

STAT 638 HW 7

Jack Cunningham

7.1)

a)

The most obvious reason that p_j cannot actually be a probability density for (θ, Σ) is that it does not depend on θ .

Since p_j does not depend on θ , it assigns equal probability mass across the interval $(-\infty, \infty)$. So if we were to integrate over the range of all possible θ the probabilities would sum to ∞ and not 1, which clearly violates a fundamental rule for probability density functions, total probability must sum to 1.

b)

We have:

$$p_j(\theta, \Sigma | y_1, \dots, y_n) \propto p_j(\theta, \Sigma) \times p(y_1, \dots, y_n | \theta, \Sigma)$$

With our assumption that $p(y_1, \dots, y_n | \theta, \Sigma) \sim \text{Multivariate Normal}(\theta, \Sigma)$ we have:

$$p(y_1, \dots, y_n | \theta, \Sigma) \propto |\Sigma|^{-n/2} \exp\left\{-\frac{1}{2} \sum_{i=1}^n (y_i - \theta)^T \Sigma^{-1} (y_i - \theta)\right\}$$

So:

$$p_j(\theta, \Sigma | y_1, \dots, y_n) \propto |\Sigma|^{-(n+p+2)/2} \exp\left\{-\frac{1}{2} \sum_{i=1}^n (y_i - \theta)^T \Sigma^{-1} (y_i - \theta)\right\}$$

For $p_J(\theta | \Sigma, y_1, \dots, y_n)$ we have:

$$p_J(\theta | \Sigma, y_1, \dots, y_n) \propto \exp\left\{-\frac{1}{2} \sum_{i=1}^n (y_i - \theta)^T \Sigma^{-1} (y_i - \theta)\right\}$$

This is clearly the Multivariate Normal($\bar{y}, \frac{1}{n}\Sigma$) distribution, any effect from the prior is ignored because it does not depend on θ .

For $p_j(\Sigma|y_1, \dots, y_n)$ its worth noticing that the prior $p_j(\Sigma)$ is inverse - Wishart(1, 0). Then we use the general result that was worked through in chapter 7:

$$p_j(\Sigma|y_1, \dots, y_n) \sim \text{inverse-Wishart}(\nu_0 + n, [S_0 + S_\theta]^{-1}) = \text{inverse-Wishart}(1 + n, S_\theta^{-1})$$

Where $S_\theta = \sum_{i=1}^n (y_i - \theta)(y_i - \theta)^T$.

7.3)

Loading data:

```
blue_crab = read.table("bluecrab.dat")
blue_crab = data.frame(depth = blue_crab$V1, width = blue_crab$V2)

orange_crab = read.table("orangecrab.dat")
orange_crab = data.frame(depth = orange_crab$V1, width = orange_crab$V2)

head(blue_crab);head(orange_crab)
```

	depth	width
1	6.7	7.0
2	7.7	7.4
3	7.8	7.7
4	7.9	8.2
5	8.0	8.2
6	9.0	9.8

	depth	width
1	6.9	7.4
2	8.2	9.0
3	8.6	9.2
4	9.0	10.1
5	9.4	10.8
6	9.4	11.2

Obtaining Posterior Distribution:

```

library(mvtnorm)
#priors
mu_blue    = colMeans(blue_crab); mu_orange = colMeans(orange_crab)
lambda_blue_0 = S_blue_0 = cov(blue_crab)
lambda_orange_0 = S_orange_0 = cov(orange_crab)
nu_0 = 4
n     = 50
theta_blue <- theta_orange <- matrix(0, nrow = 10000, ncol = 2)

sigma_blue <- theta_orange <- sigma_orange <- matrix(nrow = 10000, ncol = 4)

set.seed(10)
for(s in 1:10000){
  #theta
  Ln_blue <- solve(solve(lambda_blue_0) + n*solve(S_blue_0))
  mun_blue <- Ln_blue %*% (solve(lambda_blue_0)%*%mu_blue +
                          n*solve(S_blue_0)%*% mu_blue)
  theta_blue[s,] <- rmvnorm(1, mun_blue, Ln_blue)

  Ln_orange <- solve(solve(lambda_orange_0) + n*solve(S_orange_0))
  mun_orange <- Ln_orange %*% (solve(lambda_orange_0)%*%mu_orange +
                              n*solve(S_orange_0) %*% mu_orange)
  theta_orange[s,] <- rmvnorm(1, mun_orange, Ln_orange)

  #Sigma
  Sn_blue <- S_blue_0 +
    (t(blue_crab) - theta_blue[s])%*% t(t(blue_crab) - theta_blue[s])
  sigma_blue[s,] <- c(Sn_blue)

  Sn_orange <- S_orange_0 +
    (t(orange_crab) - theta_orange[s])%*% t(t(orange_crab) - theta_orange[s])
  sigma_orange[s,] <- c(Sn_orange)
}

