# MAST30027 Modern Applied Statistics Assignment 4

Tutorial:Wed 1-2PM, Yidi Deng

James La Fontaine

2023-10-24

## Question 1

### (a)

1.

$$x_i \sim N\left(75, \frac{1}{\tau}\right) \quad \text{for } i = 1, \ldots, 100$$

$$\Rightarrow f(x_i \mid \tau) \propto \tau^{1/2} e^{-\frac{\tau}{2}(x_i - 75)^2}$$

$$\tau \sim \text{Gamma}(2, 1)$$

$$\Rightarrow f(\tau) = \frac{1}{\Gamma(2)} \tau e^{-\tau} = \tau e^{-\tau}$$

a) $$P(\tau \mid X_1, \ldots, X_n) \propto P(X_1, \ldots, X_n \mid \tau) P(\tau)$$

$$\propto \tau e^{-\tau} \prod_{i=1}^{100} \tau^{1/2} e^{-\frac{\tau}{2}(x_i - 75)^2}$$

$$\propto \tau^{51} e^{-\tau} e^{\frac{-\tau}{2} \sum_{i=1}^{100} (x_i - 75)^2}$$

$$\propto \tau^{51} e^{-\tau\left[1 + \frac{\sum_{i=1}^{100} (x_i - 75)^2}{2}\right]}$$

```
> (posterior_rate = 1 + (sum((X-75)^2)) / 2)
>
> 1805.65
```

$\implies (\tau \mid X_1, ..., X_n) \sim \text{Gamma}(52, 1805.65)$

(b)

Let $\beta = 1 + \frac{\sum_{i=1}^{100} (x_i - 75)^2}{2}$

b) $P(\tilde{x} \mid x) = \int P(\tilde{x} \mid \tau) P(\tau \mid x) \, d\tau$

$$= \int \frac{\tau^{1/2}}{\sqrt{2\pi}} e^{-\tau/2 (\tilde{x}-75)^2} \frac{\beta^{52}}{\Gamma(52)} \tau^{51} e^{-\tau\beta} \, d\tau$$

$$= \frac{\beta^{52}}{\sqrt{2\pi}\,\Gamma(52)} \int \tau^{\frac{103}{2}} e^{-\frac{\tau}{2}(\tilde{x}-75)^2 - \tau\beta} \, d\tau$$

$$= \frac{\Gamma\left(\frac{105}{2}\right)\beta^{52}}{\sqrt{2\pi}\,\Gamma(52)} \int \tau^{\frac{105}{2}-1} e^{-\tau\left(\frac{(\tilde{x}-75)^2}{2}+\beta\right)} \frac{1}{\Gamma\left(\frac{105}{2}\right)} \, d\tau$$

$$= \frac{\Gamma\left(\frac{105}{2}\right)\beta^{52}}{\sqrt{2\pi}\,\Gamma(52)\left(\frac{(\tilde{x}-75)^2}{2}+\beta\right)^{\frac{105}{2}}} \int \frac{\left(\frac{(\tilde{x}-75)^2}{2}+\beta\right)^{\frac{105}{2}}}{\Gamma\left(\frac{105}{2}\right)} \tau^{\frac{105}{2}-1} e^{-\tau\left(\frac{(\tilde{x}-75)^2}{2}+\beta\right)} \, d\tau$$

$$= \frac{\Gamma\left(\frac{105}{2}\right)\beta^{52}}{\sqrt{2\pi}\,\Gamma(52)\left(\frac{(\tilde{x}-75)^2}{2}+\beta\right)^{\frac{105}{2}}}$$ as the integral of the Gamma pdf $= 1$

$$= \frac{\Gamma\left(\frac{105}{2}\right)\beta^{52}}{\Gamma\left(\frac{104}{2}\right)\sqrt{2\pi}} \left(\frac{(\tilde{x}-75)^2}{2}+\beta\right)^{-\frac{105}{2}}$$

$$= \frac{\Gamma\left(\frac{105}{2}\right)\beta^{52}}{\Gamma\left(\frac{104}{2}\right)\sqrt{2\pi}} \left(\beta\left(\frac{(\tilde{x}-75)^2}{2\beta}+1\right)\right)^{-\frac{105}{2}}$$

$$= \frac{\Gamma\left(\frac{105}{2}\right)\beta^{52}}{\Gamma\left(\frac{104}{2}\right)\sqrt{2\pi}} \beta^{-\frac{105}{2}} \left(1+\frac{1}{104}\frac{104(\tilde{x}-75)^2}{\beta}\right)^{-\frac{105}{2}}$$

$$= \frac{\Gamma\left(\frac{105}{2}\right)}{\Gamma\left(\frac{104}{2}\right)\sqrt{2\pi\beta}} \left(1+\frac{1}{104}\frac{52(\tilde{x}-75)^2}{\beta}\right)^{-\frac{105}{2}}$$

$$= \frac{\Gamma\left(\frac{105}{2}\right)}{\Gamma\left(\frac{104}{2}\right)\sqrt{104\pi\frac{\beta}{52}}} \left(1+\frac{1}{104}\frac{52(\tilde{x}-75)^2}{\beta}\right)^{-\frac{105}{2}}$$

