STATS551 Homework 3

Contribution:

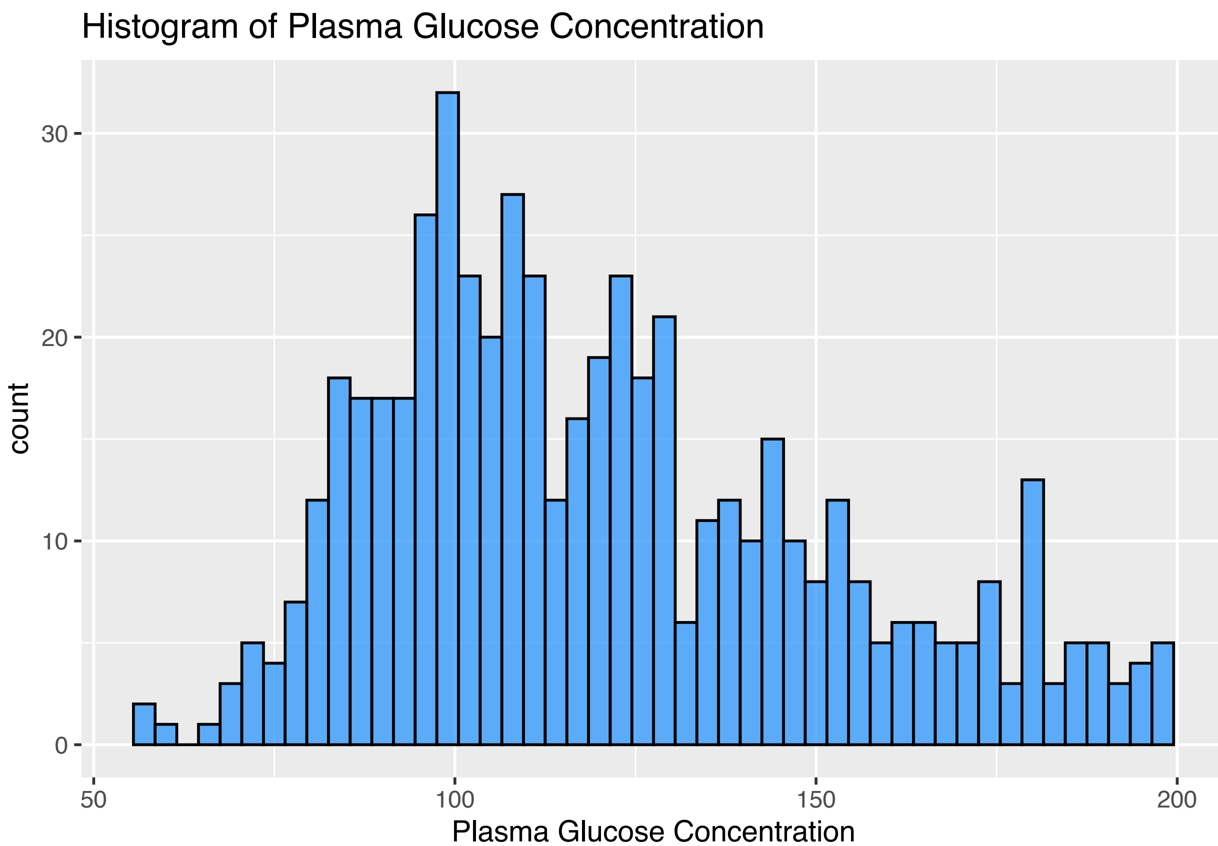
For this homework 3, we discussed all the problems and finished all the parts during the discussion.

So for each member in group 10, three of us are equally contributed.

Problem 1

(a)

```r
library(ggplot2)

# Load the dataset
glucose <- read.table("./glucose.dat")
colnames(glucose) <- c("conce")

ggplot(glucose) + geom_histogram(aes(x = conce), color ="black",
                                 fill ="dodgerblue", alpha =.7, binwidth=3) +
  xlab("Plasma Glucose Concentration") +
  ggtitle("Histogram of Plasma Glucose Concentration") + ylab ("count")
```



From the above histogram, the data seems to be a right skewed distribution, with two peaks around the value 100 and 160. It is not symmetric, and it is not like the shape of a normal distribution.

(b)
We have random variable X_i, with $P(X_i = 1) = p$ and $P(X_i = 2) = 1 - p$
We are given:

$$Y_i|X_i = 1 \sim \text{Normal}(\theta_1, \sigma_1^2)$$

$$Y_i|X_i = 2 \sim \text{Normal}(\theta_2, \sigma_2^2)$$

$$p \sim \text{Beta}(a, b)$$

$$\theta_j \sim \text{Normal}(\mu_0, \tau_0^2), j = 1, 2$$

$$\frac{1}{\sigma_j^2} \sim \text{Gamma}(\frac{\nu_0}{2}, \frac{\nu_0 \sigma_0^2}{2}), j = 1, 2$$

Let $\phi(Y_i; \theta, \sigma^2)$ denote the density of a normal distribution with mean θ and variance σ^2.

We have:

$$p(X_i = 1|Y_i, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) = \frac{p(X_i = 1|p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2)p(Y_i|X_i = 1, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2)}{p(Y_i|p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2)}$$

$$= \frac{p(X_i = 1|p)p(Y_i|X_i = 1, \theta_1, \sigma_1^2)}{p(X_i = 1|p)p(Y_i|X_i = 1, \theta_1, \sigma_1^2) + p(X_i = 2|p)p(Y_i|X_i = 2, \theta_2, \sigma_2^2)}$$

$$= \frac{p\phi(Y_i; \theta_1, \sigma_1^2)}{p\phi(Y_i; \theta_1, \sigma_1^2) + (1 - p)\phi(Y_i; \theta_2, \sigma_2^2)}$$

Similarly:

$$p(X_i = 2|Y_i, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) = \frac{(1 - p)\phi(Y_i; \theta_2, \sigma_2^2)}{p\phi(Y_i; \theta_1, \sigma_1^2) + (1 - p)\phi(Y_i; \theta_2, \sigma_2^2)}$$

Let $n_1 = \sum_{i=1}^n 1_{X_i=1}$, the number of observations with $X_i = 1$. Then $n_2 = n - n_1$.

Furthermore,

$$p(p|X_1, \dots, X_n, Y_1, \dots, Y_n, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) \propto p(p)p(X_1, \dots, X_n, Y_1, \dots, Y_n, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2|p)$$

$$\propto p(p)p(X_1, \dots, X_n|p)p(Y_1, \dots, Y_n|X_1, \dots, X_n, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2)p(\theta_1, \theta_2, \sigma_1^2, \sigma_2^2)$$

$$\propto p(p)p(X_1, \dots, X_n|p)$$

$$\propto p^{a-1}(1 - p)^{b-1}p^{n_1}(1 - p)^{n_2}$$

$$\sim \text{Beta}(a + n_1, b + n_2)$$

.

Continuing with θ_1 and θ_2:

Let $\mathbf{Y_1}$ denote the collection of Y_i such that the corresponding $X_1 = 1$, and define $\mathbf{Y_2}$ as the collection of Y_i such that the corresponding $X_i = 2$.

$$p(\theta_1|X_1\dots,X_n,Y_1,\dots,Y_n,p,\theta_2,\sigma_1^2,\sigma_2^2) \propto p(\theta_1,X_1,\dots,X_n,p,\theta_2,\sigma_1^2,\sigma_2^2)*$$
$$p(Y_1,\dots,Y_n|X_1,\dots,X_n,p,\theta_1,\theta_2,\sigma_1^2,\sigma_2^2)$$

$$\begin{gathered}\propto p(\theta_1)p(Y_1,\dots,Y_n|X_1,\dots,X_n,\theta_1,\theta_2,\sigma_1^2,\sigma_2^2)\\ \propto p(\theta_1)\phi(Y_i;\theta_1,\sigma_1^2)^{n_1}\phi(Y_i;\theta_2,\sigma_2^2)^{n_2}\\ \propto p(\theta_1)\phi(Y_i;\theta_1,\sigma_1^2)^{n_1}\\ \propto \exp(-\frac{1}{2\tau_0^2}(\theta_1-\mu_0)^2)\exp(-\frac{1}{2\sigma_1^2}\sum_{Y_i\in\mathbf{Y_1}}(Y_i-\theta_1)^2)\\ \sim \text{Normal}(\mu_{n_1},\tau_{n_1}^2)\end{gathered}$$

With:

$$\tau_{n_1}^2=\frac{1}{\frac{1}{\tau_0^2}+\frac{n_1}{\sigma_1^2}},\mu_{n_1}=\frac{\frac{\mu_0}{\tau_0^2}+\frac{n_1\bar{Y}_1}{\sigma_1^2}}{\frac{1}{\tau_0^2}+\frac{n_1}{\sigma_1^2}}$$

Similarly, $p(\theta_2|X_1\dots,X_n,Y_1,\dots,Y_n,p,\theta_1,\sigma_1^2,\sigma_2^2)\sim\text{Normal}(\mu_{n_2},\tau_{n_2}^2)$
With:

$$\tau_{n_2}^2=\frac{1}{\frac{1}{\tau_0^2}+\frac{n_2}{\sigma_2^2}},\mu_{n_1}=\frac{\frac{\mu_0}{\tau_0^2}+\frac{n_2\bar{Y}_2}{\sigma_2^2}}{\frac{1}{\tau_0^2}+\frac{n_2}{\sigma_2^2}}$$

Finally, σ_1^2 and σ_2^2.

$$p(\sigma_1^2|X_1\dots,X_n,Y_1,\dots,Y_n,p,\theta_1,\theta_2,\sigma_2^2) \propto p(\sigma_1^2,X_1,\dots,X_n,p,\theta_1,\theta_2,\sigma_2^2)*$$
$$p(Y_1,\dots,Y_n|X_1,\dots,X_n,p,\theta_1,\theta_2,\sigma_1^2,\sigma_2^2)$$

$$\begin{gathered}\propto p(\sigma_1^2)\phi(Y_i;\theta_1,\sigma_1^2)^{n_1}\phi(Y_i;\theta_2,\sigma_2^2)^{n_2}\\ \propto p(\sigma_1^2)\phi(Y_i;\theta_1,\sigma_1^2)^{n_1}\\ \propto (\sigma_1^2)^{-\frac{\nu_0}{2}-1}\exp(-\frac{1}{\sigma_1^2}\frac{\nu_0\sigma_0^2}{2})(\sigma_1^2)^{-n/2}\exp(-\frac{1}{2\sigma_1^2}\sum_{Y_i\in\mathbf{Y_1}}(Y_i-\theta_1)^2)\\ \sim \text{InverseGamma}(\frac{\nu_{n_1}}{2},\frac{\nu_{n_1}\sigma_{n_1}^2}{2})\end{gathered}$$

