma.o = cov(orange)
Sigma.b.inv = solve(Sigma.b)
Sigma.o.inv = solve(Sigma.o)

n.b = nrow(blue)
n.o = nrow(orange)
nu.0 = 4

n.iter = 10000

theta.b.post = theta.o.post = NULL
Sigma.b.post = Sigma.o.post = NULL




for(i in 1:n.iter){
  ### Update theta.b, theta.o
  Ln.b = solve(S.b.0.inv + n.b* Sigma.b.inv)
  Ln.o = solve(S.o.0.inv + n.o * Sigma.o.inv)
  mu.n.b = Ln.b %*% (L.b.0.inv %*% mu.b.0 + n.b*Sigma.b.inv %*% ybar.b)
  mu.n.o = Ln.o %*% (L.o.0.inv %*% mu.o.0 + n.o*Sigma.o.inv %*% ybar.o)
  theta.b = mvrnorm(1,mu.n.b,Ln.b)
  theta.o = mvrnorm(1,mu.n.o,Ln.o)

  theta.b.post = rbind(theta.b.post,theta.b)
  theta.o.post = rbind(theta.o.post,theta.o)
```

```
  ### Update Sigma.b, Sigma.o
  Sn.b = S.b.0 + (t(blue) - c(theta.b)) %*% t(t(blue) - c(theta.b))
  Sn.o = S.o.0 + (t(orange) - c(theta.o)) %*% t(t(orange) - c(theta.o))
  Sigma.b = solve(rWishart(1,nu.0 + n.b, solve(Sn.b))[,,1])
  Sigma.o = solve(rWishart(1,nu.0 + n.o, solve(Sn.o))[,,1])
  Sigma.b.post = rbind(Sigma.b.post,c(Sigma.b))
  Sigma.o.post = rbind(Sigma.o.post,c(Sigma.o))
}
```

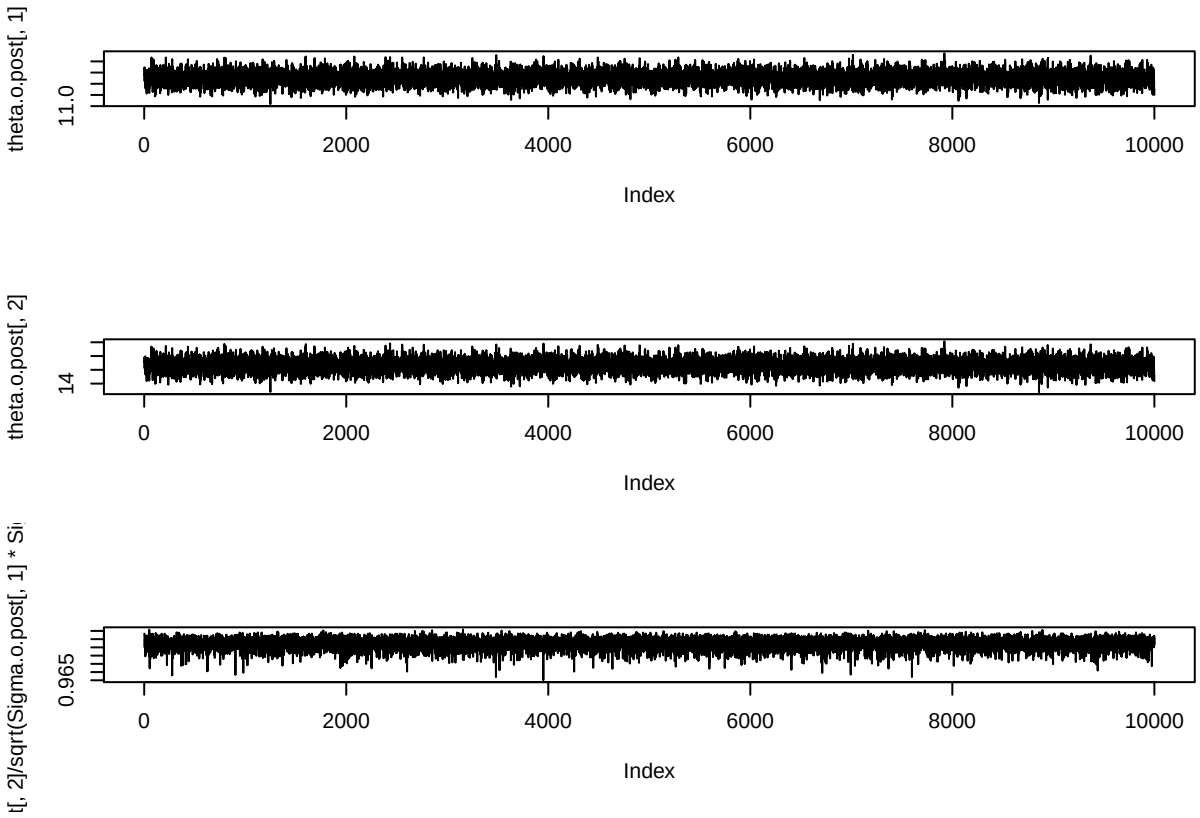
Part B