$\Rightarrow (\tilde{x} \mid x) \sim t(\nu, a, b)$ where $\nu = 104$,
$a = 75$,
$b = \frac{\beta}{52}$

from (a)
$b = \frac{1805.65}{52}$
$= 34.724$

## Question 2

### (a)

2. $x_1, \dots, x_{100}$ and $y_1, \dots, y_{150}$ i.i.d

$x_i \sim N(\mu_1, 1^2)$

$y_i \sim N\left(\mu_2, \left(\frac{1}{\sqrt{2}}\right)^2\right)$

$\begin{pmatrix}\mu_1 \\ \mu_2\end{pmatrix} \sim N(\vec{\mu}, \vec{\Sigma})$ with $\vec{\mu} = \begin{pmatrix}0 \\ 0\end{pmatrix}$ and $\vec{\Sigma} = \begin{pmatrix}\frac{3}{5} & -\frac{2}{5} \\ -\frac{2}{5} & \frac{3}{5}\end{pmatrix}$

$f_{\vec{\mu},\vec{\Sigma}}(\vec{x}) = \frac{1}{2\pi|\vec{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\vec{x}-\vec{\mu})^T \Sigma^{-1}(\vec{x}-\vec{\mu})\right)$

a) Derive $p(\mu_1 | \mu_2, x_1, \dots, x_{100}, y_1, \dots, y_{150})$

and $p(\mu_2 | \mu_1, x_1, \dots, x_{100}, y_1, \dots, y_{150})$

$$\Sigma^{-1} = \begin{pmatrix}3 & 2 \\ 2 & 3\end{pmatrix}$$

$$p(\mu_1, \mu_2) \propto e^{\left(-\frac{1}{2}(\mu_1 \; \mu_2)\begin{pmatrix}3 & 2 \\ 2 & 3\end{pmatrix}\begin{pmatrix}\mu_1 \\ \mu_2\end{pmatrix}\right)}$$
$$= e^{\left(-\frac{1}{2}(3\mu_1 + 2\mu_2, 2\mu_1 + 3\mu_2)\begin{pmatrix}\mu_1 \\ \mu_2\end{pmatrix}\right)}$$
$$= e^{\left(-\frac{1}{2}(3\mu_1^2 + 4\mu_1\mu_2 + 3\mu_2^2)\right)}$$

$$p(\mu_1 | \mu_2, X, Y) \propto P(X, Y, \mu_1, \mu_2) \text{ [drop terms without } \mu_1]$$
$$\propto e^{-\frac{1}{2}\left(\sum_{i=1}^{100}(x_i - \mu_1)^2 + 3\mu_1^2 + 4\mu_1\mu_2\right)}$$
$$\propto e^{-\frac{1}{2}\left(\sum_{i=1}^{100}(x_i^2 - 2x_i\mu_1 + \mu_1^2) + 3\mu_1^2 + 4\mu_1\mu_2\right)}$$
$$\propto e^{-\frac{1}{2}\left(100\mu_1^2 - 2\mu_1\sum_{i=1}^{100}x_i + 3\mu_1^2 + 4\mu_1\mu_2\right)}$$
$$\propto e^{-\frac{1}{2}\left(103\mu_1^2 - 2\mu_1\left(\sum_{i=1}^{100}x_i - 2\mu_2\right)\right)}$$
$$\propto e^{-\frac{103}{2}\left(\mu_1^2 - 2\mu_1\left(\frac{\sum_{i=1}^{100}x_i - 2\mu_2}{103}\right)\right)}$$
$$\propto e^{-\frac{1}{2}\frac{\left(\mu_1 - \frac{\sum_{i=1}^{100}x_i - 2\mu_2}{103}\right)^2}{\frac{1}{103}}}$$
$$\Rightarrow (\mu_1 | \mu_2, X, Y) \sim N\left(\frac{\sum_{i=1}^{100}x_i - 2\mu_2}{103}, \frac{1}{103}\right)$$

$$p(\mu_2 | \mu_1, X, Y) \propto P(X, Y, \mu_1, \mu_2) \text{ [drop terms without } \mu_2]$$
$$\propto e^{-\left(\sum_{i=1}^{150}(y_i - \mu_2)^2 + 2\mu_1\mu_2 + \frac{3}{2}\mu_2^2\right)}$$
$$\propto e^{-\left(\sum_{i=1}^{150}(y_i^2 - 2y_i\mu_2 + \mu_2^2) + \frac{3}{2}\mu_2^2 + 2\mu_1\mu_2\right)}$$
$$\propto e^{-\left(150\mu_2^2 - 2\mu_2\sum_{i=1}^{150}y_i + \frac{3}{2}\mu_2^2 + 2\mu_1\mu_2\right)}$$
$$\propto e^{-\left(\frac{303}{2}\mu_2^2 - 2\mu_2\left(\sum_{i=1}^{150}y_i - \mu_1\right)\right)}$$
$$\propto e^{-\frac{303}{2}\left(\mu_2^2 - 4\mu_2\left(\frac{\sum_{i=1}^{150}y_i - \mu_1}{303}\right)\right)}$$
$$\propto e^{\frac{-\frac{1}{2}\left(\mu_2 - 2\frac{\sum_{i=1}^{150}y_i - \mu_1}{303}\right)^2}{\frac{1}{303}}}$$
$$\Rightarrow (\mu_2 | \mu_1, X, Y) \sim N\left(2\frac{\sum_{i=1}^{150}y_i - \mu_1}{303}, \frac{1}{303}\right)$$

```r
x = scan(file="assignment4_prob2_x_2023.txt", what=double())

y = scan(file="assignment4_prob2_y_2023.txt", what=double())

length(x)
```

```
## [1] 100
```

```r
length(y)
```

```
## [1] 150
```

```r
mean(x)
```

```
## [1] 3.196441
```

```r
mean(y)
```

```
## [1] -1.979781
```