With:

$$\nu_{n_1}=\nu_0+n_1$$
$$\sigma_{n_1}^2=\frac{\nu_0\sigma_0^2+\sum_{Y_i\in\mathbf{Y_1}}(Y_i-\theta_1)^2}{\nu_{n_1}}=\frac{\nu_0\sigma_0^2+(n_1-1)S_{\mathbf{Y_1}}^2}{\nu_{n_1}}$$

Similarly, $p(\sigma_2^2|X_1,\dots,X_n,Y_1,\dots,Y_n,p,\theta_1,\theta_2,\sigma_1^2)\sim\text{InverseGamma}(\frac{\nu_{n_2}}{2},\frac{\nu_{n_2}\sigma_{n_2}^2}{2})$
With:

$$\nu_{n_2}=\nu_0+n_2$$
$$\sigma_{n_2}^2=\frac{\nu_0\sigma_0^2+\sum_{Y_i\in\mathbf{Y_2}}(Y_i-\theta_1)^2}{\nu_{n_2}}=\frac{\nu_0\sigma_0^2+(n_2-1)S_{\mathbf{Y_2}}^2}{\nu_{n_2}}$$

(c)

```r
library(coda)
y = glucose$conce
n = length(y)

# Priors
a = b =1
mu0 = 120
t20 = 200
s20 = 1000
nu0 = 10
S = 10000
THETA1 = numeric(S)
THETA2 = numeric(S)
YPRED = numeric(S)

# Posterior predictive
p = 0.5
theta1 = theta2 = mean(y)
s21 = s22 = var(y)
```

```r
#Gibbs sampling
set.seed(1)

for(s in 1:S){

  p1 = p * dnorm(y, theta1, sqrt(s21))
  p2 = (1- p) * dnorm(y, theta2, sqrt(s22))
  bernoulli_p = p1 / (p1 + p2)
  X = rbinom(n, 1, bernoulli_p)

  # Calculate group-specific summary statistics
  n1 = sum(X)
  n2 = n - n1
  y1 = y[X ==1]
  y2 = y[X ==0]
  ybar1 = mean(y1)
  ybar2 = mean(y2)
  yvar1 = var(y1)
  yvar2 = var(y2)

  # Sample p
  p = rbeta(1, a + n1, b + n2)

  # Sample thetas
  t2n1 =1/ (1/ t20 + n1 / s21)
  mun1 = (mu0 / t20 + n1 * ybar1 / s21) / (1/ t20 + n1 / s21)
  theta1 = rnorm(1, mun1, sqrt(t2n1))
  t2n2 =1/ (1/ t20 + n2 / s22)
  mun2 = (mu0 / t20 + n2 * ybar2 / s22) / (1/ t20 + n2 / s22)
  theta2 = rnorm(1, mun2, sqrt(t2n2))

  # Sample sigma^2s
```

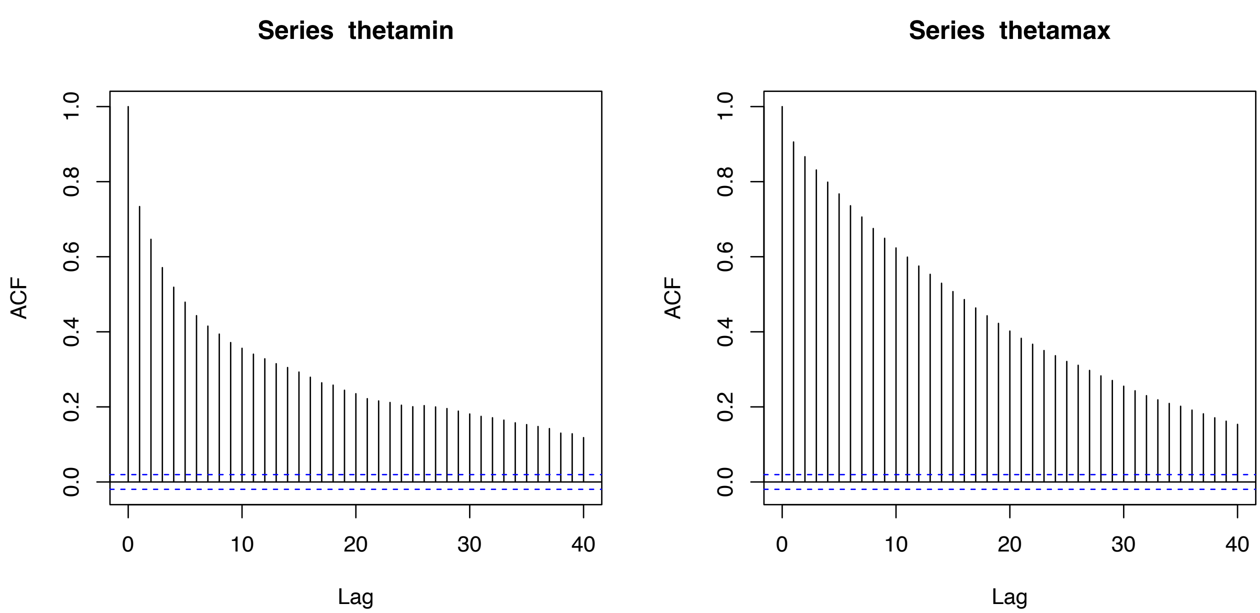
```
  nun1 = nu0 + n1
  s2n1 = (nu0 * s20 + (n1 -1) * yvar1 + n1 * (ybar1 - theta1)^2)/ nun1
  s21 = 1/ rgamma(1, nun1 /2, s2n1 * nun1 /2)
  nun2 = nu0 + n2
  s2n2 = (nu0 * s20 + (n2 -1) * yvar2 + n2 * (ybar2 - theta2)^2)/ nun2
  s22 = 1/ rgamma(1, nun2 /2, s2n2 * nun2 /2)

  # Sample posterior predictive
  xpred = runif(1) < p
  ypred = ifelse(xpred, rnorm(1, theta1, sqrt(s21)), rnorm(1, theta2, sqrt(s22)))

  # Store
  THETA1[s] = theta1
  THETA2[s] = theta2
  YPRED[s] = ypred
}

thetamin = pmin(THETA1 , THETA2)
thetamax = pmax(THETA1 , THETA2)

effectiveSize(thetamin)
```

```
##     var1
## 409.9274
```

```
effectiveSize(thetamax)
```

```
##     var1
## 217.5245
```

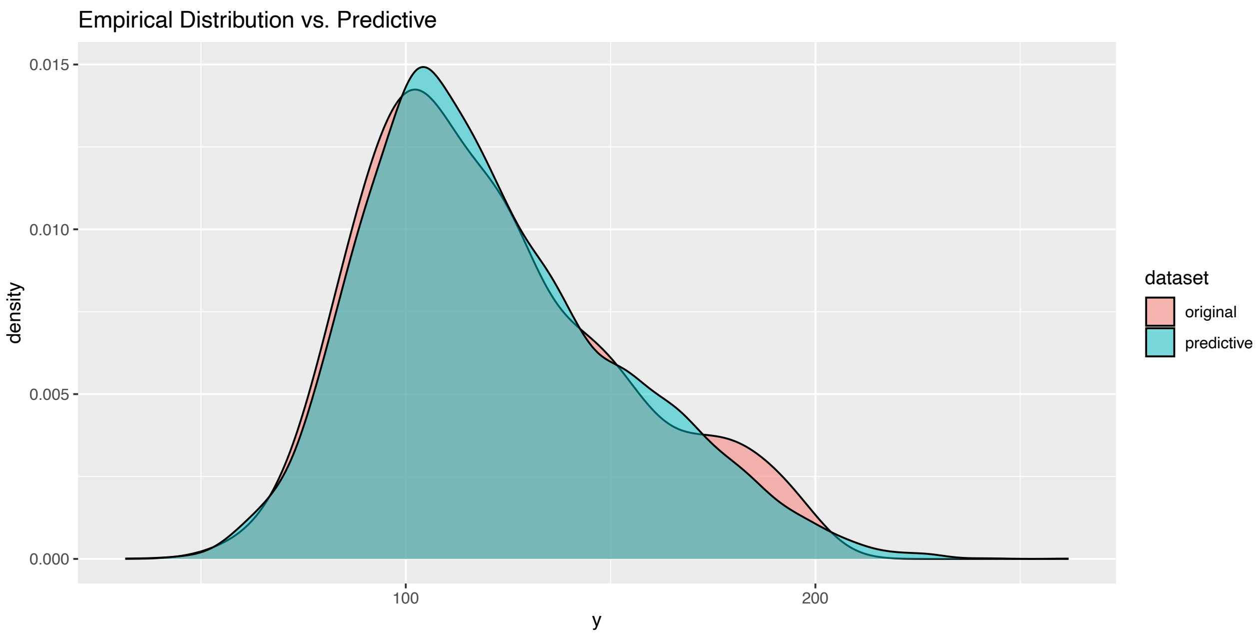
```
par(mfcol=c(1,2))
acf(thetamin)
acf(thetamax)
```



For Monte carlo sample size of 10000, the ESS for $\theta_{(1)}$ is 453.2992, and ESS for $\theta_{(2)}$ is 228.5868. Also from two above ACF plots, $\theta_{(2)}$ has larger autocorrelations.

(d)

```r
YCOMP = rbind(data.frame(y = YPRED , dataset ='predictive'),
              data.frame(y = y, dataset ='original'))

ggplot(YCOMP, aes(x = y, fill = dataset)) +
  geom_density(alpha =0.5) +
  ggtitle("Empirical Distribution vs. Predictive")
```



According to the density plot, the predictive distribution has similar right-skewness of the empirical distribution. Therefore GMM with 2 components is a reasonable model.

Problem 2

(a)

In order to obtain a semiconjugate prior distribution for $\theta = (\theta_h, \theta_w)^T$ and Σ, we can use the multivariate normal model. We model our sampling distribution $p(Y_1, \ldots, Y_n|\theta, \Sigma) \sim^{iid} N_2(\theta, \Sigma)$. Then, we have the priors $p(\theta) \sim N_2(\mu_0, \Sigma_0)$ and $p(\Sigma) \sim \text{inverseWishart}(\nu_0, S_0^{-1})$.

