HW07

Zach White

10/28/2016

Exercise 7.3

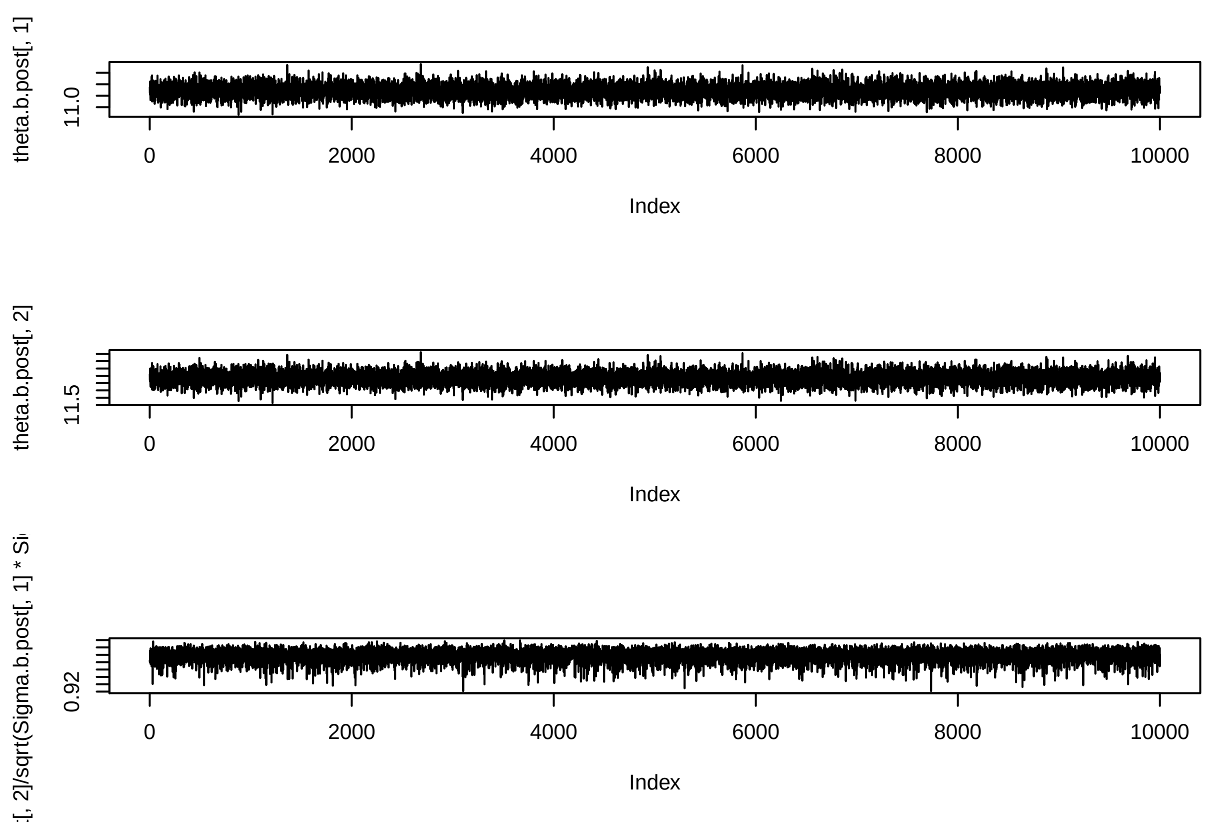
Part A

```
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))
}
```