```

b)

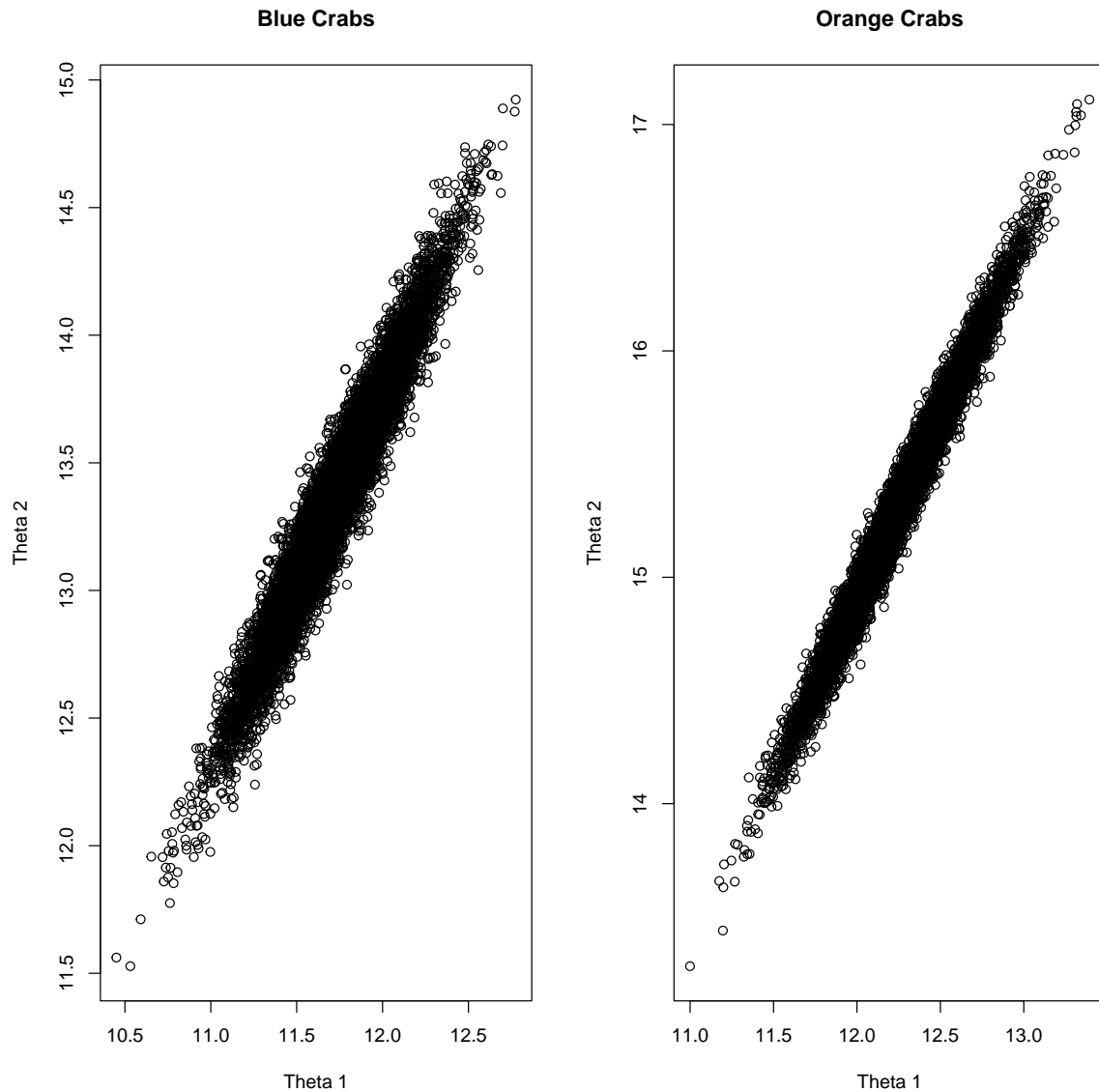
```

par(mfrow = c(1,2))

plot(theta_blue[,1], theta_blue[,2],
     main = "Blue Crabs", xlab = "Theta 1", ylab = "Theta 2")