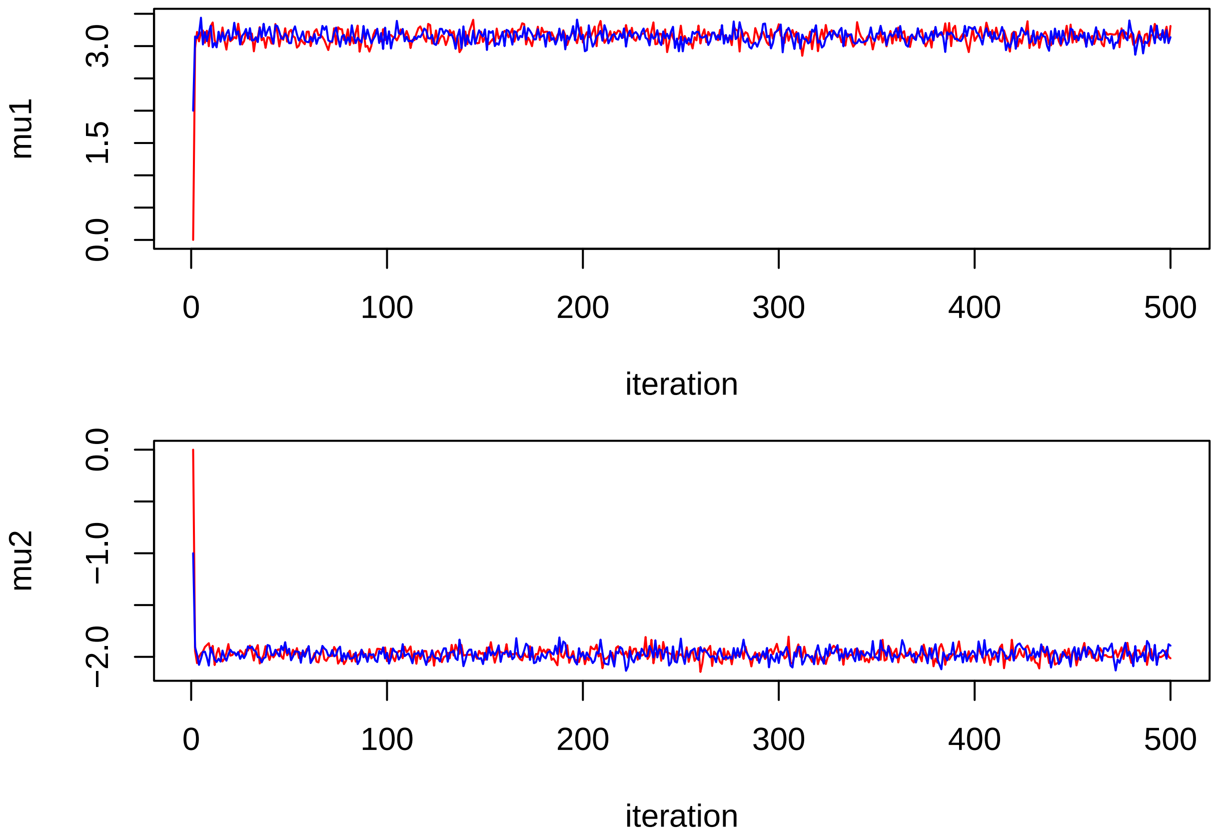
## 2b

```r
GibbsS <- function(mu1, mu2, nreps){

  Gsamples <- matrix(nrow=nreps, ncol=2)
  Gsamples[1,] <- c(mu1, mu2)

  # main loop
  for (i in 2:nreps) {
    mu1 <- rnorm(1, (sum(x) - 2*mu2)/103, sqrt(1/103))
    mu2 <- rnorm(1, 2*(sum(y) - mu1)/303, sqrt(1/303))
    Gsamples[i,] <- c(mu1, mu2)
  }

  return(Gsamples=Gsamples)
}
```

```r
set.seed(456)

# sample size
nreps <- 500

# initial values
mu1 <- 0
mu2 <- 0

GibbsS1 = GibbsS(mu1, mu2, nreps)
```

