

Bayesian Learning

Lab 3

Emil K Svensson and Rasmus Holm

2017-05-09

Question 1

```
rainfall <- read.table("../data/rainfall.dat", header=F)
```

Prior

$$\begin{aligned}\mu &\sim N(\mu_0, \tau_0^2) \\ \sigma^2 &\sim \text{Inv-}\chi^2(\nu_0, \sigma_0^2)\end{aligned}$$

Likelihood

$$\mathbf{y}|\mu, \sigma^2 \sim N(\mu, \sigma^2)$$

Posterior

$$\begin{aligned}\mu|\sigma^2, \mathbf{x} &\sim N(\mu_n, \tau_n^2) \\ \sigma^2|\mu, \mathbf{x} &\sim \text{Inv-}\chi^2\left(\nu_n, \frac{\nu_0\sigma_0^2 + \sum_{i=1}^n (x_i - \mu)^2}{n + \nu_0}\right)\end{aligned}$$

where

$$\begin{aligned}\mu_n &= \frac{\frac{1}{\tau_0^2}}{\frac{1}{\tau_0^2} + \frac{n}{\sigma^2}} \mu_0 + \frac{\frac{n}{\sigma^2}}{\frac{1}{\tau_0^2} + \frac{n}{\sigma^2}} \bar{x} \\ \frac{1}{\tau_n^2} &= \frac{1}{\tau_0^2} + \frac{n}{\sigma^2} \\ \nu_n &= \nu_0 + n\end{aligned}$$

a)

```
library(geoR)

mun <- function(x, sigmasq, hyperparams){
  n <- length(x)
  denom <- ((1 / hyperparams$tausq0) + (n / sigmasq))
```

```

    pt1 <- ((1 / hyperparams$tausq0) / denom) * hyperparams$mu0
    pt2 <- ((n / sigmasq) / denom) * mean(x)

    pt1 + pt2
}

taun <- function(x, sigmasq, hyperparams){
  hyperparams$tausq0 + (sigmasq / length(x))
  1 / hyperparams$tausq0
}

nun <- function(x, hyperparams){
  hyperparams$nu0 + length(x)
}

sigmasqn <- function(x, mu, hyperparams){
  (hyperparams$nu0 * hyperparams$sigmasq0 + sum((x - mu)^2 )) / (length(x) + hyperparams$nu0)
}

musampler <- function(x, sigmasq, hyperparams){
  mu <- mun(x, sigmasq, hyperparams)
  sigma <- sqrt(taun(x, sigmasq, hyperparams))
  rnorm(1, mu, sigma)
}

sigmasampler <- function(x, mu, hyperparams){
  scale <- sigmasqn(x, mu, hyperparams)
  df <- nun(x, hyperparams)
  rinvchisq(1, df, scale)
}

gibbs <- function(x, iter, init, hyperparams){
  samples <- matrix(NA, ncol = 2, nrow = iter + 1)
  samples[1,] <- init

  for (i in 2:(iter+1)){
    mu <- musampler(x, samples[i-1, 2], hyperparams)
    sigma <- sigmasampler(x, mu, hyperparams)
    samples[i,] <- c(mu, sigma)
  }

  colnames(samples) <- c("mu", "sigmasq")
  samples
}

set.seed(123456)

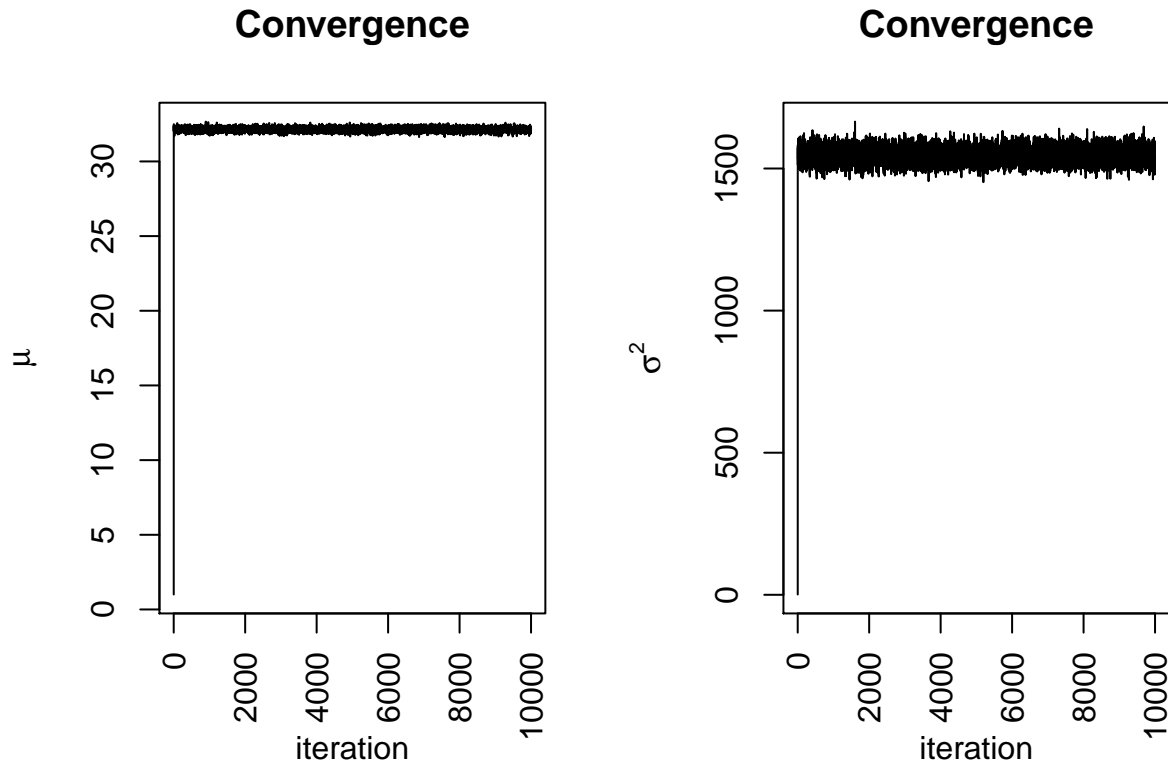
hyperparams <- list(mu0=0, tausq0=50, nu0=1, sigmasq0=50)
gibbs_params <- gibbs(x=rainfall$V1, iter = 10000, init = c(1,1), hyperparams)