```

```
plot(theta_orange[,1],theta_orange[,2],
     main = "Orange Crabs", xlab = "Theta 1", ylab = "Theta 2")
```



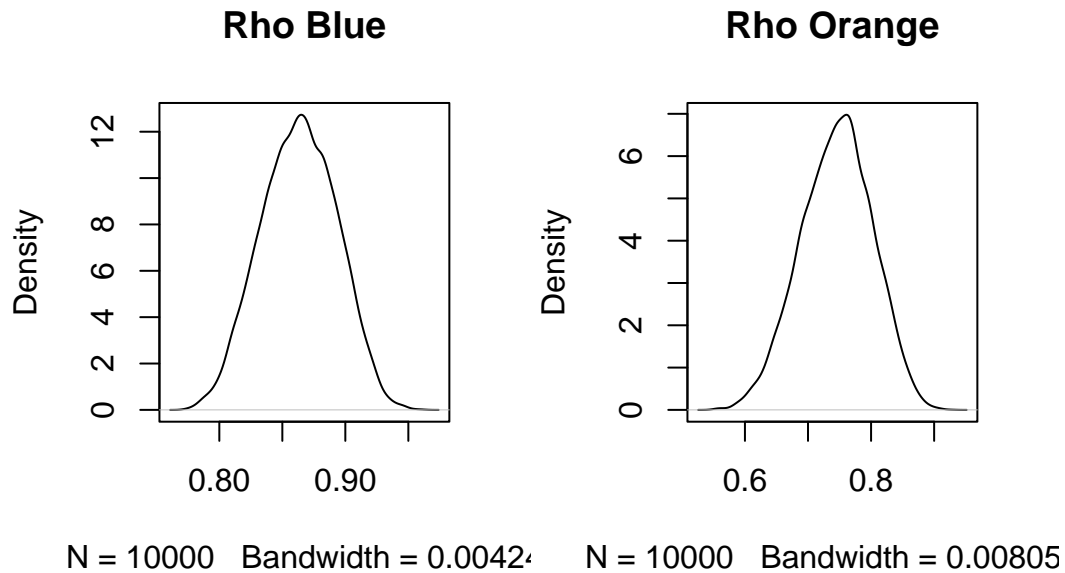
The relationship between depth and width is similar in both groups, it is tighter for orange crabs however. Both groups have similar levels of depth, but orange crabs are generally wider.

c)

```

par(mfrow = c(1,2))
rho_blue = sigma_blue[,2]/sqrt(sigma_blue[,1]*sigma_blue[,4])
rho_orange = sigma_orange[,2]/sqrt(sigma_orange[,1]*sigma_orange[,4])
plot(density(rho_blue),
     main = "Rho Blue")
plot(density(rho_orange),
     main = "Rho Orange")

```



The correlation is higher for the blue crabs as it is centered at .875 while it is centered at .775 for orange crabs. The distributions appear very similar outside of that shift.

```

p_blue_lt_orange <- mean(rho_blue < rho_orange)
p_blue_lt_orange

```

```
[1] 0.0278
```

The probability that $\rho_{\text{blue}} < \rho_{\text{orange}}$ is 0.0278. The results suggest that the correlation between depth and width is stronger in the blue crabs.

7.4)

a)

We can use:

$$\theta|y_1, \dots, y_n, \Sigma \sim \text{Multivariate Normal}(\mu_n, \Lambda_n)$$

$$\Sigma|y_1, \dots, y_n, \theta \sim \text{inverse-Wishart}(\nu_n, S_n^{-1})$$

Where $\Lambda_n = (\Lambda_0^{-1} + n\Sigma^{-1})^{-1}$, $\mu_n = (\Lambda_0^{-1} + n\Sigma^{-1})^{-1}(\Lambda_0^{-1}\mu_0 + n\Sigma^{-1}\bar{y})$. $\nu_n = \nu_0 + n$, $S_n = S_0 + S_\theta$.