For initial guess of hyperparameters, let $\mu_0 = (50, 50)^T$. To ensure most of the mass is concentrated on the range [0,100], we set $\sigma_{11} = \sigma_{22} = (\frac{50}{2})^2 = 625$. Furthermore, let the covariance $\sigma_{12} = \sigma_{21} = 0.5\sigma_{11} = 312.5$

$$\Sigma_0 = \begin{bmatrix} 625 & 312.5 \\ 312.5 & 625 \end{bmatrix}$$

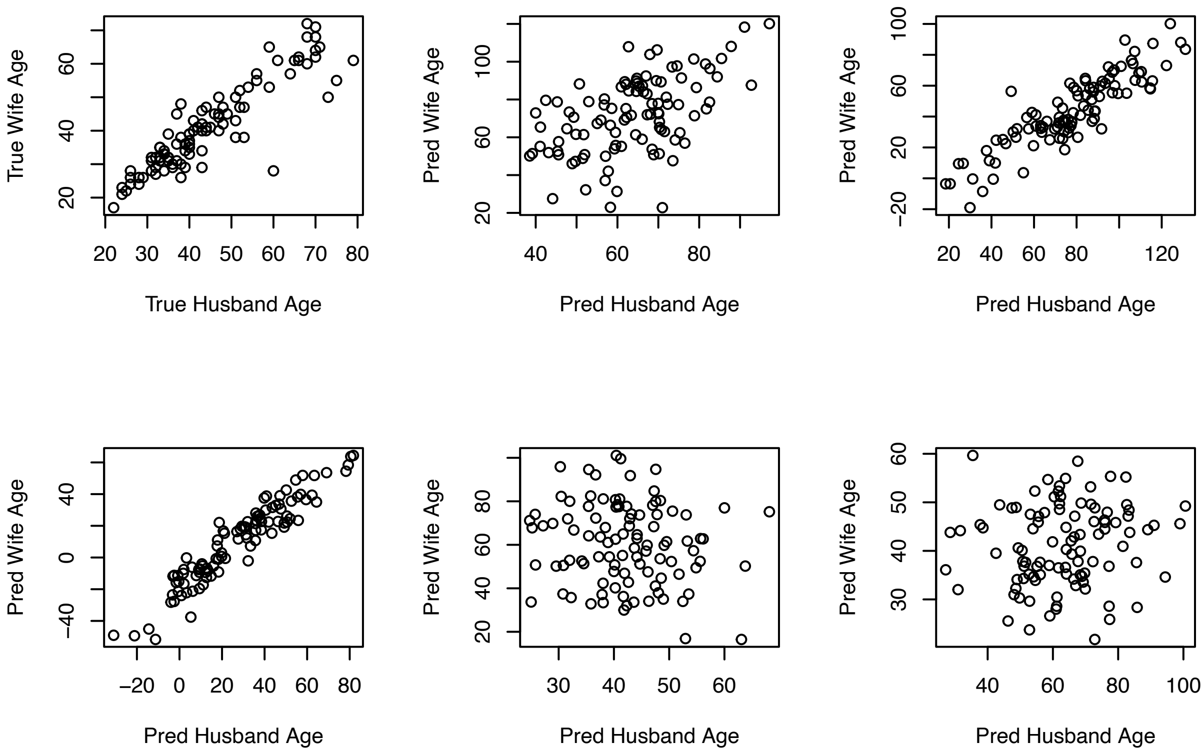
.

For the hyperparameters for Σ, we take $S_0 = \Sigma_0$ and set $\nu_0 = p + 2 = 4$ to allow for sufficient spread around Σ_0.

(b)

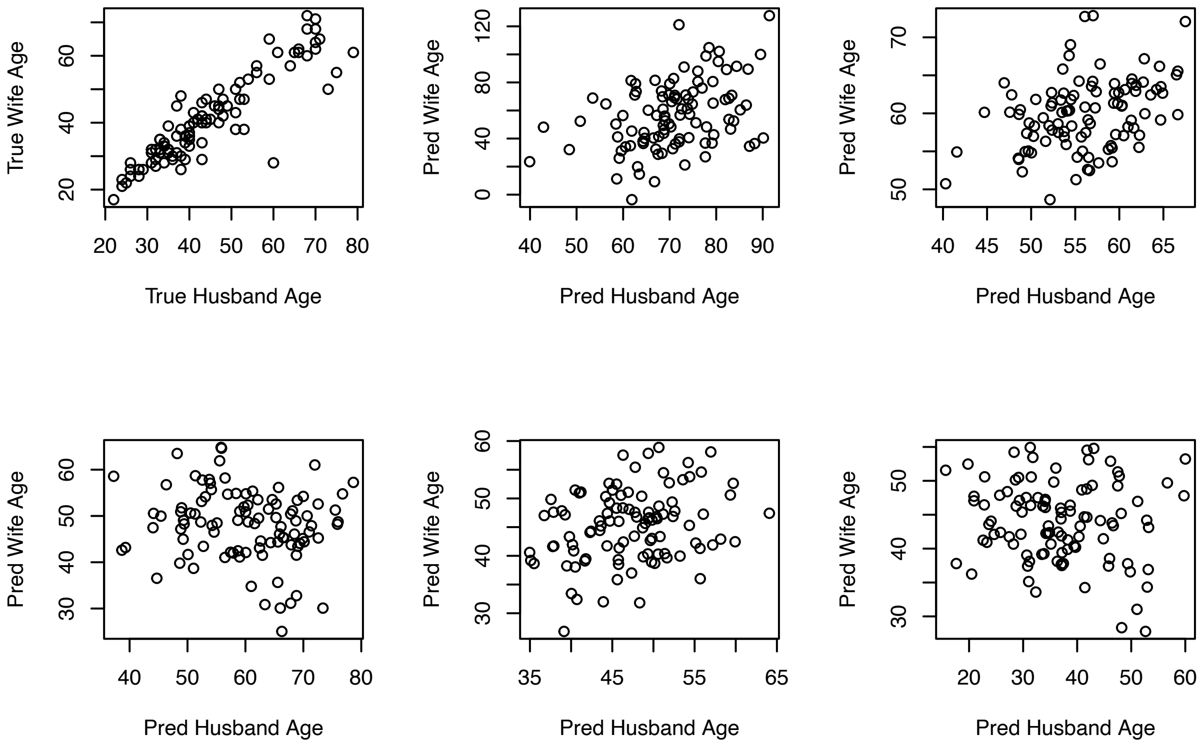
To generate a prior predictive dataset, we will first sample(θ, Σ) from the priors defined in 2a. Then we will obtain $Y_i \sim N_2(\theta, \Sigma)$ for $i = 1, \ldots, 100$.

```r
agehw <- read.table("./agehw.dat", header = T)
set.seed(551)
n = 100 ## iterations
mu0 <- c(50,50) ## hyperparameters
Sigma0 <- matrix(c(625, 312.5, 312.5, 625), nrow = 2, ncol = 2)
nu0 <- 4

### Obtain 100 Yi's from prior estimates of parameters
YS1 = matrix(0, nrow = 100, ncol = 5)
YS2 = matrix(0, nrow = 100, ncol = 5)

for(s in 1:5){
    theta = mvrnorm(1, mu0, Sigma0) ## Sample theta from N(mu0, Sigma0)
    sig = MCMCpack::riwish(nu0, Sigma0) ## Sample Sigma from InvWish(nu0, Sigma0^-1)
    vals = mvrnorm(100, theta, sig) ## Sample Y from N(theta, Sigma)
    YS1[,s] = vals[,1]
    YS2[,s] = vals[,2]
}

par(mfrow = c(2,3))
plot(agehw$ageh, agehw$agew, xlab = "True Husband Age" ,ylab = "True Wife Age")
plot(YS1[,1], YS2[,1], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,2], YS2[,2], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,3], YS2[,3], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,4], YS2[,4], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,5], YS2[,5], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
```



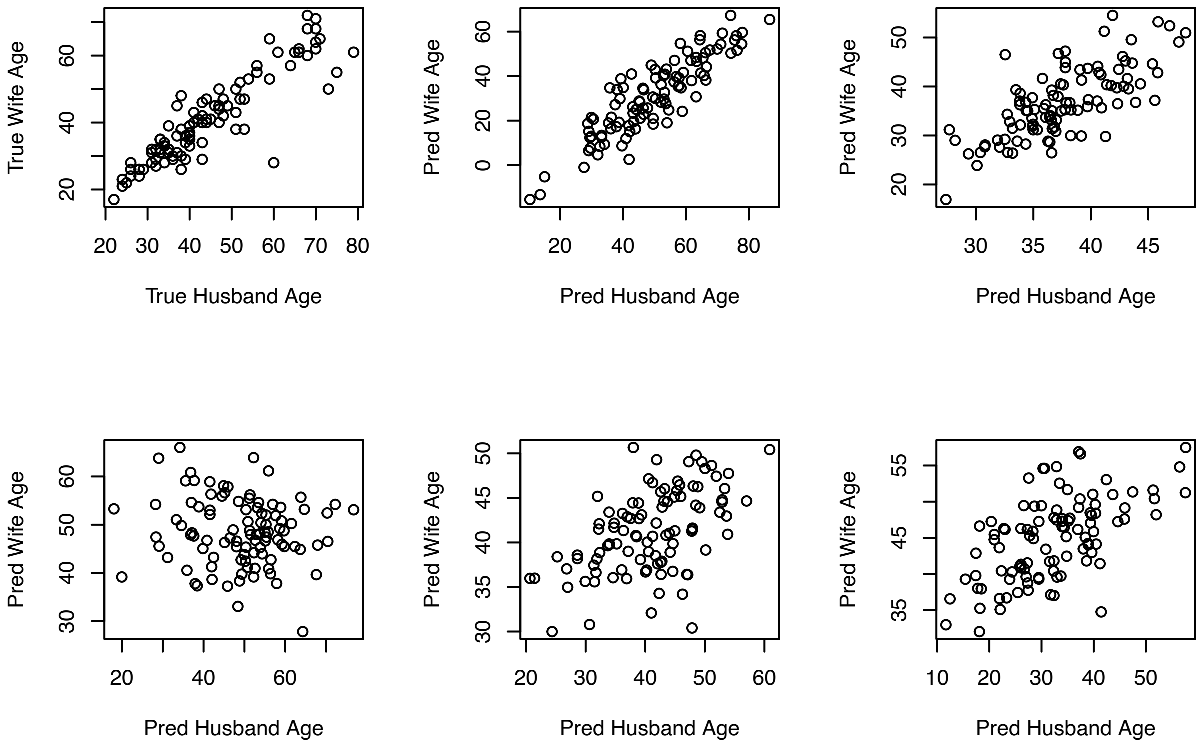
As we can see, there is too much variance in our results. We will try a prior with less variance.

```
mu0 <- c(50,50) ## hyperparameters
Sigma0 <- matrix(c(156.25, 78.125, 78.125, 156.25), nrow = 2, ncol = 2)
nu0 <- 4


### Obtain 100 Yi's from prior estimates of parameters
YS1 = matrix(0, nrow = 100, ncol = 5)
YS2 = matrix(0, nrow = 100, ncol = 5)

for(s in 1:5){
    theta = mvrnorm(1, mu0, Sigma0) ## Sample theta from N(mu0, Sigma0)
    sig = MCMCpack::riwish(nu0, Sigma0) ## Sample Sigma from InvWish(nu0, Sigma0^-1)
    vals = mvrnorm(100, theta, sig) ## Sample Y from N(theta, Sigma)
    YS1[,s] = vals[,1]
    YS2[,s] = vals[,2]
}

par(mfrow = c(2,3))
plot(agehw$ageh, agehw$agew, xlab = "True Husband Age" ,ylab = "True Wife Age")
plot(YS1[,1], YS2[,1], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,2], YS2[,2], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,3], YS2[,3], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,4], YS2[,4], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,5], YS2[,5], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
```