```

```

par(mfrow=c(1, 2), las=3)
plot(gibbs_params[, 1], type = "l", xlab="iteration", ylab=expression(mu), main="Convergence")
plot(gibbs_params[, 2], type = "l", xlab="iteration", ylab=expression(sigma^2), main="Convergence")

```



b)

```

nn <- function(I){
  n1 <- sum(I == 1)
  n2 <- sum(I == 2)
  list(n1 = n1, n2 = n2)
}

dd <- function(x, I) {
  d1 <- x[I == 1]
  d2 <- x[I == 2]
  list(d1=d1, d2=d2)
}

pisampler <- function(x, I, hyperparams){
  n <- nn(I)
  rbeta(1, shape1 = n$n1 + hyperparams$a1, shape2 = n$n2 + hyperparams$a2)
}

sigmasq2sampler <- function(x, I, hyperparams){

```

```

n <- nn(I)
d <- dd(x, I)

vnsn1 <- hyperparams$nu0 * hyperparams$sigmasq0 +
  (n$n1 - 1) * var(d$d1) +
  (((1 / hyperparams$tausq0) * n$n1) / ((1 / hyperparams$tausq0) + n$n1)) *
  ((mean(d$d1) - hyperparams$mu0)^2)

vnsn2 <- hyperparams$nu0 * hyperparams$sigmasq0 +
  (n$n2 - 1) * var(d$d2) +
  (((1 / hyperparams$tausq0) * n$n2) / ((1 / hyperparams$tausq0) + n$n2)) *
  ((mean(d$d2) - hyperparams$mu0)^2)

vn1 <- hyperparams$nu0 + n$n1
vn2 <- hyperparams$nu0 + n$n2

sn1 <- vnsn1 / vn1
sn2 <- vnsn2 / vn2

sigmasq1 <- rinvcchisq(1, vn1, sn1)
sigmasq2 <- rinvcchisq(1, vn2, sn2)

c(sigmasq1, sigmasq2)
}

mu2sampler <- function(x, I, sigmasq, hyperparams){
  d <- dd(x, I)

  sigma1 <- sqrt(taun(d$d1, sigmasq[1], hyperparams))
  mu1 <- mun(d$d1, sigmasq[1], hyperparams)

  sigma2 <- sqrt(taun(d$d2, sigmasq[2], hyperparams))
  mu2 <- mun(d$d2, sigmasq[2], hyperparams)

  mu1 <- rnorm(1, mu1, sigma1)
  mu2 <- rnorm(1, mu2, sigma2)

  c(mu1, mu2)
}

Isampler <- function(x, pi, mu, sigmasq){
  nom <- (1 - pi) * dnorm(x, mu[2], sqrt(sigmasq[2]))
  denom <- pi * dnorm(x, mu[1], sqrt(sigmasq[1])) + nom
  theta <- nom / denom
  rbinom(length(x), prob = theta, size = 1) + 1 # to get them into 1 and 2
}

ysampler <- function(pi, mu, sigmasq){
  pi * rnorm(1, mu[1], sqrt(sigmasq[1])) +
    (1 - pi) * rnorm(1, mu[2], sqrt(sigmasq[2]))
}

mixedgibbs <- function(x, iter, init, hyperparams){