I'd assume that $\mu = (32, 28)$ representing the belief that men are generally a bit older when they get married.

For Σ I am less sure. I think a feasible range for men is (20, 44) so I choose a variance of 36, a feasible range for women might be a bit narrower so like (22, 34), so I choose a variance of 9. I expect that the ages would be quite correlated, so I choose $\rho = .7$. That would leave us with $\Sigma = \begin{bmatrix} 36 & 12.6 \\ 12.6 & 9 \end{bmatrix}$.

b)

```
mu_0 <- c(32, 28)
nu_0 <- 50
sd_m <- 6
sd_w <- 3
corr <- .7
sigma_mean <- matrix(c(sd_m^2, corr*sd_m*sd_w, corr*sd_m*sd_w, sd_w^2),
                      ncol = 2)
psi <- sigma_mean*(nu_0 - 2 - 1)

set.seed(12)
n <- 100
num_datasets <- 4
library(MASS)
library(MCMCpack)
```

Warning: package 'MCMCpack' was built under R version 4.5.2

Loading required package: coda

Warning: package 'coda' was built under R version 4.5.2

##

Markov Chain Monte Carlo Package (MCMCpack)

Copyright (C) 2003-2025 Andrew D. Martin, Kevin M. Quinn, and Jong Hee Park

##

Support provided by the U.S. National Science Foundation

(Grants SES-0350646 and SES-0350613)

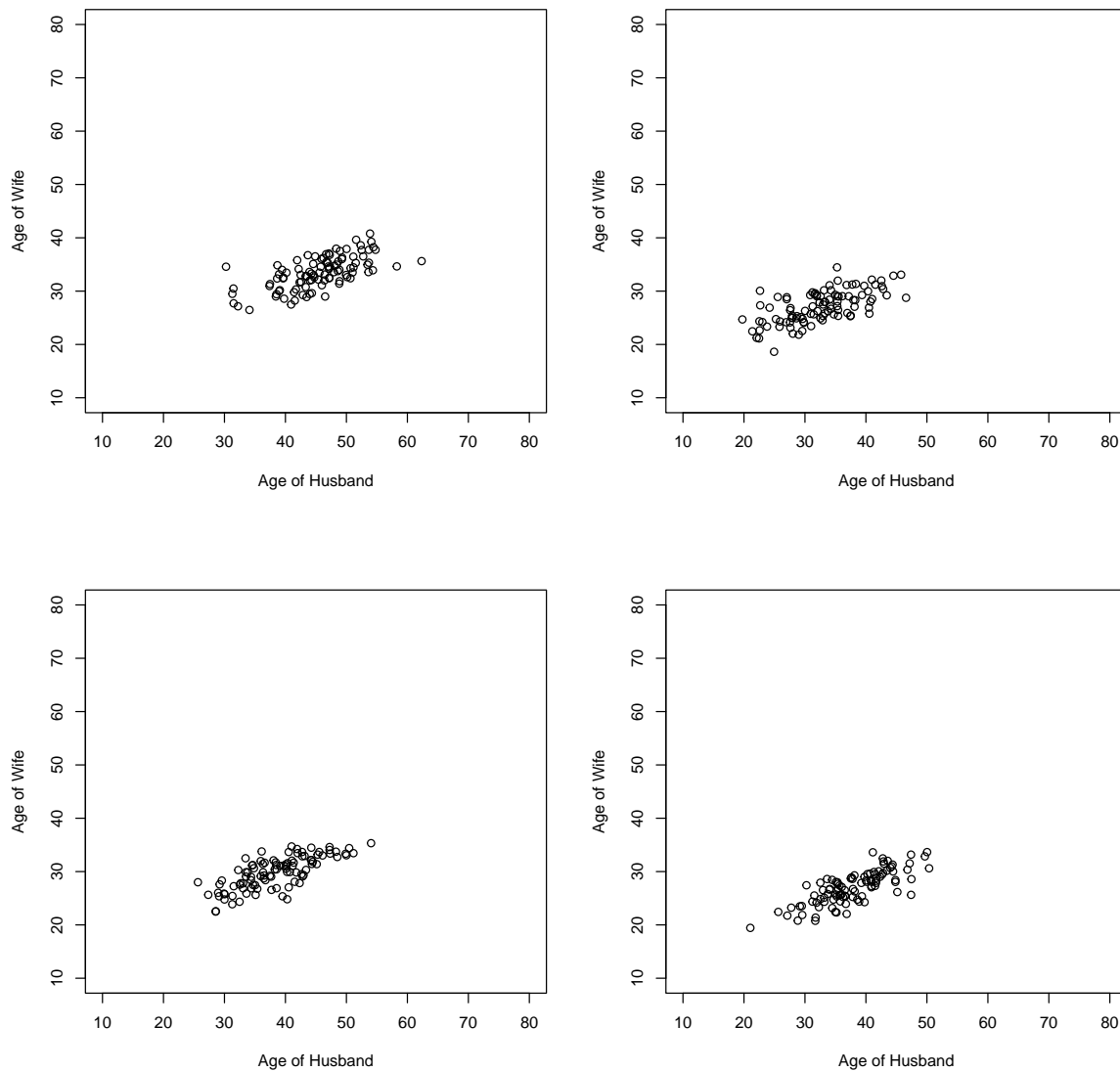
##

```
par(mfrow = c(2,2))
for(i in 1:num_datasets){
  #Sigma Sample
  sigma <- riwish(v = nu_0, S = psi)

  #Theta Sample
  theta <- mvrnorm(1, mu = mu_0, Sigma = sigma)

  #Drawing sample from distribution
  y <- mvrnorm(n, mu = theta, Sigma = sigma)

  plot(y[,1],y[,2],
        xlab = "Age of Husband",
        ylab = "Age of Wife",
        xlim = c(10,80),
        ylim = c(10,80))
}
```