The true data is still even more strongly correlated than our results,and still less variable. Furthermore, the mean vector should be adjusted based on the empirical data. Again we will formulate a new prior, with a new mean vector, and with a Σ_0 with smaller diagonal elements and increased covariance (as a proportion of σ_{11}).

```
mu0 <- c(42,42) ## hyperparameters
Sigma0 <- matrix(c(150, 90, 90, 150), nrow = 2, ncol = 2)
nu0 <- 4

### Obtain 100 Yi's from prior estimates of parameters
YS1 = matrix(0, nrow = 100, ncol = 5)
YS2 = matrix(0, nrow = 100, ncol = 5)

for(s in 1:5){
    theta = mvrnorm(1, mu0, Sigma0) ## Sample theta from N(mu0, Sigma0)
    sig = MCMCpack::riwish(nu0, Sigma0) ## Sample Sigma from InvWish(nu0, Sigma0^-1)
    vals = mvrnorm(100, theta, sig) ## Sample Y from N(theta, Sigma)
    YS1[,s] = vals[,1]
    YS2[,s] = vals[,2]
}

par(mfrow = c(2,3))
plot(agehw$ageh, agehw$agew, xlab = "True Husband Age" ,ylab = "True Wife Age")
plot(YS1[,1], YS2[,1], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,2], YS2[,2], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,3], YS2[,3], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,4], YS2[,4], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
plot(YS1[,5], YS2[,5], xlab = "Pred Husband Age" ,ylab = "Pred Wife Age")
```



Out of all priors tested, the best performing prior is the final one, in the sense that the repeated prior predictive datasets have the best fit to the original data, in the most iterations. Therefore, we will use the prior:

$p(\theta) \sim N_2(\mu_0, \Sigma_0)$ and $p(\Sigma) \sim \text{inverseWishart}(\nu_0, S_0^{-1})$

with the following hyperparameters:

$$\mu_0 = (42, 42)^T$$

$$\Sigma_0 = \begin{bmatrix} 150 & 90 \\ 90 & 150 \end{bmatrix}$$

$$\nu_0 = p + 2 = 4$$

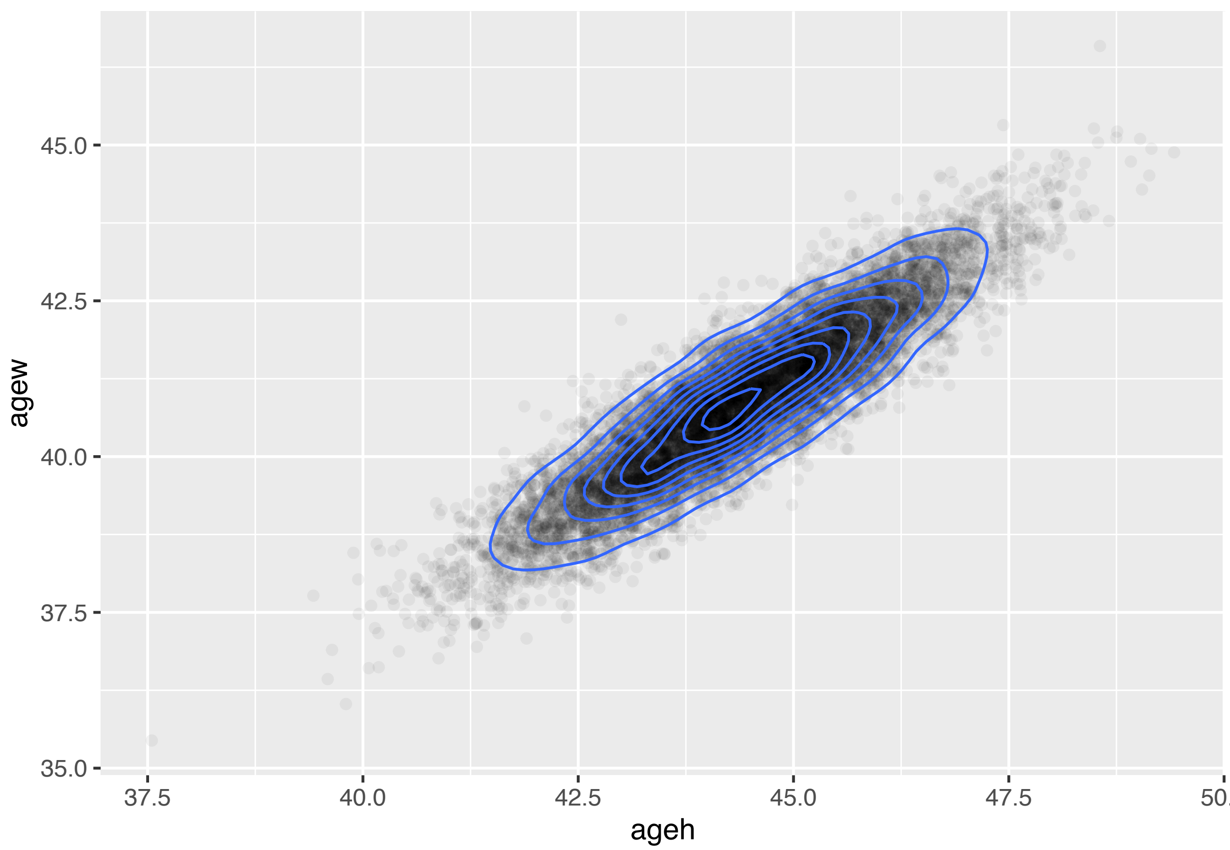
$$S_0 = \Sigma_0$$

(c)

```
library(ggplot2)
library(MASS)
agehw <- read.table("./agehw.dat", header = T)
ybar <- apply(agehw, 2, mean)
sigma <- cov(agehw)

mu0 <- c(42,42)
nu0 <- 4
L0 <- matrix(c(150, 90, 90, 150), nrow = 2, ncol = 2)
S0 <- matrix(c(150, 90, 90, 150), nrow = 2, ncol = 2)
ybar <- apply(agehw, 2, mean)
Sigma <- cov(agehw)
n <- dim(agehw)[1]

THETA<- SIGMA <- NULL
```

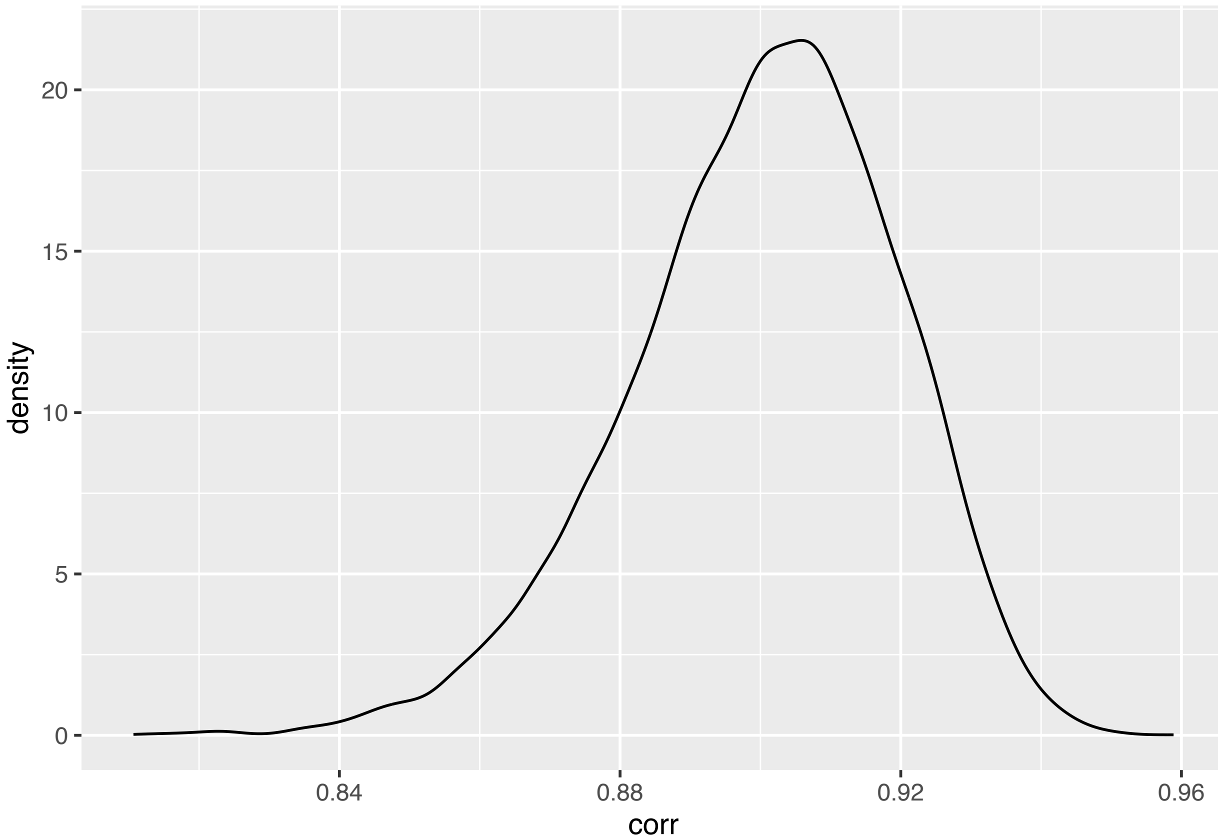
```r
set.seed(42)

for(s in 1:10000){

  ###Sample theta
  Ln <- solve(solve(L0) + n * solve(Sigma))
  mun <- Ln %*% (solve(L0) %*%  mu0 + n * solve(Sigma) %*% ybar)
  theta <- mvrnorm(1, mun, Ln)

  ###Sample Simga
  Sn <- S0 + (t(agehw) - c(theta)) %*% t(t(agehw) - c(theta))
  Sigma <- solve(rWishart(1, nu0 + n, solve(Sn))[,,1])

  ###Save results

  THETA <- rbind(THETA, theta)
  SIGMA <- rbind(SIGMA, c(Sigma))
}
```

```r
ggplot(as.data.frame(THETA), aes(x = ageh, y = agew)) + geom_point(alpha = .05) +
  geom_density2d()
```




```r
cov <- SIGMA[,2]
varh <- SIGMA[,1]
varw <- SIGMA[,4]

corr <- cov/sqrt(varh*varw)
```

```r
corr <- data.frame(corr)

ggplot(data = corr, aes(x = corr)) + geom_density()
```