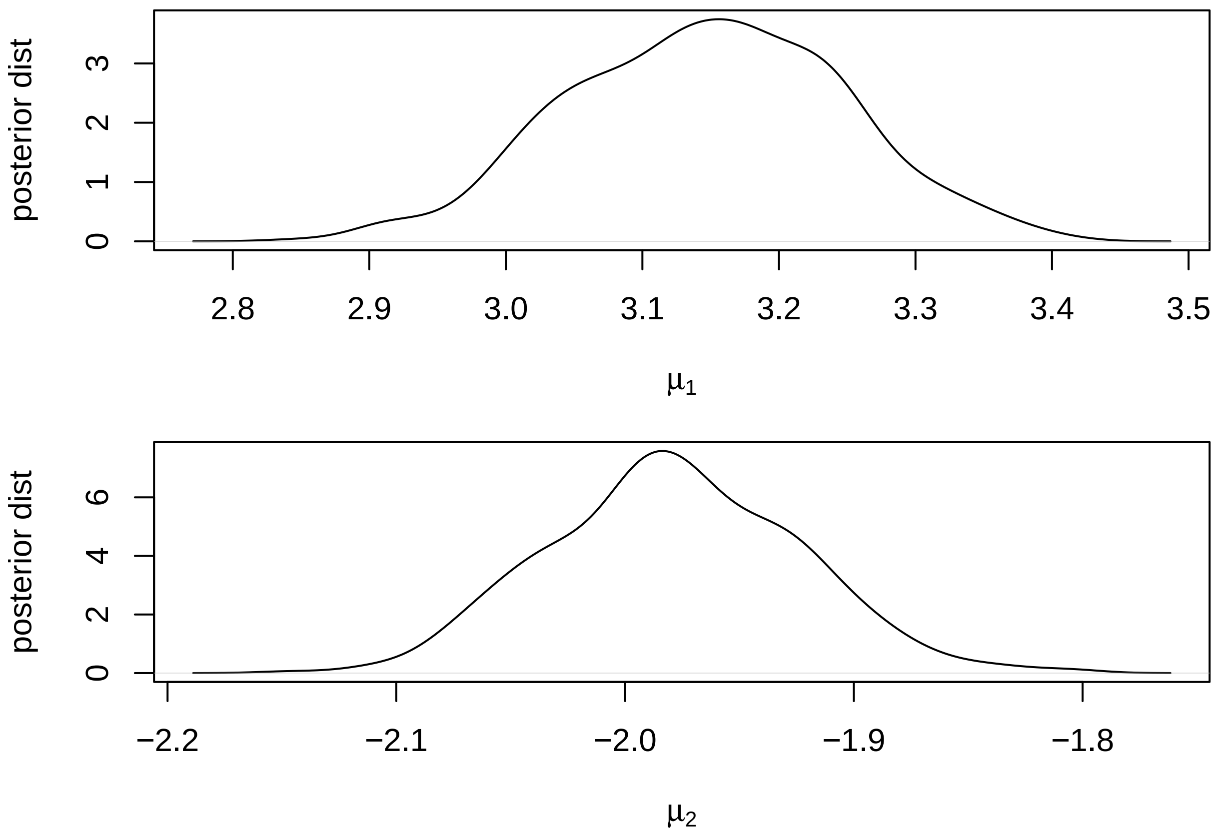
```r
# initial values
mu1 <- 2
mu2 <- -1

GibbsS2 = GibbsS(mu1, mu2, nreps)
```

```r
par(mfrow=c(2,1), mar=c(4,4,1,1))
plot(1:nreps, GibbsS1[,1], type="l", col="red", ylim = c(min(GibbsS1[,1],GibbsS2[,1]), max(GibbsS1[,1],G
     xlab = "iteration", ylab ="mu1")
points(1:nreps, GibbsS2[,1], type="l", col="blue")

plot(1:nreps, GibbsS1[,2], type="l", col="red", ylim = c(min(GibbsS1[,2],GibbsS2[,2]), max(GibbsS1[,2],G
     xlab = "iteration", ylab ="mu2")
points(1:nreps, GibbsS2[,2], type="l", col="blue")
```



**2c**

```
par(mfrow=c(2,1), mar=c(4,4,1,1))
plot(density(GibbsS1[-(1:50),1]), ylab="posterior dist", xlab=expression(mu[1]), main="")
plot(density(GibbsS1[-(1:50),2]), ylab="posterior dist", xlab=expression(mu[2]), main="")
```




```
mean(GibbsS1[-(1:50),1])
```

```
## [1] 3.146983
```

```
mean(GibbsS1[-(1:50),2])
```

```
## [1] -1.978406
```

```
quantile(GibbsS1[-(1:50),1], probs= c(0.05, 0.95))
```

```
##       5%      95%
## 2.992497 3.313051
```

```
quantile(GibbsS1[-(1:50),2], probs= c(0.05, 0.95))
```

```
##        5%       95%
## -2.065780 -1.891366
```

Estimated marginal posterior mean for $\mu_1$ = 3.1470

Estimated marginal posterior mean for $\mu_2$ = -1.9784

90% credible interval for $\mu_1$ = (2.9925, 3.3131)

90% credible interval for $\mu_2$ = (-2.0658, -1.8914)