```
theta.b.post = as.data.frame(theta.b.post)
names(theta.b.post) = c("theta1","theta2")

theta.o.post = as.data.frame(theta.o.post)
names(theta.o.post) = c("theta1","theta2")

par(mfrow = c(3,1))
plot(theta.b.post[,1], type = "l")
plot(theta.b.post[,2], type = "l")
plot(Sigma.b.post[,2] / sqrt(Sigma.b.post[,1]*Sigma.b.post[,4]), type = "l")
```




```
par(mfrow = c(3,1))
plot(theta.o.post[,1], type = "l")
```

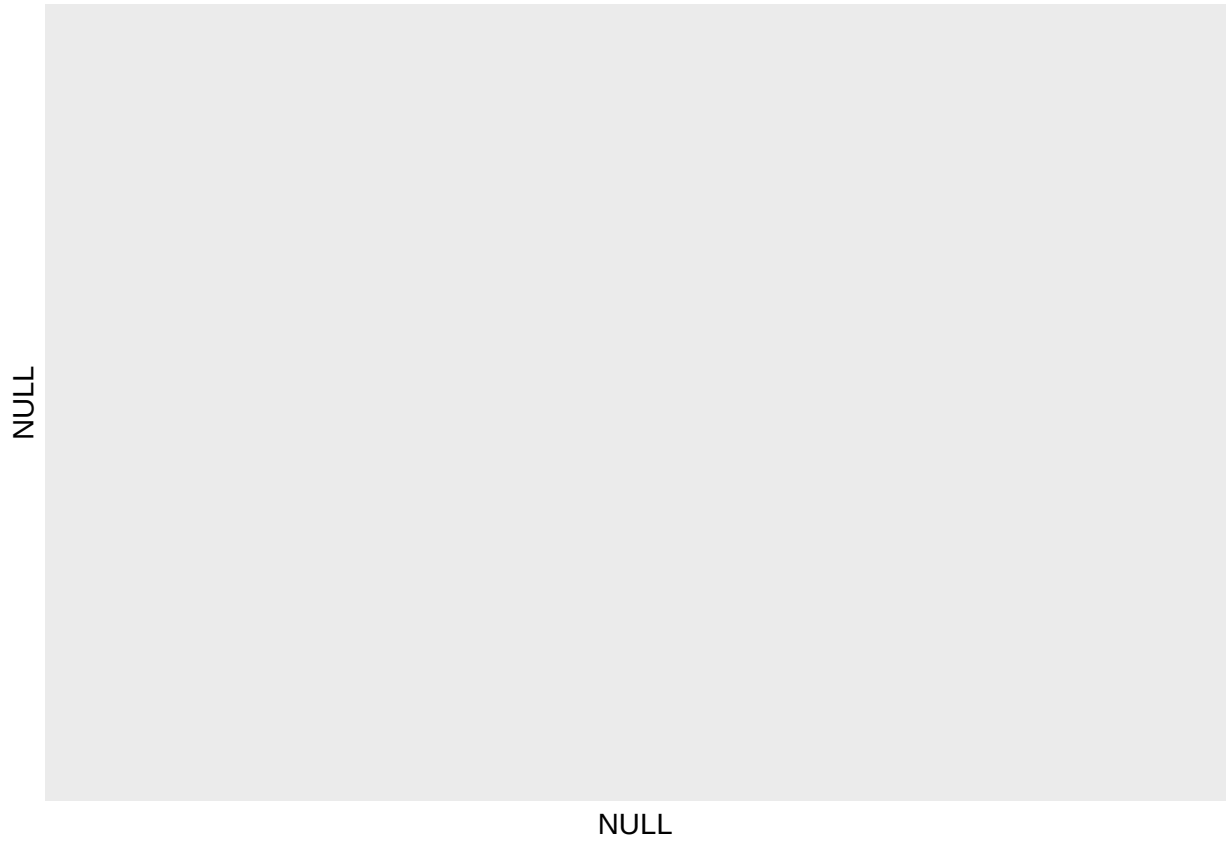
```r
plot(theta.o.post[,2], type = "l")