```

```

params_samples <- matrix(NA, ncol = 5, nrow = iter)
samples <- rep(NA, iter)
I <- init

for (i in 1:iter){
  pi <- pisampler(x, I, hyperparams)
  sigmasq <- sigmasq2sampler(x, I, hyperparams)
  mu <- mu2sampler(x, I, sigmasq, hyperparams)
  I <- Isampler(x, pi, mu, sigmasq)

  samples[i] <- ysampler(pi, mu, sigmasq)
  params_samples[i,] <- c(mu, sigmasq, pi)
}

colnames(params_samples) <- c("mu1", "mu2", "sigmasq1", "sigmasq2", "pi")
list(samples=samples, params=params_samples)
}

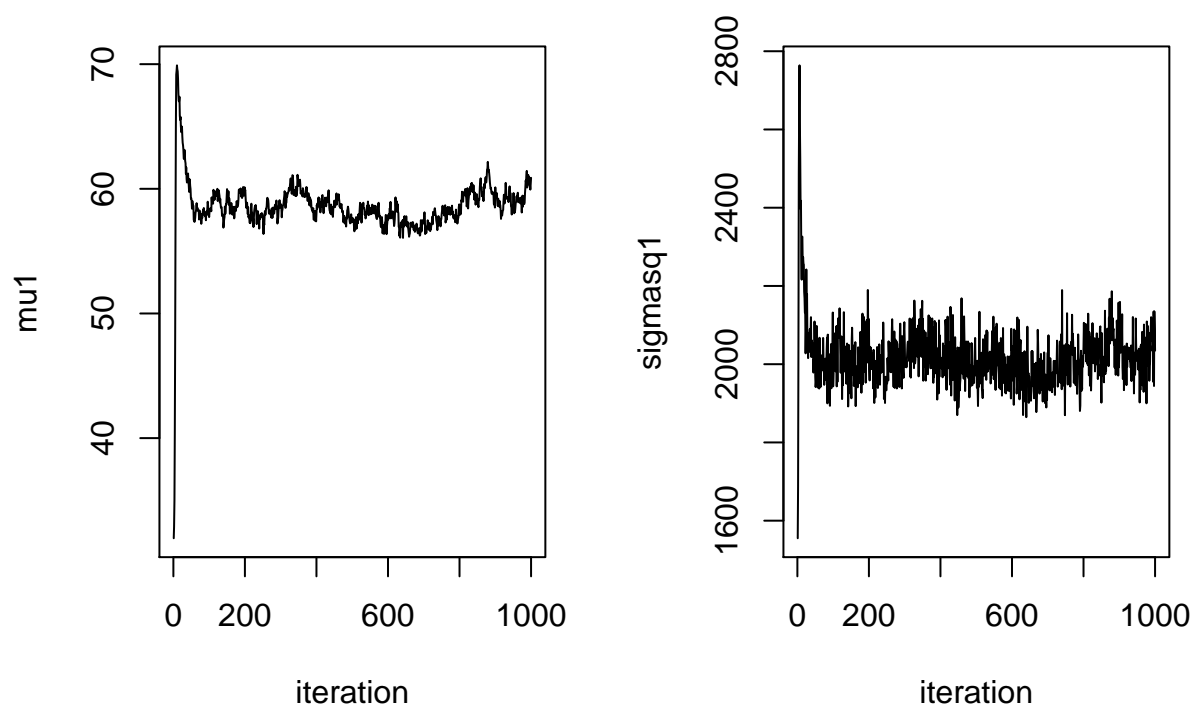
set.seed(123456)