### 2d

```r
dbinorm <- function(x, mu, Si) {
    # x and mu are vectors length 2 and Si a 2x2 matric
    # returns the density at x of a bivariate normal (mean mu, var Si)
    exp(-t(x - mu)%*%solve(Si, x - mu)/2)/2/pi/sqrt(det(Si))
}


######## MH algorithm ################
run_metropolis_MCMC <- function(startvalue, iterations){
    chain = array(dim = c(iterations+1,2))
    chain[1,] = startvalue
    for (i in 1:iterations){
        proposal = proposalfunction(chain[i,])

        probab = exp(posterior(proposal) - posterior(chain[i,]))
        if (runif(1) < probab){
            chain[i+1,] = proposal
        }else{
            chain[i+1,] = chain[i,]
        }
    }
    return(chain)
}

# propose new parameter values
proposalfunction <- function(param){
    return(rnorm(2,mean = param, sd= c(0.1,0.1)))
}

# evaluate log posterior at given parameter values
posterior <- function(param){
   return (likelihood(param) + prior(param))
}

# evaluate log prior at given parameter values
prior <- function(param){
    mu_vec = c(0, 0)
    sigma_matrix = matrix(c(3/5, -2/5, -2/5, 3/5), ncol=2)
    return(log(dbinorm(param, mu_vec, sigma_matrix)))
}

# evaluate log likelihood at given parameter values
likelihood <- function(param){
    mu1 = param[1]
    mu2 = param[2]
    ll_x = sum(dnorm(x, mean=mu1, sd=1, log = TRUE))
    ll_y = sum(dnorm(y, mean=mu2, sd=1/sqrt(2), log = TRUE))
    return(ll_x + ll_y)
}
```

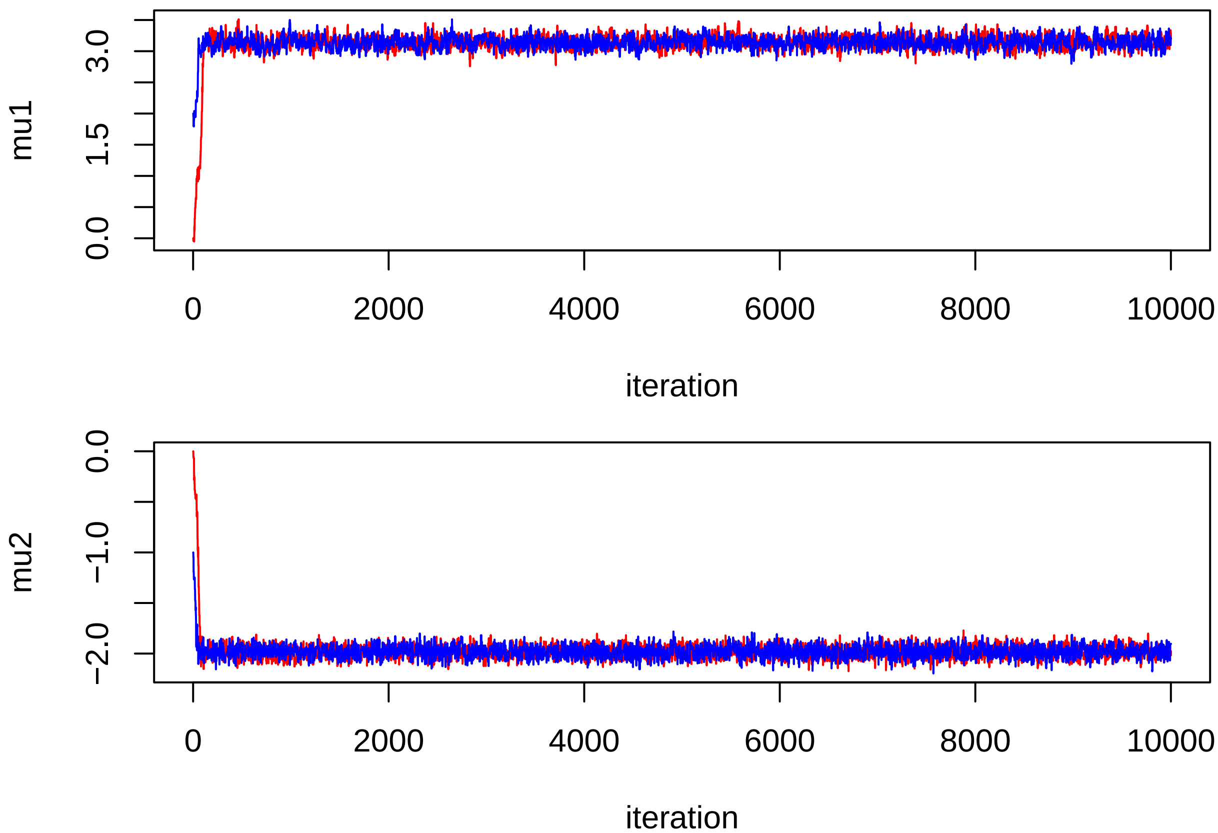
```r
set.seed(456)

nreps = 10000

# initial values
startvalue = c(0,0)
# simulate 10000 samples
MHchain1 = run_metropolis_MCMC(startvalue, nreps)

# initial values
startvalue = c(2,-1)
# simulate another 10000 samples
MHchain2 = run_metropolis_MCMC(startvalue, nreps)

# remove the first 5000 as burn-in
burnIn = 5000

# computing average acceptance probability
(acceptance1 = 1-mean(duplicated(MHchain1[-(1:burnIn),])))
```

```
## [1] 0.4373125
```

```r
(acceptance2 = 1-mean(duplicated(MHchain2[-(1:burnIn),])))
```