plot(Sigma.o.post[,2] / sqrt(Sigma.o.post[,1]*Sigma.o.post[,4]), type = "l")
```



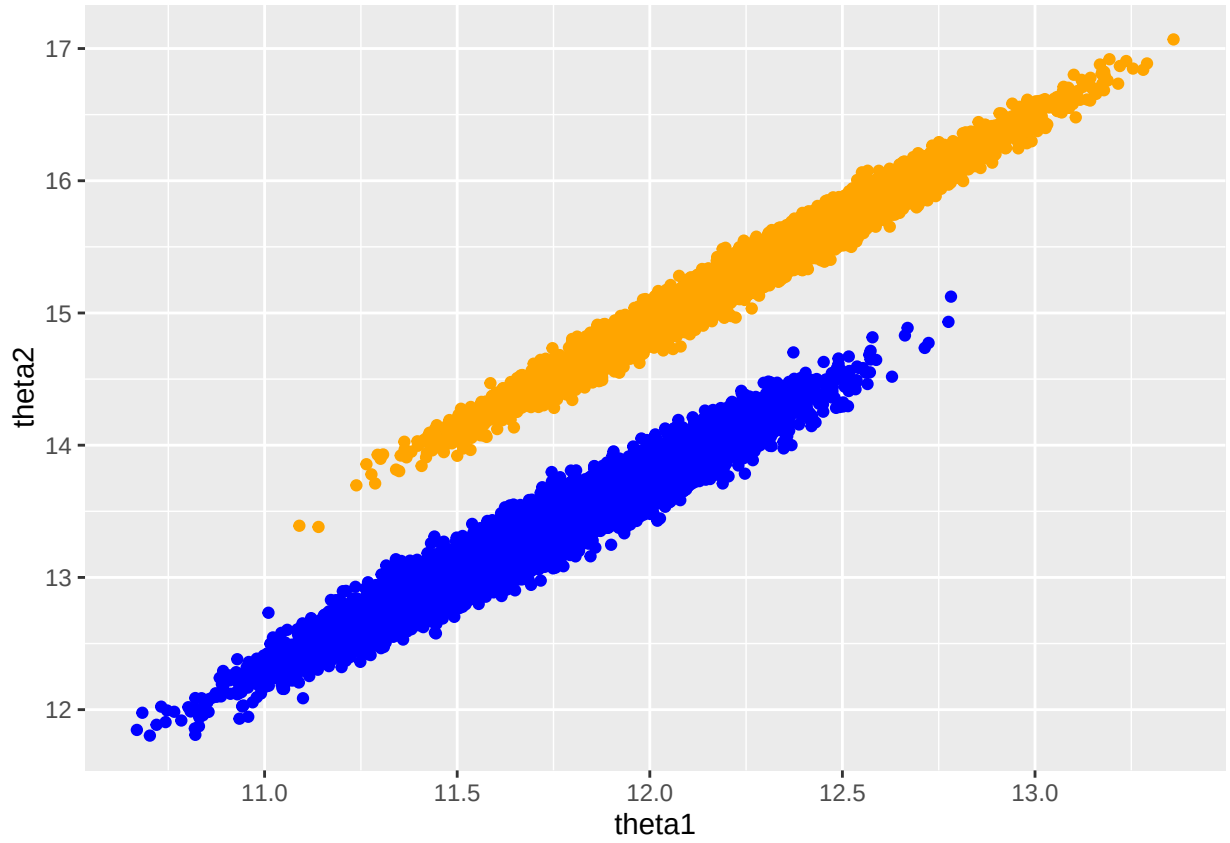

```r
both.theta.post = cbind(theta.b.post, theta.o.post)
```

```
## Warning in data.row.names(row.names, rowsi, i): some row.names duplicated:
## 2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,3

## Warning in data.row.names(row.names, rowsi, i): some row.names duplicated:
## 2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,3
```