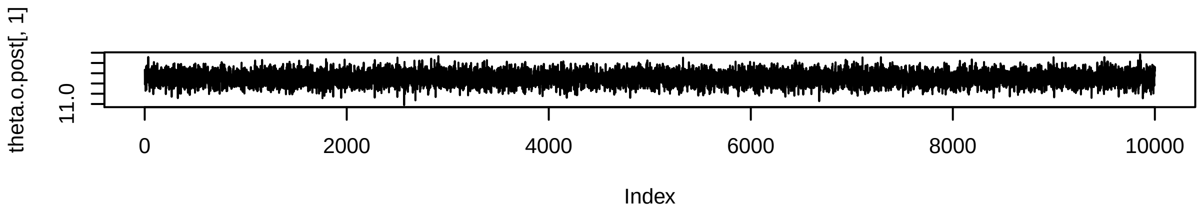
Part B

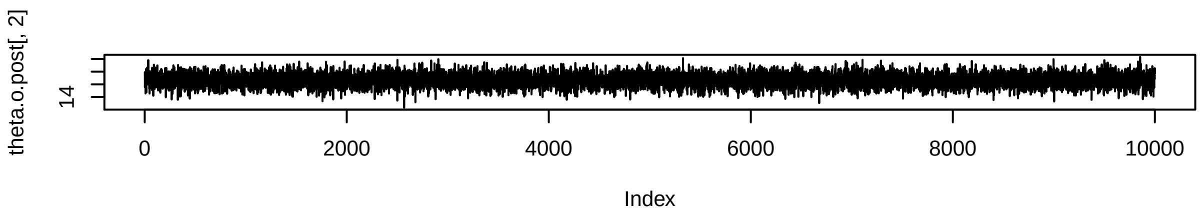
```
theta.b.post = as.data.frame(theta.b.post)
names(theta.b.post) = c("theta1","theta2")

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")
```

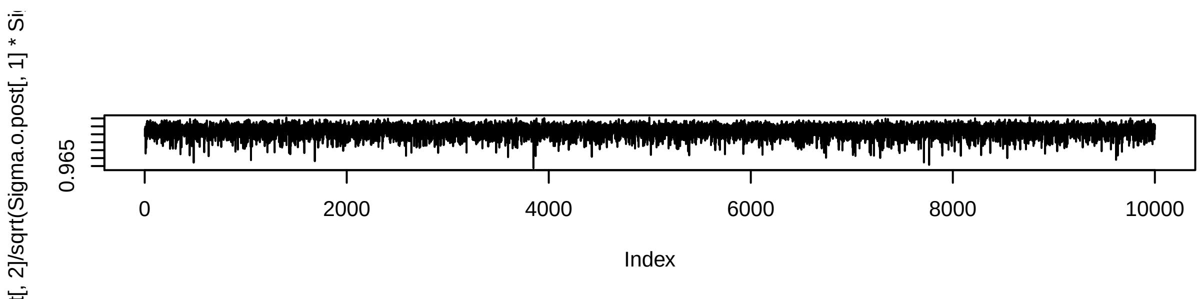



```
par(mfrow = c(3,1))
plot(theta.o.post[,1], type = "l")
```