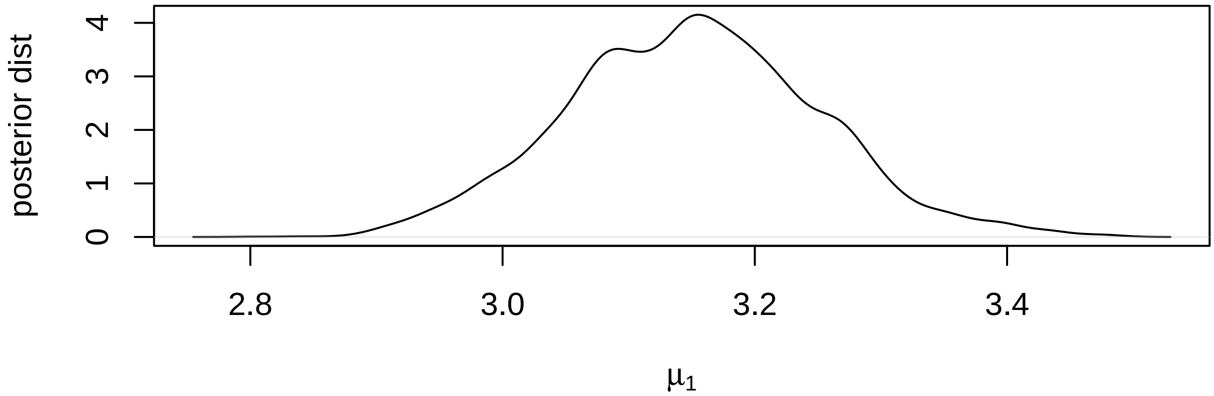
```
## [1] 0.4435113
```

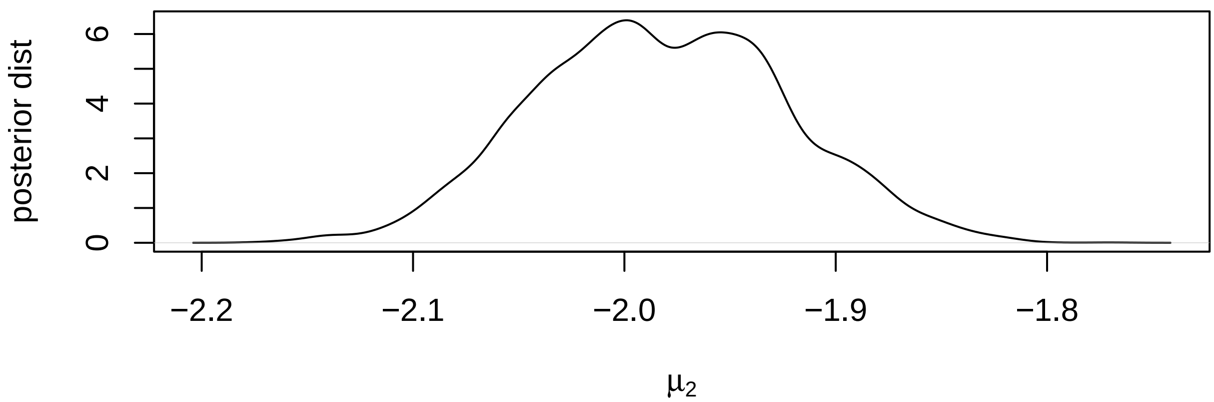
```r
par(mfrow=c(2,1), mar=c(4,4,1,1))
plot(1:(nreps+1), MHchain1[,1], type="l", col="red", ylim = c(min(MHchain1[,1], MHchain2[,1]), max(MHcha
     xlab = "iteration", ylab ="mu1")
points(1:(nreps+1), MHchain2[,1], type="l", col="blue")

plot(1:(nreps+1), MHchain1[,2], type="l", col="red", ylim = c(min(MHchain1[,2], MHchain2[,2]), max(MHcha
     xlab = "iteration", ylab ="mu2")
points(1:(nreps+1), MHchain2[,2], type="l", col="blue")
```


mu1
0.0
1.5
3.0
0
2000
4000
6000
8000
10000
iteration
mu2
−2.0
−1.0
0.0
0
2000
4000
6000
8000
10000
iteration

### 2e

```r
par(mfrow=c(2,1), mar=c(4,4,1,1))
plot(density(MHchain1[-(1:burnIn),1]), ylab="posterior dist", xlab=expression(mu[1]), main="")
plot(density(MHchain1[-(1:burnIn),2]), ylab="posterior dist", xlab=expression(mu[2]), main="")
```




```r
mean(MHchain1[-(1:burnIn),1])
```

```
## [1] 3.150803
```

```r
mean(MHchain1[-(1:burnIn),2])
```

```
## [1] -1.982535
```

```r
quantile(MHchain1[-(1:burnIn),1], probs= c(0.05, 0.95))
```

```
##       5%      95%
## 2.988247 3.312308
```

```r
quantile(MHchain1[-(1:burnIn),2], probs= c(0.05, 0.95))
```

```
##        5%       95%
## -2.077089 -1.883055
```

Estimated marginal posterior mean for $\mu_1$ = 3.1508

Estimated marginal posterior mean for $\mu_2$ = -1.9825

90% credible interval for $\mu_1$ = (2.9882, 3.3123)

90% credible interval for $\mu_2$ = (-2.0771, -1.8831)

(f)

2.

f) $\log q_1(\mu_1) \propto E_{\mu_2}\left[-\frac{1}{2}\left(\sum_{i=1}^{100}(x_i-\mu_1)^2+3\mu_1^2+4\mu_1\mu_2\right)\right]$