hyperparams <- list(a1=1, a2=1, mu0=1, tausq0=50, nu0=1, sigmasq0=50)
init <- sample(c(1, 2), size=length(rainfall$V1), replace=TRUE)
iter <- 1000

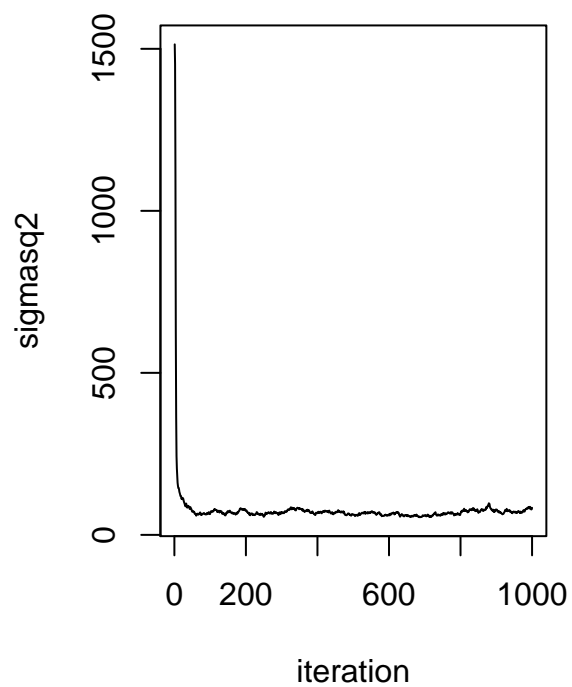
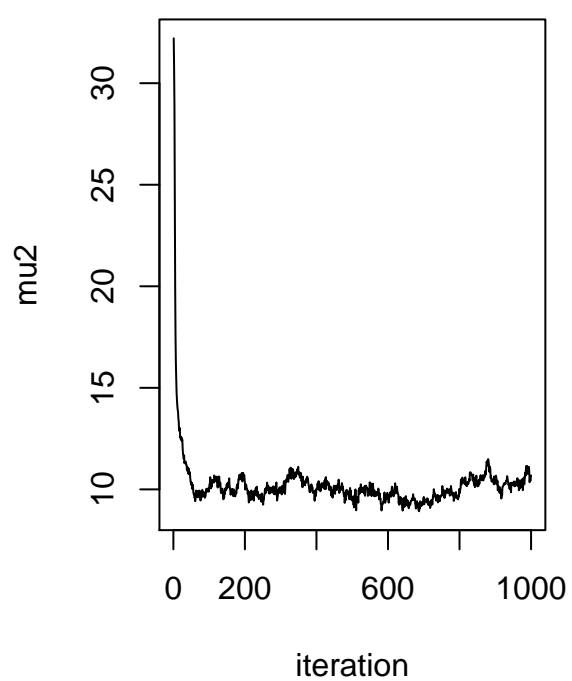
mixed <- mixedgibbs(rainfall$V1, iter, init, hyperparams)
mixed_params <- mixed$params
mixed_samples <- mixed$samples

par(mfrow = c(1, 2))
plot(x = 1:iter, mixed_params[, 1], type = "l", xlab="iteration", ylab = "mu1")
plot(x = 1:iter, mixed_params[, 3], type = "l", xlab="iteration", ylab = "sigmasq1")

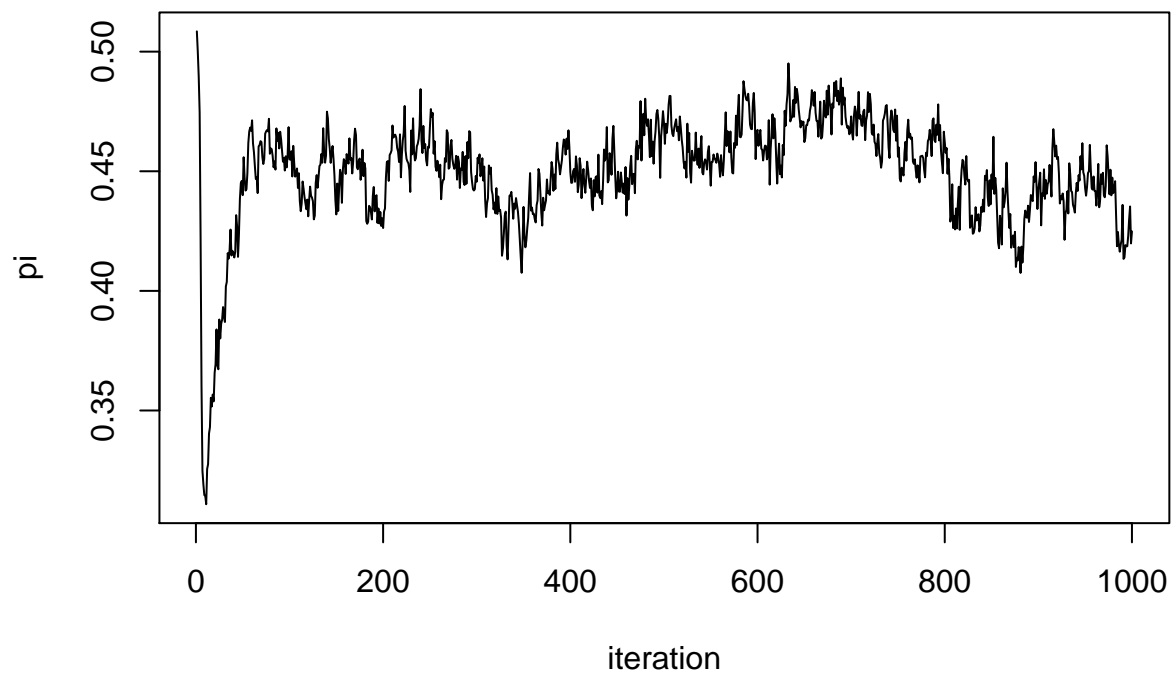
```