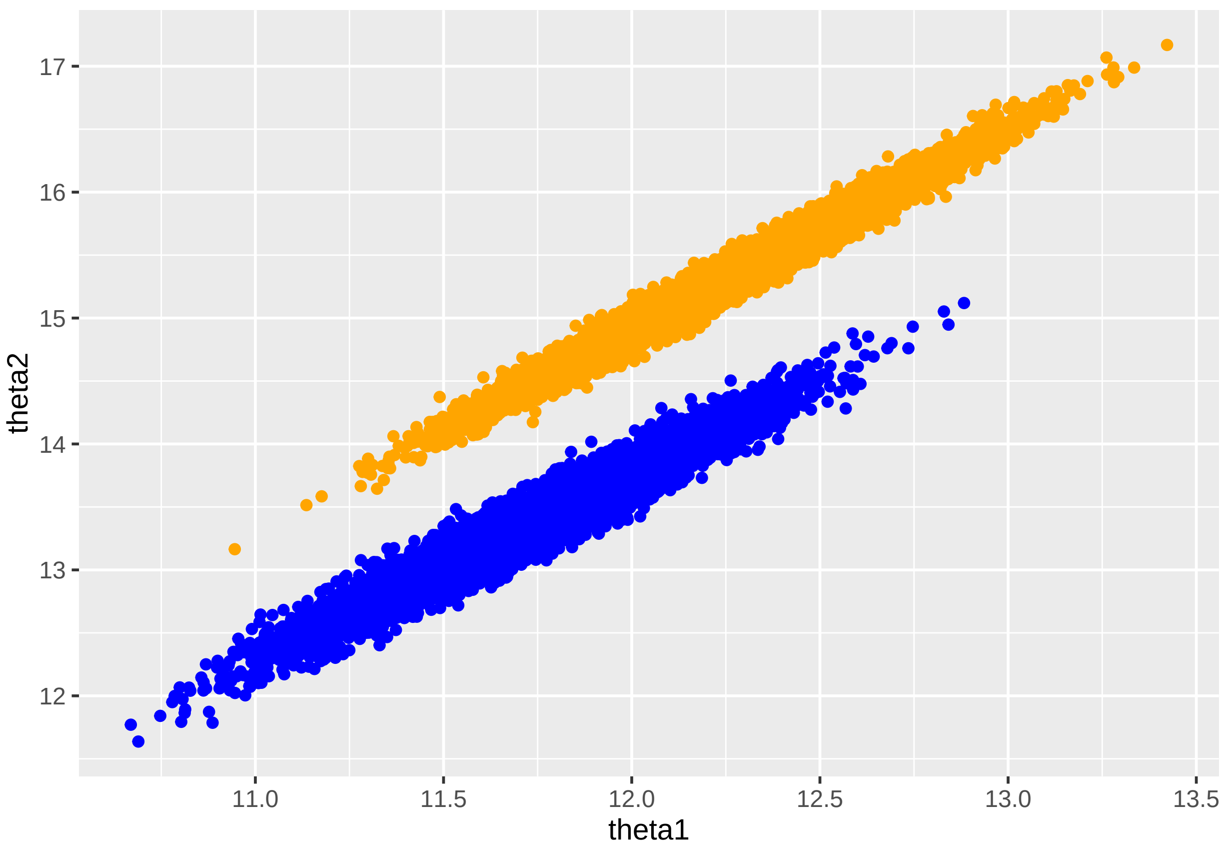
```r
plot(theta.o.post[,2], type = "l")
plot(Sigma.o.post[,2] / sqrt(Sigma.o.post[,1]*Sigma.o.post[,4]), type = "l")
```