$\log q_2(\mu_2) \propto E_{\mu_1}\left[-\left(\sum_{i=1}^{150}(y_i-\mu_2)^2+\frac{3}{2}\mu_2^2+2\mu_1\mu_2\right)\right]$

from (a)

$\Rightarrow q_1(\mu_1)$ : pdf of $N(\mu_1^*, \sigma_1^{2*})$, $\mu_1^* = \frac{\sum_{i=1}^{100} x_i - 2E_{\mu_2}(\mu_2)}{103}$, $\sigma_1^{2*} = \frac{1}{103}$

$\Rightarrow q_2(\mu_2)$ : pdf of $N(\mu_2^*, \sigma_2^{2*})$, $\mu_2^* = 2\,\frac{\sum_{i=1}^{150} y_i - E_{\mu_1}(\mu_1)}{303}$, $\sigma_2^{2*} = \frac{1}{303}$

$E_{\mu_2}(\mu_2) = \mu_2^*$

$E_{\mu_1}(\mu_1) = \mu_1^*$

(g)

2.

g) $ELBO\left(q^*_{\mu_1}(\mu_1), q^*_{\mu_2}(\mu_2)\right) = E_{\mu_1\mu_2}\left[\log p(X, Y, \mu_1, \mu_2) - \log\left(q^*_{\mu_1}(\mu_1) q^*_{\mu_2}(\mu_2)\right)\right]$

$= E_{\mu_1,\mu_2}\left[\log\left(p(X|\mu_1)\, p(Y|\mu_2)\, p(\mu_1,\mu_2)\right) - \log q^*_{\mu_1}(\mu_1) - \log q^*_{\mu_2}(\mu_2)\right.$

$= E_{\mu_1,\mu_2}\left[\log\left(p(X|\mu_1)\, p(Y|\mu_2)\, p(\mu_1,\mu_2)\right)\right] - E_{\mu_1\mu_2}\left[\log q^*_{\mu_1}(\mu_1)\right] - E_{\mu_1\mu_2}\left[\log q^*_{\mu_2}(\mu_2)\right]$

$E_{\mu_1,\mu_2}\left[\log q^*_{\mu_1}(\mu_1)\right] \propto -\frac{1}{2}\log\sigma_1^{2*} - \frac{E_{\mu_1}\left[(\mu_1-\mu_1^*)^2\right]}{2\sigma_1^{2*}}$

$\propto -\frac{1}{2}\log\sigma_1^{2*} - \frac{\sigma_1^{2*}}{2\sigma_1^{2*}}$

$\propto -\frac{1}{2}\log\sigma_1^{2*}$

$\Rightarrow E_{\mu_1\mu_2}\left[\log q^*_{\mu_2}(\mu_2)\right] \propto -\frac{1}{2}\log\sigma_2^{2*}$

$\propto E_{\mu_1\mu_2}\left[-\frac{1}{2}\sum_{i=1}^{100}(x_i-\mu_1)^2 - \sum_{i=1}^{150}(y_i-\mu_2)^2 - \frac{3}{2}\mu_1^2 - 2\mu_1\mu_2 - \frac{3}{2}\mu_2^2\right] + \frac{1}{2}\log\sigma_1^{2*} + \log\sigma_2^{2*}$

from (a)

$\propto -\frac{103}{2}\left(\sigma_1^{2*} + \mu_1^{2*}\right) + \mu_1^*\sum_{i=1}^{100}x_i - \frac{303}{2}\left(\sigma_2^{2*} + \mu_2^{2*}\right)$

$+ 2\mu_2^*\sum_{i=1}^{150}y_i - 2\mu_1\mu_2 + \log\sigma_1^{2*} + \log\sigma_2^{2*} + \text{const.}$

$\propto -\frac{103}{2}\left(\mu_1^{2*}\right) + \mu_1^*\sum_{i=1}^{100}x_i - \frac{303}{2}\left(\mu_2^{2*}\right) + 2\mu_2^*\sum_{i=1}^{150}y_i - 2\mu_1^*\mu_2^* + \text{const.}$