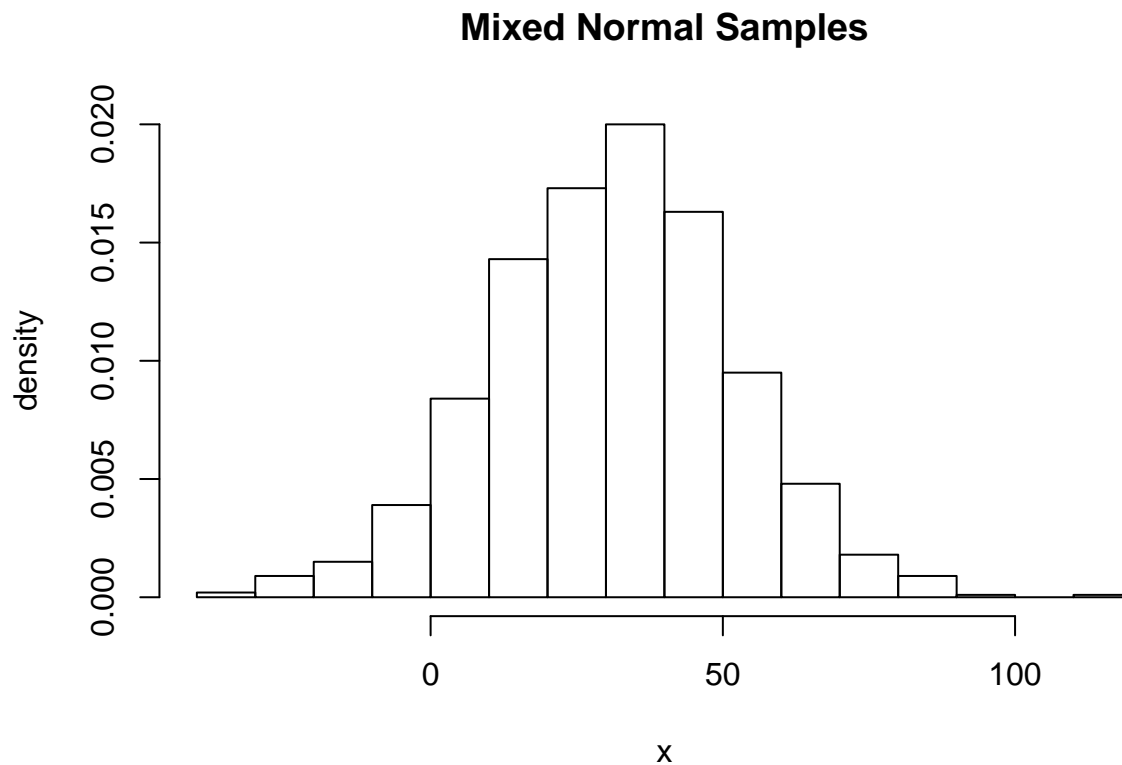
```
plot(x = 1:iter, mixed_params[, 2], type = "l", xlab="iteration", ylab = "mu2")
plot(x = 1:iter, mixed_params[, 4], type = "l", xlab="iteration", ylab = "sigmasq2")
```



```
par(mfrow = c(1,1))
plot(x = 1:iter, mixed_params[, 5], type = "l", xlab="iteration", ylab = "pi")
```



```
hist(mixed_samples, main="Mixed Normal Samples", xlab="x", ylab="density", freq=FALSE)
```