The marginal posterior density of the correlation between Y_h and Y_w is above.

```r
### theta_h CI
quantile(THETA[,1], c(.025,.975))
```

```
##     2.5%    97.5%
## 41.70745 47.04493
```

The 95% Confidence Interval of θ_h is (41.70745, 47.04493)

```r
### theta_w CI
quantile(THETA[,2], c(.025,.975))
```

```
##     2.5%    97.5%
## 38.39368 43.40601
```

The 95% Confidence Interval of θ_w is (38.39368, 43.40601)

```r
### correlation coefficient CI
quantile(corr[,1], c(.025,.975))
```

```
##      2.5%     97.5%
## 0.8585927 0.9326866
```

The 95% Confidence Interval of correlation coefficient is (0.8585927, 0.9326866)

(d)(i)

```r
###prior parameters
mu0 <- c(0,0)
nu0 <- 3
L0 <- matrix(c(10^5, 0, 0, 10^5), nrow = 2, ncol = 2)
S0 <- matrix(c(1000, 0, 0, 1000), nrow = 2, ncol = 2)
ybar <- apply(agehw, 2, mean)
Sigma <- cov(agehw)
n <- dim(agehw)[1]

THETA1 <-SIGMA1 <- NULL

set.seed(42)

for(s in 1:10000){

  ###Sample theta
  Ln <- solve(solve(L0) + n * solve(Sigma))
  mun <- Ln %*% (solve(L0) %*%  mu0 + n * solve(Sigma) %*% ybar)
  theta <- mvrnorm(1, mun, Ln)

  ###Sample Simga
  Sn <- S0 + (t(agehw) - c(theta)) %*% t( t(agehw) - c(theta))
  Sigma <- solve(rWishart(1, nu0 + n, solve(Sn))[,,1])

  ###Save results

  THETA1 <- rbind(THETA1, theta)
  SIGMA1 <- rbind(SIGMA1, c(Sigma))
}
```

```r
cov1 <- SIGMA1[,2]
varh1 <- SIGMA1[,1]
varw1 <- SIGMA1[,4]

corr1 <- cov1/sqrt(varh1*varw1)
corr1 <- data.frame(corr1)
```

```r
### theta_h CI
quantile(THETA1[,1], c(.025,.975))
```

```
##     2.5%    97.5%
## 41.62057 47.15739
```

The 95% Confidence Interval of θ_h is (41.62057, 47.15739)

```r
### theta_w CI
quantile(THETA1[,2], c(.025,.975))
```

```
##     2.5%    97.5%
## 38.28229 43.49813
```

The 95% Confidence Interval of θ_w is (38.28229, 43.49813)

```r
### correlation coefficient CI
quantile(corr1[,1], c(.025,.975))
```

```
##      2.5%     97.5%
## 0.7929508 0.9005727
```

The 95% Confidence Interval of correlation coefficient is (0.7929508, 0.9005727)

(d)(ii)

Given the Jeffreys' prior, from HW3 Question 4, we derive that:

$$p_J(\theta|\Sigma, \mathbf{y}_1, \cdots, \mathbf{y}_n) \propto |\Sigma/n|^{-1/2} \exp\left[-\frac{1}{2}(\theta-\bar{y})^T \left(\frac{\Sigma}{n}\right)^{-1} (\theta-\bar{y})\right] \sim N\left(\bar{y}, \frac{\Sigma}{n}\right)$$

$$p_J(\Sigma|\mathbf{y}_1, \cdots, \mathbf{y}_n, \theta) \propto |\Sigma|^{-((n+p+2)/2)} \exp\left[-\frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}S\right)\right] \sim \text{Inv-Wishart}_{n+1}(S^{-1})$$

```r
###prior parameters
library(MASS)

ybar <- apply(agehw, 2, mean)
Sigma <- cov(agehw)
n <- dim(agehw)[1]

THETA2 <-SIGMA2 <- NULL

set.seed(42)

for(s in 1:10000){

  ###Sample theta
  co <- Sigma/n
  mu <- ybar
  theta <- mvrnorm(1, mu, co)

  ###Sample Sigma
  Sig <- (t(agehw) - c(theta)) %*% t(t(agehw) - c(theta))
  Sigma <- solve(rWishart(1, n+1, solve(Sig))[,,1])

  ###Save results

  THETA2 <- rbind(THETA2, theta)
  SIGMA2 <- rbind(SIGMA2, c(Sigma))
}
```

```r
cov2 <- SIGMA2[,2]
varh2 <- SIGMA2[,1]
varw2 <- SIGMA2[,4]

corr2 <- cov2/sqrt(varh2*varw2)
corr2 <- data.frame(corr2)
```

```r
### theta_h CI
quantile(THETA2[,1], c(.025,.975))
```

```
##     2.5%    97.5%
## 41.66548 47.11193
```

The 95% Confidence Interval of θ_h is (41.66548, 47.11193)

```r
### theta_w CI
quantile(THETA2[,2], c(.025,.975))
```

```
##     2.5%    97.5%
## 38.35341 43.46039
```

The 95% Confidence Interval of θ_w is (38.35341, 43.46039)

```r
### correlation coefficient CI
quantile(corr2[,1], c(.025,.975))
```

```
##      2.5%     97.5%
## 0.8612923 0.9349475
```

The 95% Confidence Interval of correlation coefficient is (0.8612923, 0.9349475)

(e)

The confidence intervals for θ obtained using a diffuse prior and Jeffreys' prior were slightly **larger** than the confidence intervals obtained using in (c).

For estimating Σ, diffuse prior $>$ Jeffreys' prior $\approx$ part(c) .

Since using (c) prior information resulted in more precise posterior distributions, We would prefer it to the diffuse prior and Jeffreys' prior. On the other hand, if the sample size was 25, We would rather have the diffuse prior and Jeffreys' prior due to the relative lack of prior information. Part (c) prior information highly depends on the data samples.

Problem 3

(a)

By definition of this mixture distribution, we have

$$
\begin{aligned}
&p(Z_i = j) = p_j,\ j = 1,2, \quad Z_i \text{ i.i.d} \\
&X_i|Z_i = 1 \sim N(\mu_1, \Sigma_1), \quad X_i|Z_i = 2 \sim N(\mu_2, \Sigma_2), \quad X_i \text{ i.i.d}
\end{aligned}
$$

It's worth noting that Z_i follows a discrete distribution with only two parameters and we have $p_1 + p_2 = 1$, so it's more convenient to express the distribution as Bernoulli, i.e.

$$
C_i = (Z_i - 1)|p_1 \overset{\text{i.i.d}}{\sim} \text{Bern}(p_2),\ \text{i.e. } p(C_i = 1) = p_2
$$

So, in the following, we use C_i, p_2, which is equavilent to Z_i, p_1, p_2

We need to specify priors for $p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2$. Note that these parameters exhibit certain structures, i.e. we should assume the following independencies

$$
p(p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2) = p(p_2) \cdot p(\mu_1, \Sigma_1) \cdot p(\mu_2, \Sigma_2)
$$

This assumption is natural, since in a mixture model, different components should not be interrelated, and the same holds between components and class labels.

For simplicity, we use beta distribution for $p(p_2)$ and Jeffreys' prior for $p(\mu_1, \Sigma_1)$ and $p(\mu_2, \Sigma_2)$. That is

$$
\begin{aligned}
&p(p_2) \sim \text{Beta}(\alpha, \beta) \\
&p(\mu_1, \Sigma_1) \propto |\Sigma_1|^{-5/2} \\
&p(\mu_2, \Sigma_2) \propto |\Sigma_2|^{-5/2}
\end{aligned}
$$

For hyperparameter (α, β), we didn't get any extra information about the clusters, so we just choose $\alpha = \beta = 1$.

Overall, the model is

$$
\begin{aligned}
&p(p_2) \sim \text{Beta}(\alpha, \beta) \\
&p(\mu_1, \Sigma_1) \propto |\Sigma_1|^{-5/2} \\
&p(\mu_2, \Sigma_2) \propto |\Sigma_2|^{-5/2} \\
&p(p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2) = p(p_2) \cdot p(\mu_1, \Sigma_1) \cdot p(\mu_2, \Sigma_2) \\
&C_i|p_2 \overset{\text{i.i.d}}{\sim} \text{Bern}(p_2) \\
&X_i|C_i = 0 \sim N(\mu_1, \Sigma_1),\ X_i|C_i = 1 \sim N(\mu_2, \Sigma_2),\ X_i \text{ i.i.d}
\end{aligned}
$$

(b)

Note that only X_is are observable and it's helpful to introduce C_is to the posterior.
Joint posterior

$$
\begin{aligned}
&p(p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2, C_1, \cdots, C_n|\mathbf{X}) \\
\propto\, &p(\mathbf{X}|p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2, C_1, \cdots, C_n) \cdot p(p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2, C_1, \cdots, C_n) \\
\propto\, &\prod_i p(X_i|C_i, \mu_1, \Sigma_1, \mu_2, \Sigma_2) \cdot p(C_1, \cdots, C_n|p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2) \cdot p(p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2) \\
\propto\, &\left(\prod_i \Big[(1 - C_i)N(X_i|\mu_1, \Sigma_1) + C_i N(X_i|\mu_2, \Sigma_2)\Big]\right) \cdot \left(\prod_i p(C_i|p_2)\right) \cdot p(p_2) \cdot p(\mu_1, \Sigma_1) \cdot p(\mu_2, \Sigma_2)
\end{aligned}
$$

Now we consider the full conditional distributions

$$\begin{aligned}
p(p_2|\cdot) &\propto \left(\prod_i p(C_i|p_2)\right) \cdot p(p_2) \\
&\propto \prod_i (1-p_2)^{1-C_i} p_2^{C_i} \\
&\propto p_2^{\Sigma C_i}(1-p_2)^{n-\Sigma C_i} \\
&\sim \text{Beta}(\Sigma C_i + 1, n - \Sigma C_i + 1)
\end{aligned}$$

$$p(\mu_1|\cdot) \propto \left(\prod_i \Big[(1-C_i)N(X_i|\mu_1,\Sigma_1) + C_i N(X_i|\mu_2,\Sigma_2)\Big]\right) \cdot p(\mu_1,\Sigma_1)$$