The distributions generally look good, but we can see that the wife age in the top right plot seems quite low. To remedy this I change $\rho = .8$ and increase the mean age of the wife to 30.

```
mu_0 <- c(32,30)
nu_0 <- 50
sd_m <- 6
```



```

sd_w <- 3
corr <- .8
sigma_mean <- matrix(c(sd_m^2,corr*sd_m*sd_w,corr*sd_m*sd_w,sd_w^2),
                      ncol = 2)
psi <- sigma_mean*(nu_0 - 2 - 1)

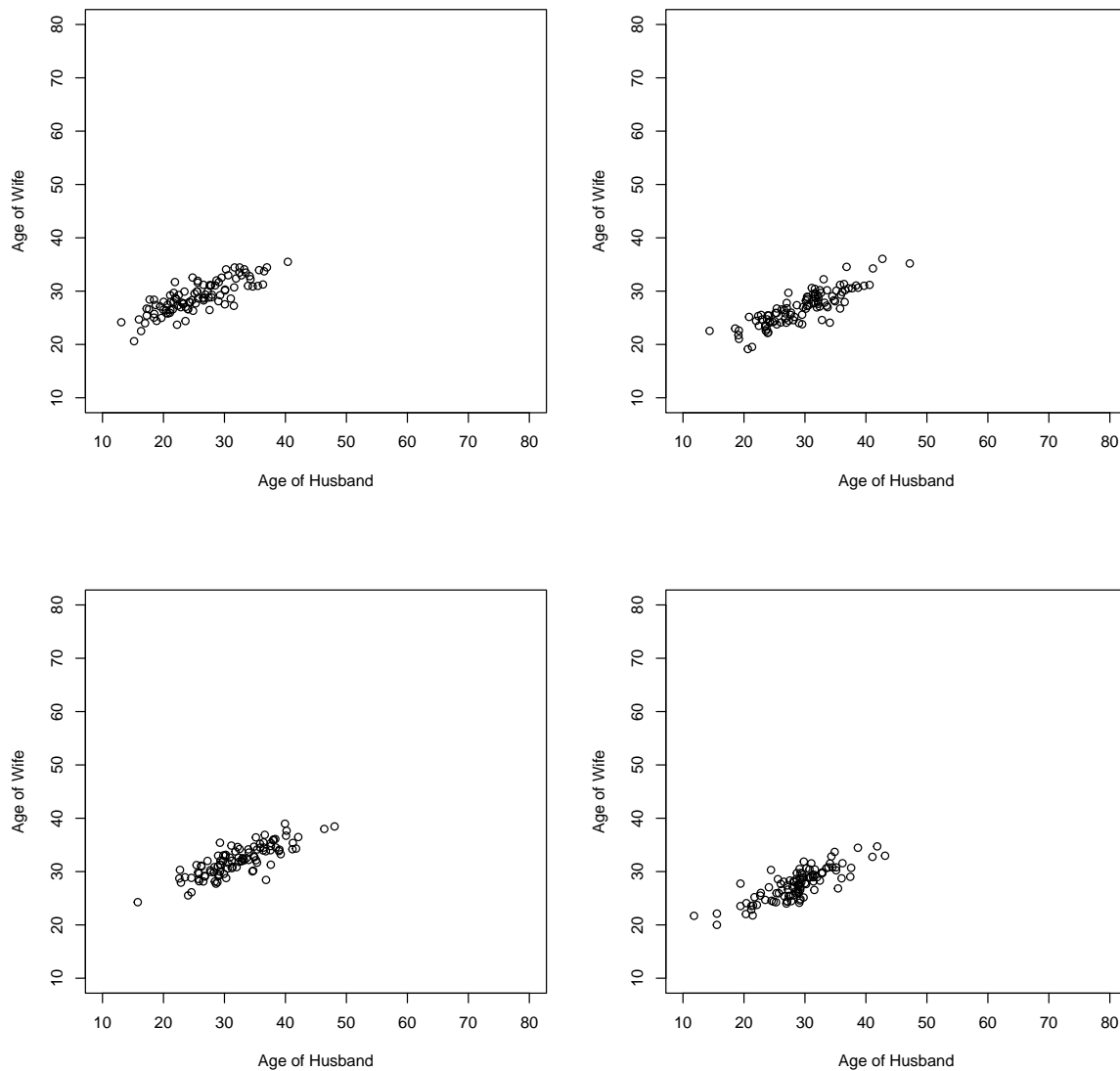
set.seed(16)
n <- 100
num_datasets <- 4
library(MASS)
library(MCMCpack)
par(mfrow = c(2,2))
for(i in 1:num_datasets){
  #Sigma Sample
  sigma <- riwish(v = nu_0, S = psi)

  #Theta Sample
  theta <- mvrnorm(1, mu = mu_0, Sigma = sigma)

  #Drawing sample from distribution
  y <- mvrnorm(n, mu = theta, Sigma = sigma)

  plot(y[,1],y[,2],
       xlab = "Age of Husband",
       ylab = "Age of Wife",
       xlim = c(10,80),
       ylim = c(10,80))
}