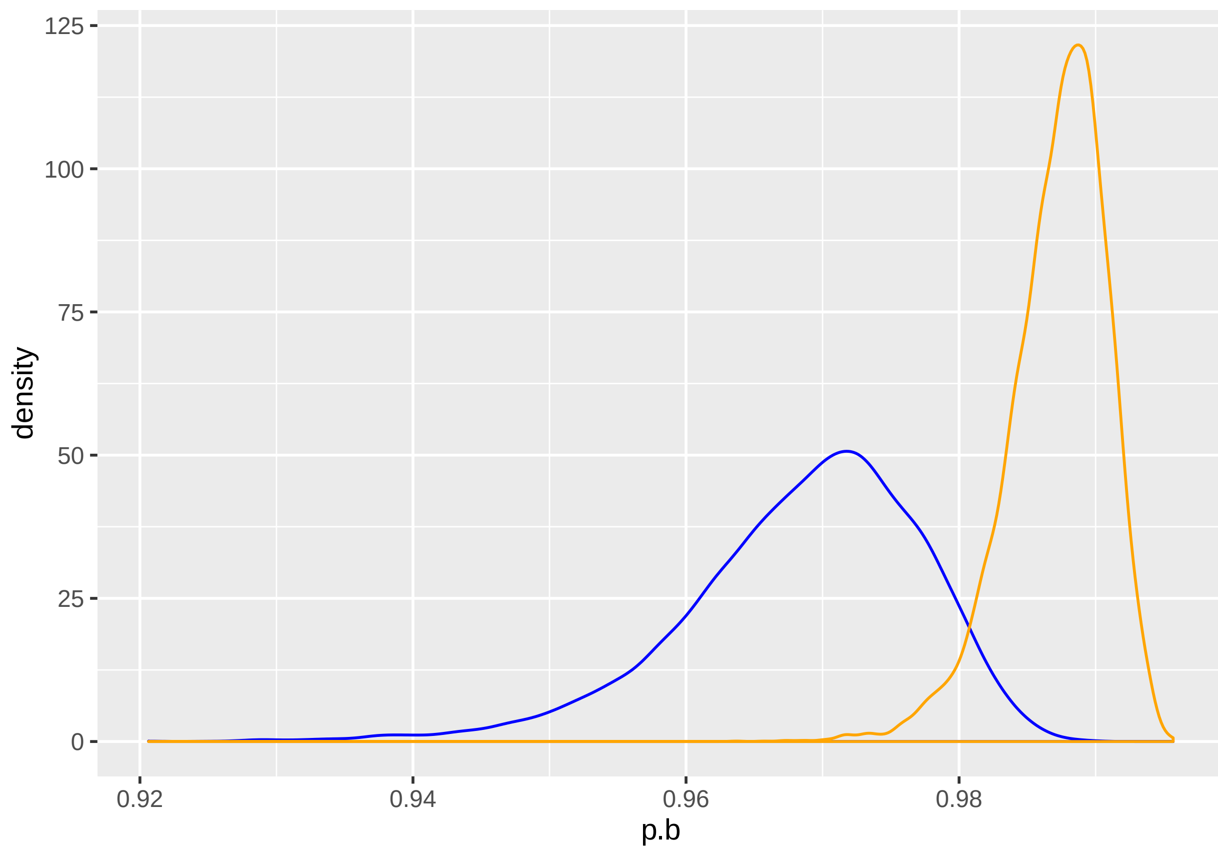



```r
both.theta.post = cbind(theta.b.post, theta.o.post)
id = rownames(both.theta.post)
both.theta.post = cbind(id,both.theta.post)
names(both.theta.post) = c("id","b.theta1","b.theta2","o.theta1","o.theta2")
both.theta.post = as.data.frame(both.theta.post)

par(mfrow = c(1,1))
ggplot(data = 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] 0.9894
```

We can be pretty sure that $\rho_b < \rho_0$, since 0.9894 is pretty high.

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)
theta
```

```
##       yA       yB
## 24.20049 24.80535
```

```
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]
sigma2
```

```
##       yA       yB
## 4.092800 4.691578
```

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

```
## [1] 0.6164509
```

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")
```

```r
#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)

  for (j in miss.A) {
    Y[j, 1] <- rnorm(1,
                     theta[1] + (rho*sqrt(sigma2.A/sigma2.B))*(Y[j, "yB"] - theta[2]),
                     sqrt(sigma2.A*(1 - rho)))
  }
  for (j in miss.B) {
    Y[j, 2] <- rnorm(1,
                     theta[2] + (rho*sqrt(sigma2.B/sigma2.A))*(Y[j, "yA"] - theta[1]),
                     sqrt(sigma2.B*(1 - rho)))
  }


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

  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%
## -1.0436781 -0.1749426
```

The credible interval of interest is -1.0436781, -0.1749426, which means that there is a .95 probability that $\theta_A - \theta_B|y$ lies in th interval -1.0436781, -0.1749426. When comparing this result with the results from the t test, ours are a little bit wider. Our lower extreme is lower than the one from the traditional t test. However, we now have significantly more information with our analysis because we have posteriors on $\Sigma, \theta_A, \theta_B$ and finally we actually have distributions on the imputed values.