Note that C_i is a binary variable, so it's convenient to rewrite

$$\begin{aligned}
p(\mu_1|\cdot) \propto \left(\prod_i \left|\Sigma_i^{\text{mix}}\right|^{-1/2}\right) &\exp\left[-\frac{1}{2}\sum_{i:\,C_i=0}(X_i-\mu_1)^T\,\Sigma_1^{-1}\,(X_i-\mu_1)\right] \\
&\times \exp\left[-\frac{1}{2}\sum_{i:\,C_i=1}(X_i-\mu_2)^T\,\Sigma_2^{-1}\,(X_i-\mu_2)\right] \\
\propto \exp&\left[-\frac{1}{2}\sum_{i:\,C_i=0}(\mu_1-X_i)^T\,\Sigma_1^{-1}\,(\mu_1-X_i)\right] \\
\propto \exp&\left[-\frac{1}{2}\Big(\operatorname{tr}\left(\Sigma_1^{-1}S_1\right) + n_1(\mu_1-\bar{X}_1)^T\Sigma_1^{-1}(\mu_1-\bar{X}_1)\Big)\right] \\
\sim N(&\bar{X}_1, \frac{\Sigma_1}{n_1})
\end{aligned}$$

in which

$$\begin{aligned}
\Sigma_i^{\text{mix}} &= (1-C_i)\Sigma_1 + C_i\Sigma_2 \\
n_1 &= n - \sum_i C_i \\
\bar{X}_1 &= \frac{1}{n_1}\sum_{i:\,C_i=0} X_i \\
S_1 &= \sum_{i:\,C_i=0}^{n}(X_i-\bar{X}_1)(X_i-\bar{X}_1)^T
\end{aligned}$$

Similarly, we can get

$$p(\mu_2|\cdot) \sim N(\bar{X}_2, \frac{\Sigma_2}{n_2})$$

in which

$$\begin{aligned}
n_2 &= \sum_i C_i \\
\bar{X}_2 &= \frac{1}{n_2}\sum_{i:\,C_i=1} X_i
\end{aligned}$$

For Σ_1 and Σ_2, we have

$$
\begin{aligned}
p(\Sigma_1|\cdot) &\propto \left(\prod_i \Big[(1-C_i)N(X_i|\mu_1,\Sigma_1) + C_i N(X_i|\mu_2,\Sigma_2)\Big]\right)\cdot p(\mu_1,\Sigma_1) \\
&\propto \left(\prod_i \left|\Sigma_i^{\text{mix}}\right|^{-1/2}\right)\exp\left[-\frac{1}{2}\sum_{i:\,C_i=0}(X_i-\mu_1)^T\,\Sigma_1^{-1}\,(X_i-\mu_1)\right]|\Sigma_1|^{-5/2} \\
&\propto |\Sigma_1|^{-n_1/2}|\Sigma_1|^{-5/2}\exp\left[-\frac{1}{2}\sum_{i:\,C_i=0}(X_i-\mu_1)^T\,\Sigma_1^{-1}\,(X_i-\mu_1)\right] \\
&\propto |\Sigma_1|^{-(n_1+1+3+1)/2}\exp\left[-\frac{1}{2}\operatorname{tr}\left(\Sigma_1^{-1}S_1^0\right)\right] \\
&\propto \text{Inv-Wishart}_{n_1+1}((S_1^0)^{-1})
\end{aligned}
$$

in which n_1 follows above, and

$$
S_1^0 = \sum_{i:\,C_i=0}(X_i-\mu_1)(X_i-\mu_1)^T
$$

Similarly,

$$
p(\Sigma_2|\cdot) \sim \text{Inv-Wishart}_{n_2+1}((S_2^0)^{-1})
$$

in which n_2 follows above, and

$$
S_1^0 = \sum_{i:\,C_i=1}(X_i-\mu_2)(X_i-\mu_2)^T
$$

For C_is,

$$
\begin{aligned}
p(C_i|\cdot) &\propto \left(\prod_i \Big[(1-C_i)N(X_i|\mu_1,\Sigma_1) + C_i N(X_i|\mu_2,\Sigma_2)\Big]\right)\cdot\left(\prod_i p(C_i|p_2)\right) \\
&\propto \Big[(1-C_i)N(X_i|\mu_1,\Sigma_1) + C_i N(X_i|\mu_2,\Sigma_2)\Big]p_2^{C_i}(1-p_2)^{1-C_i}
\end{aligned}
$$

in which, we use $N(X_i,|\cdot,\cdot)$ to denote the density function of multivariate normal distribution.

Overall, we have

$$
\begin{aligned}
&p(p_2|\cdot) \sim \text{Beta}(n_2+1, n_1+1) \\
&p(\mu_1|\cdot) \sim N(\bar{X}_1, \frac{\Sigma_1}{n_1}) \\
&p(\mu_2|\cdot) \sim N(\bar{X}_2, \frac{\Sigma_2}{n_2}) \\
&p(\Sigma_1|\cdot) \sim \text{Inv-Wishart}_{n_1+1}((S_1^0)^{-1}) \\
&p(\Sigma_2|\cdot) \sim \text{Inv-Wishart}_{n_2+1}((S_2^0)^{-1}) \\
&p(C_i|\cdot) \propto \Big[(1-C_i)N(X_i|\mu_1,\Sigma_1) + C_i N(X_i|\mu_2,\Sigma_2)\Big]p_2^{C_i}(1-p_2)^{1-C_i}
\end{aligned}
$$

(c)

We draw samples for $(p_2,\mu_1,\Sigma_1,\mu_2,\Sigma_2,C_1,\cdots,C_n)$. As for $p_1,Z_1,\cdots,Z_n$, just use

$$
p_1 = 1-p_2,\ \ Z_i = C_i + 1
$$

```
library(MCMCpack)
```

```
## Loading required package: coda
## Loading required package: MASS
```

```
## ##
## ## Markov Chain Monte Carlo Package (MCMCpack)
## ## Copyright (C) 2003-2023 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)
## ##
```

```r
library(mvtnorm)
set.seed(102932)
mixgauss <- read.table("./mixgauss.dat", header=FALSE)
n <- nrow(mixgauss)
p <- 3

#we need initial value for  p_2, \mu_1, \mu_2, \Sigma_1, \Sigma_2, C_1,...,C_n
p2.0 <- 0.5
Ci.0 <- sample(0:1, n, replace=TRUE) # 0: class 1, 1: class 2
mu1.0 <- colMeans(mixgauss[Ci.0==0, ])
mu2.0 <- colMeans(mixgauss[Ci.0==1, ])
Sigma1.0 <- cov(mixgauss[Ci.0==0, ])
Sigma2.0 <- cov(mixgauss[Ci.0==1, ])

# we sample in the order p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2, C_1,...,C_n

#' Gibbs sampler for p_2, \mu_1, \Sigma_1, \mu_2, \Sigma_2, C_1,...,C_n
#' @param ...  sample of step t-1
#' @return sample of step t
Gibbs <- function(p2, mu1, mu2, Sigma1, Sigma2, Ci){
  n2 <- sum(Ci)
  n1 <- n - n2

  #sample p
  p2.t <- rbeta(1, n2+1, n1+1)
  #sample mu
  mu1.t <- c(mvtnorm::rmvnorm(1,
                         colMeans(mixgauss[Ci==0, ]),
                         Sigma1/n1))
  mu2.t <- c(mvtnorm::rmvnorm(1,
                         colMeans(mixgauss[Ci==1, ]),
                         Sigma2/n2))
  #sample Sigma
  X1 <- t(mixgauss[Ci==0, ])
  X2 <- t(mixgauss[Ci==1, ])
  S1 <- tcrossprod(X1 - mu1.t)
  S2 <- tcrossprod(X2 - mu2.t)
  Sigma1.t <- MCMCpack::riwish(n1+1, S1)
  Sigma2.t <- MCMCpack::riwish(n2+1, S2)
  #unnormalized prob
  pCi.0 <- dmvnorm(mixgauss, mean=mu1.t, sigma=Sigma1.t) * (1-p2.t)
  pCi.1 <- dmvnorm(mixgauss, mean=mu2.t, sigma=Sigma2.t) * p2.t

  Ci.t <- mapply(function(p0, p1) sample(0:1, 1, prob=c(p0, p1)), pCi.0, pCi.1)
```

```r
  return(list(p2=p2.t, mu1=mu1.t, mu2=mu2.t,
              Sigma1=Sigma1.t, Sigma2=Sigma2.t, Ci=Ci.t))
}

#sampling
samp.size <- 10000
samples.out <- vector("list", samp.size)
samples.out[[1]] <- list(p2=p2.0, mu1=mu1.0, mu2=mu2.0,
                        Sigma1=Sigma1.0, Sigma2=Sigma2.0, Ci=Ci.0)
for (i in 2:(samp.size+1)){
  samples.out[[i]] <- do.call(Gibbs, args=samples.out[[i-1]])
}

#To make the final result of cluster labels Z_1...Z_n, and p_1, p_2, just
samples.out <- lapply(samples.out, function(sublist) {
  sublist$p1 <- 1-sublist$p2
  sublist$Zi <- sublist$Ci+1
  return(sublist)
})

#one example
samples.out[[samp.size+1]]
```

```
## $p2
## [1] 0.3218191
##
## $mu1
## [1] 0.6015645 2.5807451 2.0396036
##
## $mu2
## [1] -2.107712 -1.786528 -4.037533
##
## $Sigma1
##            V1         V2        V3
## V1 15.4093627 14.7806061 0.8651437
## V2 14.7806061 14.8324454 0.5200656
## V3  0.8651437  0.5200656 0.6063008
##
## $Sigma2
##            V1         V2         V3
## V1  1.1577500  0.4834668 -0.2417130
## V2  0.4834668  1.0503805 -0.1975337
## V3 -0.2417130 -0.1975337  0.9115112
##
## $Ci
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
##   1   0   1   1   0   0   0   0   0   0   0   0   0   1   0   0   1   0   1   0
##  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40
##   0   0   0   0   0   1   1   1   0   0   1   0   1   0   1   1   1   1   0   0
##  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60
##   0   0   0   0   0   0   0   0   1   0   0   1   0   0   0   1   0   0   1   0
##  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80
##   1   0   0   0   0   0   1   0   1   0   0   0   0   0   0   0   1   0   0   0
##  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100
```

```
##   1   0   1   1   0   1   0   1   1   0   1   1   0   0   1   1   0   1   1   0
## 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120
##   1   0   0   0   0   1   0   1   0   0   0   1   0   0   1   0   0   0   1   0
## 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140
##   0   0   0   0   0   1   0   0   1   0   1   1   0   0   1   0   1   0   0   0
## 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160
##   0   0   1   0   0   1   1   0   0   0   0   0   0   0   0   1   1   0   0   0
## 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
##   1   1   0   0   0   1   1   0   0   0   0   1   0   0   0   0   1   0   1   0
## 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200
##   0   0   0   1   0   0   1   0   0   0   1   0   0   1   1   0   0   0   0   0
## 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220
##   1   1   0   1   0   0   0   1   0   0   1   0   0   0   1   0   1   1   0   1
## 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240
##   0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   0   0   0   1   1
## 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260
##   1   0   1   1   0   0   0   0   1   1   0   0   0   0   0   0   0   1   0   0
## 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280
##   0   0   1   0   0   0   0   0   0   0   0   1   1   0   1   0   0   0   0   1
## 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300
##   0   1   0   0   0   0   1   1   1   0   0   0   0   0   0   0   0   0   0   1
##
## $p1
## [1] 0.6781809
##
## $Zi
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
##   2   1   2   2   1   1   1   1   1   1   1   1   1   2   1   1   2   1   2   1
##  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40
##   1   1   1   1   1   2   2   2   1   1   2   1   2   1   2   2   2   2   1   1
##  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60
##   1   1   1   1   1   1   1   1   2   1   1   2   1   1   1   2   1   1   2   1
##  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80
##   2   1   1   1   1   1   2   1   2   1   1   1   1   1   1   1   2   1   1   1
##  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100
##   2   1   2   2   1   2   1   2   2   1   2   2   1   1   2   2   1   2   2   1
## 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120
##   2   1   1   1   1   2   1   2   1   1   1   2   1   1   2   1   1   1   2   1
## 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140
##   1   1   1   1   1   2   1   1   2   1   2   2   1   1   2   1   2   1   1   1
## 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160
##   1   1   2   1   1   2   2   1   1   1   1   1   1   1   1   2   2   1   1   1
## 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
##   2   2   1   1   1   2   2   1   1   1   1   2   1   1   1   1   2   1   2   1
## 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200
##   1   1   1   2   1   1   2   1   1   1   2   1   1   2   2   1   1   1   1   1
## 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220
##   2   2   1   2   1   1   1   2   1   1   2   1   1   1   2   1   2   2   1   2
## 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240
##   1   1   1   1   1   1   1   1   2   1   1   1   1   1   1   1   1   1   2   2
## 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260
##   2   1   2   2   1   1   1   1   2   2   1   1   1   1   1   1   1   2   1   1
## 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280
##   1   1   2   1   1   1   1   1   1   1   1   2   2   1   2   1   1   1   1   2
```

```
## 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300
##   1   2   1   1   1   1   2   2   2   1   1   1   1   1   1   1   1   1   1   2
```