```r
id = rownames(both.theta.post)
both.theta.post = cbind(id,both.theta.post)
names(both.theta.post) = c("id","b.theta1","b.theta2","o.theta1","o.theta2")
both.theta.post = as.data.frame(both.theta.post)

par(mfrow = c(1,1))
ggplot(data = both.theta.post,aes(x = NULL,y = NULL))
```

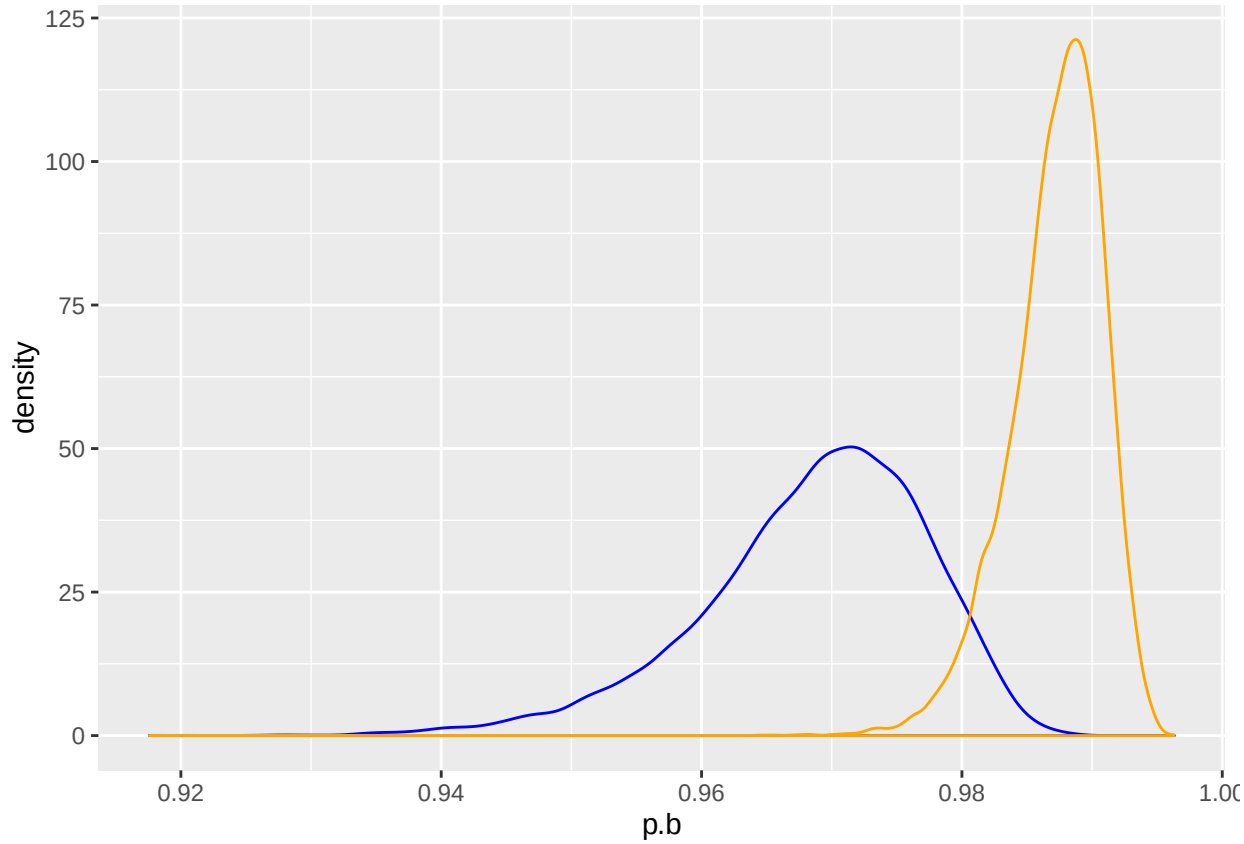



```
ggplot(data = theta.b.post, aes(x = theta1, y = theta2)) + geom_point(color = "blue") + geom_point(data
```



Part C

```
p.b = Sigma.b.post[,2] / sqrt(Sigma.b.post[,1] * Sigma.b.post[,4])
p.o = Sigma.o.post[,2] / sqrt(Sigma.o.post[,1]*Sigma.o.post[,4])

ggplot(data = as.data.frame(p.b) , aes(x = p.b)) + geom_density(color = "blue") + geom_density(data = as
```