Exercise 8.2

$$\mu \sim N(\mu_0, \gamma_o^2) = N(75, 100)$$
$$\frac{1}{\sigma^2} \sim Gamma(\eta_0, \tau_0^2) = Gamma(1, 100)$$
$$\delta \sim N(\delta_0, \tau_0^2)$$

We perform this analysis with varying levels of δ_0 and τ_0^2. The levels are as follows $\delta_0 \in \{-4, -2, 0, 2, 4\}$ and $\tau_0^2 \in \{10, 50, 100, 500\}$

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
sum.y.a = n.a * y.bar.a
sum.y.b = n.b * y.bar.b

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




# Hyperparams
mu.0 = 75
gamma2.0 = 100
nu.0 = 1
sigma2.0 = 100


# Starting values
mu = (y.bar.a + y.bar.b )/2
delta = (y.bar.a - y.bar.b) / 2

# Form some arays to store values

mu.array =sigma2.array = delta.array =  array(0,c(length(delta0),length(tau2.0),n.iter))


## Gibbs Sampler
n.iter = 10000
for(i in 1:n.iter){
  for(j in 1:length(delta0)){
    for(k in 1:length(tau2.0)){

      # Update Sigma2
      nu.sigma = (nu.0*sigma2.0 + (n.a -1)*s.a^2 + n.a*(y.bar.a - (mu + delta))^2 + (n.b-1)*s.b^2 + n.b
      sigma2 = 1 / rgamma(1,(nu.0 + n.a + n.b)/2,nu.sigma)
      sigma2.array[j,k,i] = sigma2
```

```
      ## Update mu
      var.mu = 1/(1/gamma2.0 + (n.a + n.b)/sigma2)
      mean.mu = var.mu * (mu.0 / gamma2.0 +  ((sum.y.a - n.a*mu)/sigma2) + ((sum.y.b + n.b*mu)/sigma2))
      mu = rnorm(1,mean.mu,sqrt(var.mu))
      mu.array[j,k,i] = mu

      ## Update delta
      var.delta = 1/(1/tau2.0[k] + (n.a + n.b)/sigma2)
      mean.delta = var.delta*(delta0[j]/tau2.0[k] + ((sum.y.a - n.a*mu)/sigma2) - (sum.y.b - n.b*mu)/sig
      delta = rnorm(1,mean.delta,sqrt(var.delta))
      delta.array[j,k,i] = delta
    }
  }
}

## Post Analysis
#sapply(delta.array,c(1,2), mean < 0)
prob.0 = lower.bound = upper.bound = post.cor = prior.cor= matrix(0,5,4)
delta.n = length(delta0)
tau.n = length(tau2.0)
theta.A = mu.array + delta.array
theta.B = mu.array - delta.array
for(i in 1:delta.n){
  for(j in 1:tau.n){
    prob.0[i,j] = mean(delta.array[i,j,] < 0)
    lower.bound[i,j] = quantile(delta.array[i,j,],.025)
    upper.bound[i,j] = quantile(delta.array[i,j,],.975)
    post.cor[i,j] = cor(theta.A[i,j,],theta.B[i,j,])
    prior.cor[i,j]= (gamma2.0 - tau2.0[j]) / (gamma2.0 + tau2.0[j])
  }
}
prob.0
```

```
##        [,1]   [,2]   [,3]   [,4]
## [1,] 0.8763 0.7629 0.7386 0.7229
## [2,] 0.7939 0.7321 0.7263 0.7153
## [3,] 0.6832 0.7144 0.7125 0.7123
## [4,] 0.5496 0.6858 0.6915 0.7059
## [5,] 0.4174 0.6461 0.6832 0.7152
```

```
apply(delta.array,c(1,2),quantile,c(.025,.975))
```

```
## , , 1
##
##            [,1]      [,2]      [,3]      [,4]      [,5]
## 2.5%  -5.450320 -4.837394 -4.213454 -3.513136 -2.923450
## 97.5%  1.342516  1.986148  2.598117  3.232731  3.833153
##
## , , 2
##
##            [,1]      [,2]      [,3]      [,4]      [,5]
## 2.5%  -5.300503 -5.130024 -4.889195 -4.769708 -4.625827
## 97.5%  2.444761  2.689346  2.853988  3.013621  3.255961
##
```