(d)

```
mu1.sample <- do.call(rbind, lapply(samples.out, function(p) p$mu1))
mu2.sample <- do.call(rbind, lapply(samples.out, function(p) p$mu2))
Zi.sample <- do.call(rbind, lapply(samples.out, function(p) p$Zi))
Sigma1.sample <- do.call(rbind, lapply(samples.out, function(p) c(p$Sigma1)))
Sigma2.sample <- do.call(rbind, lapply(samples.out, function(p) c(p$Sigma2)))

#point estimate
#use last 5000 sample
mu1.hat <- colMeans(mu1.sample[5000:samp.size+1, ]); mu1.hat
```

```
##        V1        V2        V3
## 0.6884259 2.7566725 1.9692044
```

```
mu2.hat <- colMeans(mu2.sample[5000:samp.size+1, ]); mu2.hat
```

```
##        V1        V2        V3
## -1.962445 -1.882698 -4.161210
```

```
Zi.hat <- colMeans(Zi.sample[5000:samp.size+1, ])
Sigma1.hat <- matrix(colMeans(Sigma1.sample[5000:samp.size+1, ]), ncol=3); Sigma1.hat
```

```
##            [,1]       [,2]      [,3]
## [1,] 16.5095971 16.1041852 0.5489031
## [2,] 16.1041852 16.4013590 0.2295081
## [3,]  0.5489031  0.2295081 0.5678515
```

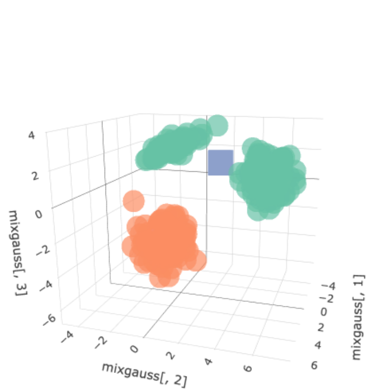
```
Sigma2.hat <- matrix(colMeans(Sigma2.sample[5000:samp.size+1, ]), ncol=3); Sigma2.hat
```

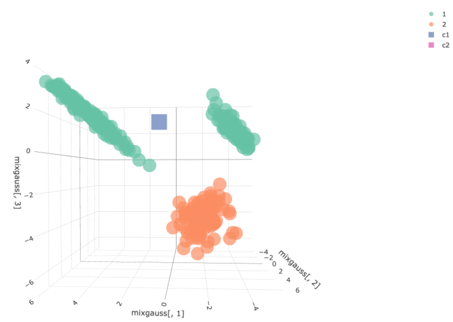
```
##            [,1]        [,2]        [,3]
## [1,]  0.8536147  0.19620152 -0.13165265
## [2,]  0.1962015  0.99272414 -0.01717723
## [3,] -0.1316526 -0.01717723  0.88331230
```

```
#especially, for class label, to visualize, if Zi>=1.5, then cluster 2, else cluster 1
Zi.hat.vi <- (Zi.hat>=1.5)+1; head(Zi.hat.vi)
```

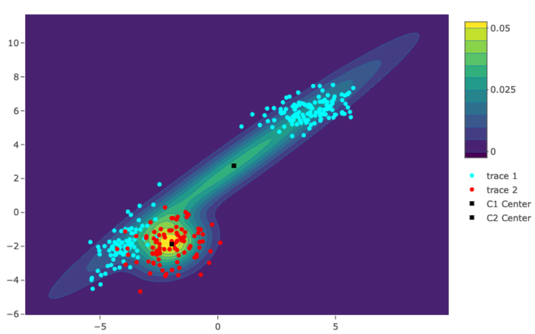
```
## 1 2 3 4 5 6
## 2 1 2 2 1 1
```

It's not easy to embed an interactive plotly object in a pdf file. So we use two ways to visualize. First, we take two snapshots of these points.

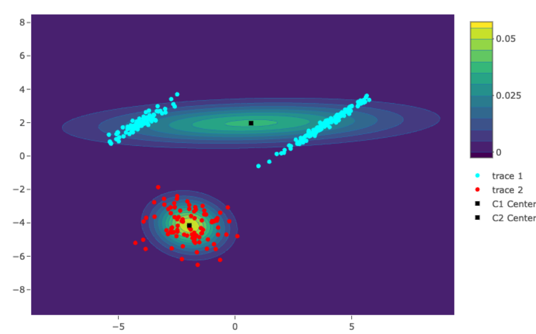




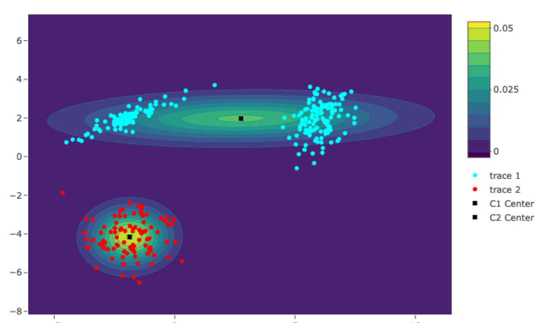
Second, we plot the projection of the data points and the contour of marginal distribution in three coordinate planes. To get the contour, we use the point estimate of p_2 to calculate the density.



(a) X-Y Plane



(b) X-Z Plane



(c) Y-Z Plane

We show the cluster centers by square symbol. However, from the plots, maybe it's not appropriate to assume that the data comes from 2 classes.

```r
library(plotly, warn.conflicts=FALSE)
# Create a 3D scatter plot with different colors for each class
scatter_plot <- plot_ly(data = mixgauss,
                        x = ~mixgauss[, 1], y = ~mixgauss[, 2], z = ~mixgauss[, 3],
                        type = "scatter3d", color = ~factor(Zi.hat.vi),
                        mode="markers",
                        opacity=0.7)

# Add the cluster center to the scatter plot
center <- data.frame(rbind(mu1.hat, mu2.hat))
colnames(center) <- c("x", "y", "z")
center$class <- c("c1", "c2")
final_plot <- scatter_plot %>%
  add_trace(data=center,
            x=~x,
            y=~y,
            z=~z,
            color=~class,
            mode="markers",
            type="scatter3d",
            opacity=1,
            marker = list(size = 10, symbol = "square"))

p2.hat <- mean(do.call(c, lapply(samples.out, function(p) p$p2)))

# x-y plane
x <- seq(-10, 12, length.out = 200)
y <- seq(-10, 12, length.out = 200)
grid <- expand.grid(x, y)

itr <- list(c(1,2,3), c(1,3,2), c(2,3,1))
out <- vector("list", 3)
for (i in 1:3){
  a <- itr[[i]][3]
  o1 <- itr[[i]][1]
  o2 <- itr[[i]][2]
# Calculate density values for both distributions
density1 <- dmvnorm(grid, mean = mu1.hat[-a], sigma = Sigma1.hat[-a, -a])
density2 <- dmvnorm(grid, mean = mu2.hat[-a], sigma = Sigma2.hat[-a, -a])
density <- (1-p2.hat)*density1+p2.hat*density2
contour_plot1 <- plot_ly(x = x, y = y,
                        z = matrix(density, ncol=200, byrow=TRUE),
                        type = "contour")
options(warn = -1)
p <- contour_plot1 %>% add_trace(x = mixgauss[Zi.hat.vi==1,o1],
                                y = mixgauss[Zi.hat.vi==1,o2],
                                type = "scatter", mode = "markers",
                                marker = list(color = "cyan") ) %>%
                  add_trace(x = mixgauss[Zi.hat.vi==2,o1],
                                y = mixgauss[Zi.hat.vi==2,o2],
                                type = "scatter", mode = "markers",
                                marker = list(color = "red")) %>%
                  add_trace(x = mu1.hat[o1],
```

```r
                          y = mu1.hat[o2],
                          type = "scatter", mode = "markers",
                          marker = list(color = "black",symbol="square"),
                          name = "C1 Center") %>%
                add_trace(x = mu2.hat[o1],
                          y = mu2.hat[o2],
                          type = "scatter", mode = "markers",
                          marker = list(color = "black", symbol="square"),
                          name = "C2 Center")
out[[i]] <- p
}
```

Problem 4

(a)