```
mean(p.b < p.o)
```

```
## [1] 0.9902
```

Exercise 7.5

Part A

```
inter.exp = read.table("/home/grad/zmw5/Fall 2016/STA 601/sta601/HW/interexp.dat", header = TRUE)
apply(inter.exp,2,mean)
```

```
## yA yB
## NA NA
```

```
theta = sapply(inter.exp,mean,na.rm = TRUE)
thetaA = theta[1]
thetaB = theta[2]
## Calculate Sigma
sapply(inter.exp,var)
```

```
## yA yB
## NA NA
```

```
sigma2 = sapply(inter.exp,var,na.rm = TRUE)
sigma2.A = sigma2[1]
```

```
sigma2.B = sigma2[2]
## Calculate correlation matrix
complete.rho = cor(inter.exp, use = "complete")
rho = complete.rho[1,2]
```

Part B

```
A.missingB = inter.exp[is.na(inter.exp[,2]),1]
B.missingA = inter.exp[is.na(inter.exp[,1]),2]

impute.B = thetaB + (A.missingB - thetaA)* rho *sqrt(sigma2.B / sigma2.A)
impute.A = thetaA + (B.missingA - thetaB) * rho * sqrt(sigma2.A / sigma2.B)

imp.data = inter.exp
imp.data[is.na(imp.data[,2]),2] = impute.B
imp.data[is.na(imp.data[,1]),1] = impute.A

t.results = t.test(imp.data[,1],imp.data[,2], paired = TRUE)
t.results$conf.int
```

```
## [1] -0.9850730 -0.2383347
## attr(,"conf.level")
## [1] 0.95
```

Part C

I will use the unit information prior

```
ybar = apply(inter.exp,2,mean,na.rm = TRUE)
complete = which(complete.cases(inter.exp))
## Prior on Sigma
S = (t(inter.exp[complete,]) - ybar) %*% t(t(inter.exp[complete,]) - ybar)/length(complete)
nu.0 = nrow(S) + 2
n = nrow(inter.exp)

n.iter = 10000

y.A.samps = y.B.samps = matrix(0, nrow = n.iter,ncol = n)
theta.post = matrix(0,nrow=n.iter, ncol = 2)
names(theta.post) = c("thetaA","thetaB")

#STarting values
Y = imp.data
Sigma = S
theta = ybar

miss.A = which(is.na(inter.exp$yA))
miss.B = which(is.na(inter.exp$yA))

for(i in 1:n.iter){
  # Update theta
  y.bar.samp = apply(Y,2,mean)
```

```
  theta = mvrnorm(1,y.bar.samp,Sigma / (n+1))
  theta.post[i, ] <- theta

  # Update Sigma
  Sn<- S + ( t(Y)-c(theta) )%*%t( t(Y)-c(theta) )
  Sigma<-solve(rWishart(1, nu.0+n, solve(Sn))[, , 1])
  sigma2.A <- Sigma[1, 1]
  sigma2.B <- Sigma[2, 2]
  rho <- Sigma[1,2] / sqrt(sigma2.A*sigma2.B)

  Y[miss.A,1] = rnorm(length(miss.A),(rho*sqrt(sigma2.A/sigma2.B))*(Y[miss.A,"yB"] - theta[2]),sqrt(sig
  Y[miss.B,2] = rnorm(length(miss.B),(rho*sqrt(sigma2.B/sigma2.A))*(Y[miss.B,"yA"] - theta[1]), sqrt(si

  y.A.samps[i, ] <- Y[, 1]
  y.B.samps[i, ] <- Y[, 2]
}

theta.diff = theta.post[,1] - theta.post[,2]
cred.int = quantile(theta.diff, c(.025,.975))
cred.int
```

```
##      2.5%     97.5%
## -3.603018  3.864774
```

The credible interval of interest is -3.6030181, 3.8647738, which means that there is a .95 probability that $\theta_A - \theta_B|y$ lies in th interval -3.6030181, 3.8647738. # Exercise 8.2

Part A

```
n.a = n.b = 16
y.bar.a = 75.2
s.a = 7.3
y.bar.b = 77.5
s.b = 8.1

delta0 = seq(-4,4,by = 2)
tau2.0 = c(10,50,100,500)
```