```
## , , 3
##
##              [,1]      [,2]      [,3]      [,4]     [,5]
## 2.5%  -5.363094 -5.267547 -5.030863 -5.046394 -4.97473
## 97.5%  2.892480  2.791190  2.873086  3.060677  3.08940
##
## , , 4
##
##              [,1]      [,2]      [,3]      [,4]      [,5]
## 2.5%  -5.234266 -5.201923 -5.311064 -5.165292 -5.222817
## 97.5%  2.845649  2.895587  2.847964  2.952792  2.915023
```

```
apply(delta.array,c(1,2),quantile,.025)
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,] -5.450320 -5.300503 -5.363094 -5.234266
## [2,] -4.837394 -5.130024 -5.267547 -5.201923
## [3,] -4.213454 -4.889195 -5.030863 -5.311064
## [4,] -3.513136 -4.769708 -5.046394 -5.165292
## [5,] -2.923450 -4.625827 -4.974730 -5.222817
```

```
apply(delta.array,c(1,2),quantile,.975)
```

```
##          [,1]     [,2]     [,3]     [,4]
## [1,] 1.342516 2.444761 2.892480 2.845649
## [2,] 1.986148 2.689346 2.791190 2.895587
## [3,] 2.598117 2.853988 2.873086 2.847964
## [4,] 3.232731 3.013621 3.060677 2.952792
## [5,] 3.833153 3.255961 3.089400 2.915023
```

This matrices above shows the lower and upper bounds for credible intervals on δ for the varying combinations of the prior specifications for δ_0 and τ_0^2.

```
lower.bound
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,] -5.450320 -5.300503 -5.363094 -5.234266
## [2,] -4.837394 -5.130024 -5.267547 -5.201923
## [3,] -4.213454 -4.889195 -5.030863 -5.311064
## [4,] -3.513136 -4.769708 -5.046394 -5.165292
## [5,] -2.923450 -4.625827 -4.974730 -5.222817
```

```
upper.bound
```

```
##          [,1]     [,2]     [,3]     [,4]
## [1,] 1.342516 2.444761 2.892480 2.845649
## [2,] 1.986148 2.689346 2.791190 2.895587
## [3,] 2.598117 2.853988 2.873086 2.847964
## [4,] 3.232731 3.013621 3.060677 2.952792
## [5,] 3.833153 3.255961 3.089400 2.915023
```

The two matrices above shows the posterior credible intervals on δ for each of the combinations of the prior values.

```
prior.cor
```

```
##           [,1]      [,2] [,3]       [,4]
## [1,] 0.8181818 0.3333333    0 -0.6666667
## [2,] 0.8181818 0.3333333    0 -0.6666667
```

```
## [3,] 0.8181818 0.3333333    0 -0.6666667
## [4,] 0.8181818 0.3333333    0 -0.6666667
## [5,] 0.8181818 0.3333333    0 -0.6666667
```

```
post.cor
```

```
##           [,1]       [,2]         [,3]         [,4]
## [1,] 0.1768536 0.02609228 -0.015277319  0.003050304
## [2,] 0.1485950 0.02910766  0.004146842 -0.005350301
## [3,] 0.1686626 0.01661998  0.008281233 -0.032555456
## [4,] 0.1742838 0.03430610 -0.020647034 -0.025602462
## [5,] 0.1624333 0.02090742 -0.002131438 -0.008760478
```

The above shows the prior correlations along with the posterior correlations.

Part B

It's clear that for the most part that for each prior choice, we have a $p(\delta|y) < 0$ is greater than .5. However, we can see that when we have a highly positive δ_0 with a low τ_0^2, then the probability is greatly diminished. However, for most priors, there is a good chance that $p(\delta|y) < 0$, and so we can speak and say that unless the prior is strong and negative, then we can be pretty sure that is the case. Thus even, if they are very certain that $\delta > 0$, then they should consider the evidence that this analysis brings because the only one where this probability is less than .5, is a strong positive δ_0 with a very low prior uncertainty or low τ_0^2