```



This looks closer to my beliefs.

c)

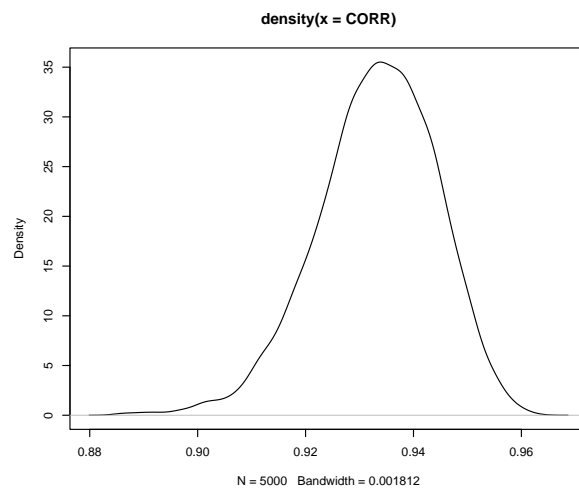
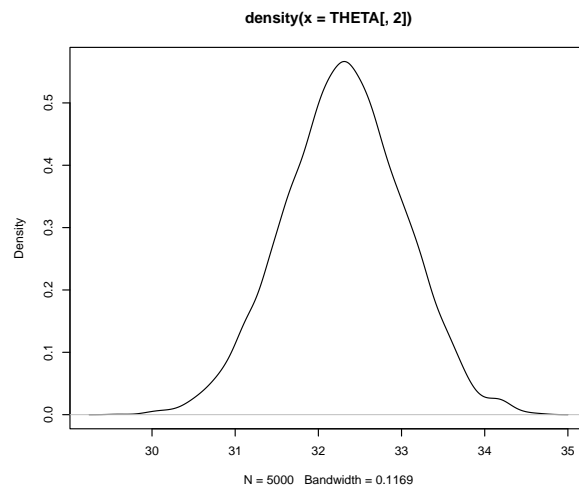
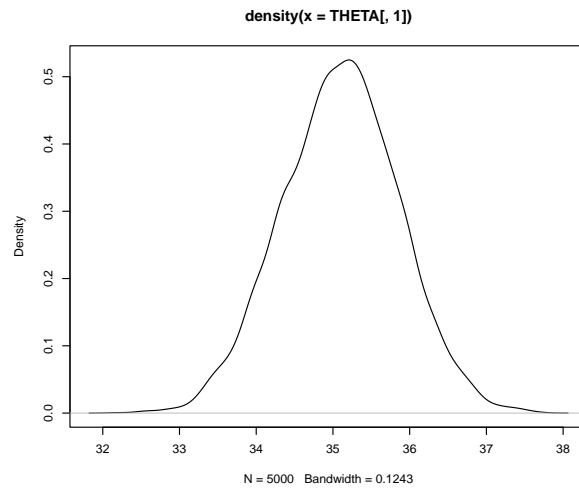
```
agehw <- read.table("agehw.dat", header = TRUE)
```

```

set.seed(15)
n <- dim(agehw)[1]; theta_bar <- apply(agehw, 2, mean)
L0 <- diag(2)*1
S0 <- sigma_mean
sigma <- cov(agehw)
THETA <- SIGMA <- NULL
set.seed(1)
for(s in 1:5000){
  #theta update
  Ln <- solve(solve(L0) + n*solve(sigma))
  mun <- Ln %*%( solve(L0) %*% mu_0 + n * solve(sigma) %*% theta_bar)
  theta <- rmvnorm(1, mean = c(mun), sigma = Ln)
  #Sigma update
  diff <- t(agehw) - c(theta)
  Sn <- S0 + diff %*% t(diff)
  sigma <- solve(rwish(nu_0 + n, solve(Sn)))
  #Saving
  THETA <- rbind(THETA, theta)
  SIGMA <- rbind(SIGMA, c(sigma))
}
CORR <- SIGMA[,2]/sqrt(SIGMA[,1]*SIGMA[,4])

par(mfrow = c(3,1))
plot(density(THETA[,1]))
plot(density(THETA[,2]))
plot(density(CORR))

```



```
theta_CIs <- apply(THETA, 2, quantile, probs = c(0.025, 0.975))
CORR_CI <- quantile(CORR, c(0.025, 0.975))
cat("\n95% credible intervals for theta:\n")
```

95% credible intervals for theta:

```
print(round(theta_CIs, 3))
```

```
      [,1]      [,2]
2.5%  33.59  30.889
97.5%  36.58  33.681
```

```
cat("95% CI for correlation:", round(CORR_CI, 3), "\n")
```

95% CI for correlation: 0.91 0.952

d)

i.

```
set.seed(25)
n <- dim(agehw)[1]; theta_bar <- apply(agehw, 2, mean)
S0 <- matrix(0, 2, 2)
nu_0 = 0
sigma <- cov(agehw)
THETA <- SIGMA <- NULL
for(s in 1:5000){
  #theta update
  theta <- rmvnorm(1, mean = theta_bar, sigma = sigma/n)
  #Sigma update
  diff <- t(agehw) - c(theta)
  Sn <- S0 + diff %*% t(diff)
  sigma <- solve(rwish(nu_0 + n, solve(Sn)))
  #Saving
  THETA <- rbind(THETA, theta)
  SIGMA <- rbind(SIGMA, c(sigma))
}
CORR <- SIGMA[,2]/sqrt(SIGMA[,1]*SIGMA[,4])
```

```
theta_CIs <- apply(THETA, 2, quantile, probs = c(0.025, 0.975))
CORR_CI <- quantile(CORR, c(0.025, 0.975))
cat("\n95% credible intervals for theta:\n")
```