c)

```
n <- 10000

## Data Density
plot(density(rainfall$V1), lwd=2, col="black",
      xlim=c(-100, 200), xlab="x", ylab="density",
      main="")

## Model A
mu <- mean(gibbs_params[, "mu"])
sigmasq <- mean(gibbs_params[, "sigmasq"])

samples <- rnorm(n, mu, sqrt(sigmasq))

lines(density(samples), lwd=2, col="orange")

## Model B
mu1 <- mean(mixed_params[, "mu1"])
sigmasq1 <- mean(mixed_params[, "sigmasq1"])

mu2 <- mean(mixed_params[, "mu2"])
sigmasq2 <- mean(mixed_params[, "sigmasq2"])
```

```

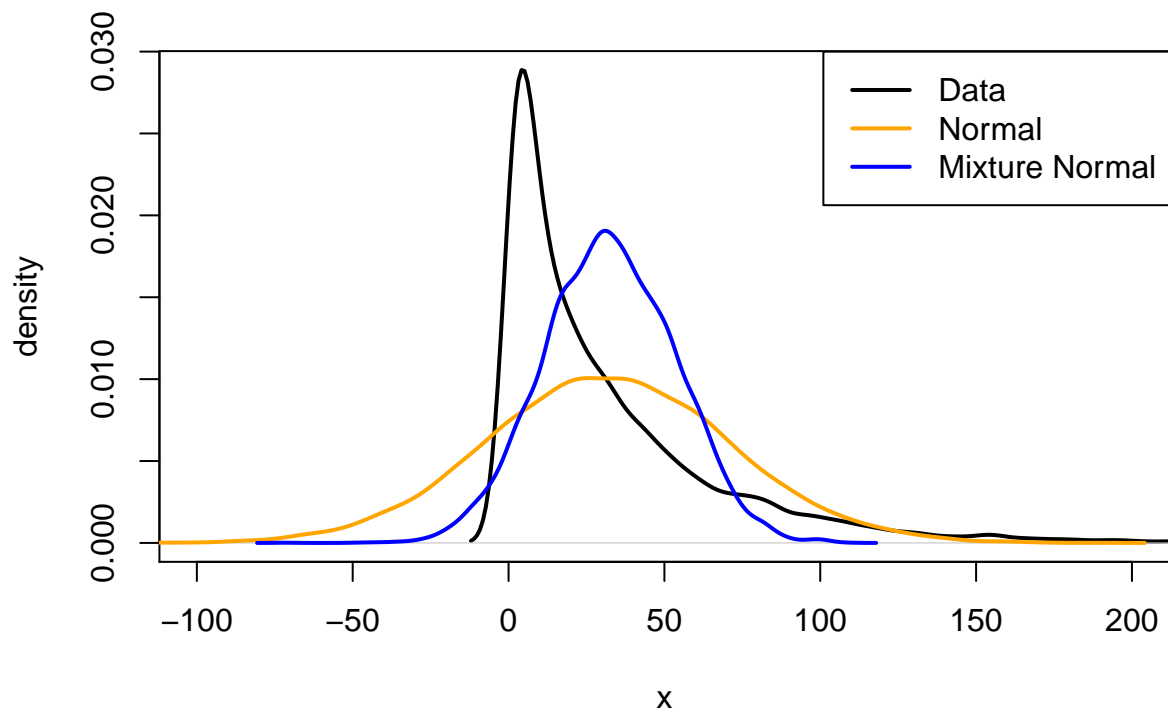
pi <- mean(mixed_params[, "pi"])

samples <- pi * rnorm(n, mu1, sqrt(sigmasq1)) + (1 - pi) * rnorm(n, mu2, sqrt(sigmasq2))

lines(density(samples), lwd=2, col="blue")

legend(101, 0.03, lty=c(1, 1, 1), lwd=c(2, 2, 2), col=c("black", "orange", "blue"),
      legend=c("Data", "Normal", "Mixture Normal"))

```



Question 2

a)

b)

c)