We consider the normal model of multiple observations. The likelihood is

$$\begin{aligned} p(\mathbf{y}|\theta,\sigma^2) &= \prod_i p(y_i|\theta,\sigma^2) \\ &= \prod_i \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y_i-\mu)^2}{2\sigma^2}\right] \end{aligned}$$

so

$$\begin{aligned} \log p(\mathbf{y}|\theta,\sigma^2) &= -\sum_i \log\sqrt{2\pi\sigma^2} + \frac{(y_i-\mu)^2}{2\sigma^2} \\ &= \text{const} - \frac{n}{2}\log\sigma^2 - \frac{(n-1)s_y^2 + n(\bar{y}-\mu)^2}{2\sigma^2} \end{aligned}$$

in which, $s_y^2 = \sum_i (y_i-\bar{y})^2/(n-1)$.
Therefore, we have

$$\begin{aligned} \frac{\partial \log p(\mathbf{y}|\theta,\sigma^2)}{\partial\mu} &= \frac{n(\bar{y}-\mu)}{\sigma^2} \\ \frac{\partial \log p(\mathbf{y}|\theta,\sigma^2)}{\partial\sigma^2} &= -\frac{n}{2\sigma^2} + \frac{(n-1)s_y^2 + n(\bar{y}-\mu)^2}{2(\sigma^2)^2} \end{aligned}$$

and then

$$\begin{aligned} \frac{\partial^2 \log p(\mathbf{y}|\theta,\sigma^2)}{\partial\mu^2} &= -\frac{n}{\sigma^2} \\ \frac{\partial^2 \log p(\mathbf{y}|\theta,\sigma^2)}{\partial\sigma^2} &= \frac{n}{2\sigma^4} - \frac{(n-1)s_y^2 + n(\bar{y}-\mu)^2}{\sigma^6} \\ \frac{\partial^2 \log p(\mathbf{y}|\theta,\sigma^2)}{\partial\mu\partial\sigma^2} &= \frac{\partial^2 \log p(\mathbf{y}|\theta,\sigma^2)}{\partial\sigma^2\partial\mu} = -\frac{n(\bar{y}-\mu)}{\sigma^4} \end{aligned}$$

Therefore, the Fisher Information is

$$\begin{aligned} I(\mu,\sigma^2) &= -\mathbb{E}\begin{bmatrix} -\frac{n}{\sigma^2} & -\frac{n(\bar{y}-\mu)}{\sigma^4} \\ -\frac{n(\bar{y}-\mu)}{\sigma^4} & \frac{n}{2\sigma^4} - \frac{(n-1)s_y^2+n(\bar{y}-\mu)^2}{\sigma^6} \end{bmatrix} \\ &= -\begin{bmatrix} -\frac{n}{\sigma^2} & 0 \\ 0 & \frac{n}{2\sigma^4} - \frac{(n-1)\sigma^2+\sigma^2}{\sigma^6} \end{bmatrix} \\ &= \begin{bmatrix} \frac{n}{\sigma^2} & 0 \\ 0 & \frac{n}{2\sigma^4} \end{bmatrix} \end{aligned}$$

in which, we use

$$\mathbb{E}(\bar{y}-\mu) = 0, \mathbb{E}(\bar{y}-\mu)^2 = \mathbb{E}s_y^2 = \sigma^2$$

Therefore, the Jeffreys' prior is

$$p_J(\mu,\sigma^2) \propto \sqrt{\frac{n^2}{2\sigma^6}} \propto (\sigma^2)^{-3/2}$$

(b)

The posterior distribution is

$$\begin{aligned}
p_J(\mu,\sigma^2|\mathbf{y}) &\propto p_J(\mu,\sigma^2)p(\mathbf{y}|\mu,\sigma^2) \\
&\propto (\sigma^2)^{-3/2}\prod_i \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left[-\frac{(y_i-\mu)^2}{2\sigma^2}\right] \\
&\propto \left(\sigma^2\right)^{-3/2}\cdot\left(\sigma^2\right)^{-n/2}\exp\left[-\frac{(n-1)s_y^2+n(\bar{y}-\mu)^2}{2\sigma^2}\right] \\
&\propto \sigma^{-1}\cdot(\sigma^2)^{-(n/2+1)}\exp\left[-\frac{1}{2\sigma^2}\left(n\cdot\frac{\sum_i(y_i-\bar{y})^2}{n}+n(\bar{y}-\mu)^2\right)\right]
\end{aligned}$$

It's known that this term follows a Normal-Inverse-χ^2 distribution, formally (using the notation in the book Bayesian Data Analysis Third edition),

$$p_J(\mu,\sigma^2|\mathbf{y}) \sim \text{N-Inv-}\chi^2\left(\bar{y},\frac{\sum_i(y_i-\bar{y})^2}{n^2};n,\frac{\sum_i(y_i-\bar{y})^2}{n}\right)$$

To see more clearly, we can rewrite

$$\begin{aligned}
p_J(\mu|\sigma^2,\mathbf{y}) &\propto \sigma^{-1}\exp\left[-\frac{1}{2\sigma^2/n}\left(\mu-\bar{y}\right)^2\right] \sim N\left(\bar{y},\frac{\sigma^2}{\kappa_n}\right), \kappa_n = n \\
p_J(\sigma^2|\mathbf{y}) &\propto (\sigma^2)^{-(n/2+1)}\exp\left[-\frac{1}{2\sigma^2}\left(n\cdot\frac{\sum_i(y_i-\bar{y})^2}{n}\right)\right] \\
&\propto (\sigma^2)^{-(\nu_n/2+1)}\exp\left[-\frac{1}{2\sigma^2}\left(\nu_n\cdot\sigma_n^2\right)\right] \\
&\sim \text{Inv-}\chi^2(\nu_n,\sigma_n^2), \quad \nu_n = n, \sigma_n^2 = \frac{\sum_i(y_i-\bar{y})^2}{n} \\
&\sim \text{Inv-Gamma}\left(\frac{n}{2},\frac{\sum_i(y_i-\bar{y})^2}{2}\right)
\end{aligned}$$

Hence, this joint density $p_J(\mu,\sigma^2|\mathbf{y})$ can be considered a proper posterior density, which is

$$\text{N-Inv-}\chi^2\left(\bar{y},\frac{\sum_i(y_i-\bar{y})^2}{n^2};n,\frac{\sum_i(y_i-\bar{y})^2}{n}\right)$$

(c)

We can rewrite the prior of (θ,Σ) to

$$p_J(\theta,\Sigma) = C_1|\Sigma|^{-(p+2)/2}$$

in which C_1 is a constant.
Now, we assume that $p_J(\theta,\Sigma)$ is proper, that is

$$\int p_J(\theta,\Sigma)d\theta d\Sigma = \int C_1|\Sigma|^{-(p+2)/2}d\theta d\Sigma = 1$$

By Fubini's Theorem, as well as this post, the marginal distribution of Σ should also be proper (a.s.). However, if we calculate the marginal distribution directly, note that the support of *theta* is $\mathbb{R}^p$

$$\begin{aligned}
p(\Sigma) = \int_{\mathbb{R}^p} p_J(\theta,\Sigma)d\theta &= C_1|\Sigma|^{-(p+2)/2}\int_{\mathbb{R}^p} 1\cdot d\theta \\
&= C_1|\Sigma|^{-(p+2)/2}\cdot\infty \\
&= \infty
\end{aligned}$$

Obviously, the integral above is divergent, which means $p(\Sigma)$ is not proper. By contradiction, $p_J(\theta, \Sigma)$ must be improper. So it cannot actually be a probability density for (θ, Σ).

(d)

Just do it.
Prior

$$p_J(\theta, \Sigma) \propto |\Sigma|^{-(p+2)/2}$$

Likelihood

$$\begin{aligned} p(\mathbf{y}_1, \cdots, \mathbf{y}_n|\theta, \Sigma) &\propto |\Sigma|^{-n/2} \exp\left[-\frac{1}{2}\sum_{i=1}^{n} (y_i - \theta)^T \Sigma^{-1} (y_i - \theta)\right] \quad y \in \mathbb{R}^p \\ &\propto |\Sigma|^{-n/2} \exp\left[-\frac{1}{2} \operatorname{tr}\left(\Sigma^{-1} S_0\right)\right], \quad S_0 = \sum_{i=1}^{n} (y_i - \theta)(y_i - \theta)^T \end{aligned}$$

Posterior

$$\begin{aligned} p_J(\theta, \Sigma|\mathbf{y}_1, \cdots, \mathbf{y}_n) &\propto p_J(\theta, \Sigma) p(\mathbf{y}_1, \cdots, \mathbf{y}_n|\theta, \Sigma) \\ &\propto |\Sigma|^{-(n+p+2)/2} \exp\left[-\frac{1}{2} \operatorname{tr}\left(\Sigma^{-1} S_0\right)\right] \end{aligned}$$

Note that

$$\begin{aligned} \operatorname{tr}\left(\Sigma^{-1} S_0\right) &= \sum_{i=1}^{n} (y_i - \theta)^T \Sigma^{-1} (y_i - \theta) \\ &= \sum_{i=1}^{n} (y_i - \bar{y} + \bar{y} - \theta)^T \Sigma^{-1} (y_i - \bar{y} + \bar{y} - \theta) \\ &= \sum_{i=1}^{n} (y_i - \bar{y})^T \Sigma^{-1} (y_i - \bar{y}) - 2\sum_{i=1}^{n} (y_i - \bar{y})^T \Sigma^{-1}(\bar{y} - \theta) + n(\bar{y} - \theta)^T \Sigma^{-1}(\bar{y} - \theta) \\ &= \sum_{i=1}^{n} (y_i - \bar{y})^T \Sigma^{-1} (y_i - \bar{y}) + n(\theta - \bar{y})^T \Sigma^{-1}(\theta - \bar{y}) \\ &= \operatorname{tr}\left(\Sigma^{-1} S\right) + n(\theta - \bar{y})^T \Sigma^{-1}(\theta - \bar{y}), \quad S = \sum_{i=1}^{n} (y_i - \bar{y})(y_i - \bar{y})^T \end{aligned}$$

therefore

$$p_J(\theta, \Sigma|\mathbf{y}_1, \cdots, \mathbf{y}_n) \propto |\Sigma|^{-((n+p)/2+1)} \exp\left[-\frac{1}{2}\left(\operatorname{tr}\left(\Sigma^{-1} S\right) + n(\mu - \bar{y})^T \Sigma^{-1}(\mu - \bar{y})\right)\right]$$

It's known that this term follows a Normal-Inverse-Wishart distribution, formally (using the notation in the book Bayesian Data Analysis Third edition),

$$p_J(\theta, \Sigma|\mathbf{y}_1, \cdots, \mathbf{y}_n) \sim \text{Normal-Inverse-Wishart}\left(\bar{y}, \frac{S}{n}; n, S\right)$$

And we can get

$$p_J(\theta|\Sigma, \mathbf{y}_1, \cdots, \mathbf{y}_n) \propto |\Sigma/n|^{-1/2} \exp\left[-\frac{1}{2}(\theta - \bar{y})^T \left(\frac{\Sigma}{n}\right)^{-1} (\theta - \bar{y})\right] \sim N\left(\bar{y}, \frac{\Sigma}{n}\right)$$

$$p_J(\Sigma|\mathbf{y}_1, \cdots, \mathbf{y}_n) \propto |\Sigma|^{-((n+p+1)/2)} \exp\left[-\frac{1}{2} \operatorname{tr}\left(\Sigma^{-1} S\right)\right] \sim \text{Inv-Wishart}_n(S^{-1})$$