## 2h

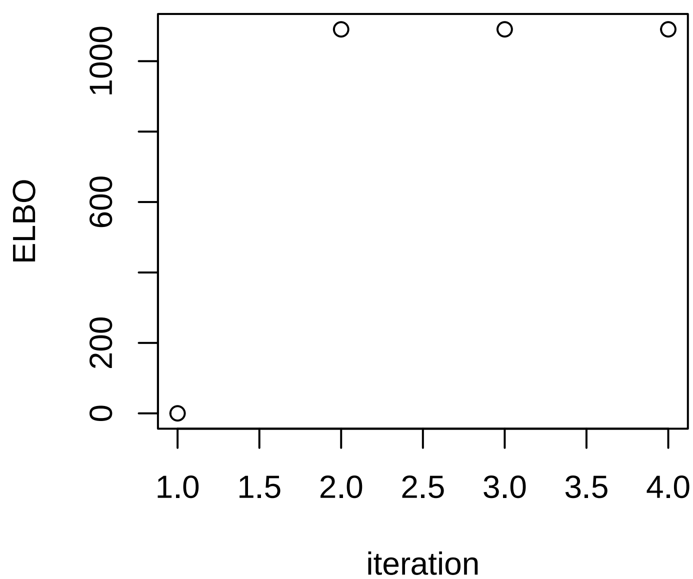
```r
# x,y: data
# mu10, mu20 : prior for mu
# initial values for mu1*, mu2*,: mu1.vi.init, mu2.vi.init
# epsilon : If the ELBO has changed by less than epsilon, the CAVI algorithm will stop
# max.iter : maximum number of iteration
cavi.normal <- function(x, y, mu1.vi.init, mu2.vi.init, epsilon=1e-5, max.iter=100) {

  mu1.vi = mu1.vi.init
  mu2.vi = mu2.vi.init


  # store the ELBO for each iteration
  elbo = c()

  # I will store mu*, sigma2*, a*, b* for each iteration
  mu1.vi.list = mu2.vi.list = c()

  # compute the ELBO using initial values of mu*

  elbo = c(elbo, (-103/2) * mu1.vi^2 + mu1.vi*sum(x) - (303/2) * mu2.vi^2 + 2*mu2.vi*sum(y) - 2*mu1.vi*m
  mu1.vi.list = c(mu1.vi.list, mu1.vi)
  mu2.vi.list = c(mu2.vi.list, mu2.vi)

  # set the change in the ELBO with 1
  delta.elbo = 1

  # number of iteration
  n.iter = 1

  # If the elbo has changed by less than epsilon, the CAVI will stop.
  while((delta.elbo > epsilon) & (n.iter <= max.iter)){

    # Update mu.vi and sigma2.vi
    mu1.vi = (sum(x) - 2*mu2.vi)/103
    mu2.vi = (2*sum(y)-2*mu1.vi)/303


    # compute the ELBO using the current values of mu*
    elbo = c(elbo, (-103/2) * mu1.vi^2 + mu1.vi*sum(x) - (303/2) * mu2.vi^2 + 2*mu2.vi*sum(y) - 2*mu1.vi
    mu1.vi.list = c(mu1.vi.list, mu1.vi)
    mu2.vi.list = c(mu2.vi.list, mu2.vi)

    # compute the change in the elbo
    delta.elbo = elbo[length(elbo)]  - elbo[length(elbo)-1]

    # increase the number of iteration
    n.iter = n.iter + 1
  }

  return(list(elbo = elbo, mu1.vi.list = mu1.vi.list,
              mu2.vi.list=mu2.vi.list))
```

```r
}
```

```r
set.seed(456)

cavi1 = cavi.normal(x, y, mu1.vi.init=0, mu2.vi.init=0, epsilon=1e-5, max.iter=100)
cavi.res = cavi1
cavi.res$elbo
```

```
## [1]    0.000 1090.321 1090.397 1090.397
```

```r
plot(cavi.res$elbo, ylab='ELBO', xlab='iteration')
```



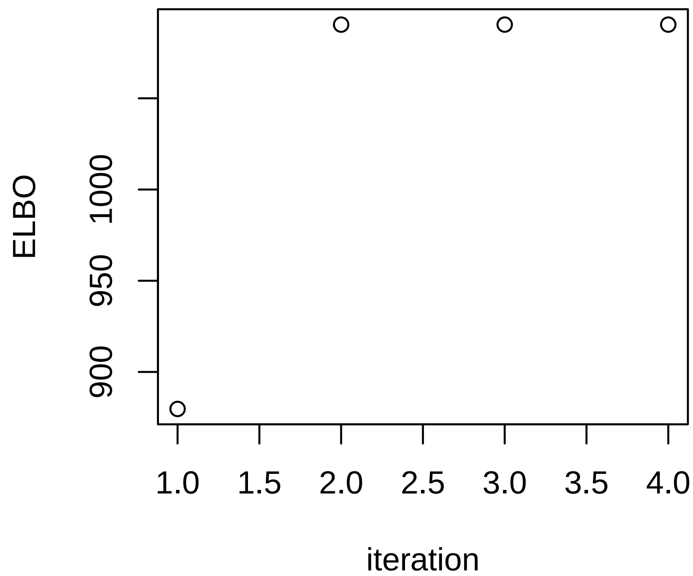

```r
print(paste("mu1* and mu2* = (",
            round(cavi.res$mu1.vi.list[length(cavi.res$mu1.vi.list)],2), ",",
            round(cavi.res$mu2.vi.list[length(cavi.res$mu2.vi.list)],2), ")", sep=""))
```

```
## [1] "mu1* and mu2* = (3.14,-1.98)"
```

```r
cavi2 = cavi.normal(x, y, mu1.vi.init=2, mu2.vi.init=-1, epsilon=1e-5, max.iter=100)
cavi.res = cavi2
cavi.res$elbo
```

```
## [1]  879.7223 1090.3781 1090.3968 1090.3968
```

```r
plot(cavi.res$elbo, ylab='ELBO', xlab='iteration')
```




```r
print(paste("mu1* and mu2* = (",
            round(cavi.res$mu1.vi.list[length(cavi.res$mu1.vi.list)],2), ",",
            round(cavi.res$mu2.vi.list[length(cavi.res$mu2.vi.list)],2), ")", sep=""))
```

```
## [1] "mu1* and mu2* = (3.14,-1.98)"
```