95% credible intervals for theta:

```
print(round(theta_CIs, 3))
```

```
      ageh  agew
2.5%  41.686 38.329
97.5%  47.138 43.463
```

```
cat("95% CI for correlation:", round(CORR_CI, 3), "\n")
```

95% CI for correlation: 0.86 0.935

ii.

```
set.seed(30)
n <- dim(agehw)[1]; theta_bar <- apply(agehw, 2, mean)
L0 <- cov(agehw)
S0 <- cov(agehw)
nu_0 = 4
mu_0 = theta_bar
sigma <- cov(agehw)
THETA <- SIGMA <- NULL
for(s in 1:5000){
  #theta update
  Ln <- solve(solve(L0) + n*solve(sigma))
  mun <- Ln %*%( solve(L0) %*% mu_0 + n * solve(sigma) %*% theta_bar)
  theta <- rmvnorm(1, mean = c(mun), sigma = Ln)
  #Sigma update
  diff <- t(agehw) - c(theta)
  Sn <- S0 + diff %*% t(diff)
  sigma <- solve(rwish(nu_0 + n, solve(Sn)))
  #Saving
  THETA <- rbind(THETA, theta)
```

```

    SIGMA <- rbind(SIGMA, c(sigma))
  }
  CORR <- SIGMA[,2]/sqrt(SIGMA[,1]*SIGMA[,4])

  theta_CIs <- apply(THETA, 2, quantile, probs = c(0.025, 0.975))
  CORR_CI <- quantile(CORR, c(0.025, 0.975))
  cat("\n95% credible intervals for theta:\n")

```

95% credible intervals for theta:

```

print(round(theta_CIs, 3))

```

```

      [,1]    [,2]
2.5%  41.816  38.512
97.5%  47.036  43.393

```

```

cat("95% CI for correlation:", round(CORR_CI, 3), "\n")

```

95% CI for correlation: 0.862 0.934

iii.

```

set.seed(20)
n <- dim(agehw)[1]; theta_bar <- apply(agehw, 2, mean)
L0 <- diag(2)*10^5
S0 <- diag(2)*1000
nu_0 = 3
mu_0 = c(0,0)
sigma <- cov(agehw)
THETA <- SIGMA <- NULL
for(s in 1:5000){
  #theta update
  Ln <- solve(solve(L0) + n*solve(sigma))
  mun <- Ln %*%( solve(L0) %*% mu_0 + n * solve(sigma) %*% theta_bar)
  theta <- rmvnorm(1, mean = c(mun), sigma = Ln)
  #Sigma update
  diff <- t(agehw) - c(theta)

```

```

Sn <- S0 + diff %*% t(diff)
sigma <- solve(rwish(nu_0 + n, solve(Sn)))
#Saving
THETA <- rbind(THETA, theta)
SIGMA <- rbind(SIGMA, c(sigma))
}
CORR <- SIGMA[,2]/sqrt(SIGMA[,1]*SIGMA[,4])

theta_CIs <- apply(THETA, 2, quantile, probs = c(0.025, 0.975))
CORR_CI <- quantile(CORR, c(0.025, 0.975))
cat("\n95% credible intervals for theta:\n")

```

95% credible intervals for theta:

```
print(round(theta_CIs, 3))
```

```

      [,1]    [,2]
2.5% 41.550 38.205
97.5% 47.103 43.434

```

```
cat("95% CI for correlation:", round(CORR_CI, 3), "\n")
```

95% CI for correlation: 0.792 0.899

e)

The confidence intervals for θ are extremely similar to each other in part d, they are quite different from my strong prior in part c. I severely underestimated the ages for both groups. The correlation intervals are all pretty similar however.

My prior information was not helpful because it was not a good estimate of reality it seems, I was way off. Instead I would use the Jeffrey's prior to reflect that I do not have a prior information to add to the problem.

If n was smaller I'd consider doing some research to get a reasonable prior to center μ , but I would avoid making it too strong.