Bayesian Changepoint Detection

Yuanhe Zhang

Load library

```
library(ggplot2)
library(hrbrthemes)
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.

##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and

##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(viridisLite)
library(viridis)
library(dplyr)
```

Load dataset

```
load("C:/360Downloads/R/ST407/Assignment 2/a2data.Rdata")
```

Question 1

(a)

Generally,

Prior distribution of k,

$$P(k) = \begin{cases} \frac{k}{c} & k \leq \left\lfloor \frac{n-1}{2} \right\rfloor \\ \frac{n-l}{c} & k > \left\lfloor \frac{n-1}{2} \right\rfloor \end{cases}$$

Prior distribution of μ_1 and μ_2,

$$f^{prior}(\mu_1, \mu_2) = \frac{1}{\sqrt{20\pi}} exp(-\frac{\mu_1^2}{20}) \frac{1}{\sqrt{20\pi}} exp(-\frac{\mu_2^2}{20}) = \frac{1}{20\pi} exp(-\frac{\mu_1^2 + \mu_2^2}{20})$$

And Likelihood:

$$l(\mathbf{y}|\mu_1, \mu_2, k) = \prod_{p=1}^{k} \phi_{\mu_1,1}(y_p) \prod_{q=k+1}^{n} \phi_{\mu_2,1}(y_q) = \prod_{p=1}^{k} \frac{1}{\sqrt{2\pi}} exp(-\frac{(y_p - \mu_1)^2}{2}) \prod_{q=k+1}^{n} \frac{1}{\sqrt{2\pi}} exp(-\frac{(y_q - \mu_2)^2}{2})$$

Set $k \leq \left\lfloor \frac{n-1}{2} \right\rfloor$, then prior distribution of k is $P(k) = \frac{k}{c}$,

then the joint posterior is,

$$\begin{aligned} & f_1^{(post)}(\mu_1, \mu_2, k|\mathbf{y}) \\ & \propto f^{prior}(\mu_1, \mu_2) * P(k) * l(\mathbf{y}|\mu_1, \mu_2, k) \\ &= \frac{k}{c} \frac{1}{\sqrt{20\pi}} exp(-\frac{\mu_1^2}{20}) \frac{1}{\sqrt{20\pi}} exp(-\frac{\mu_2^2}{20}) \prod_{p=1}^{k} \frac{1}{\sqrt{2\pi}} exp(-\frac{(y_p - \mu_1)^2}{2}) \prod_{q=k+1}^{n} \frac{1}{\sqrt{2\pi}} exp(-\frac{(y_q - \mu_2)^2}{2}) \\ &= \frac{k}{20\pi c} exp(-\frac{\mu_1^2 + \mu_2^2}{20}) (\frac{1}{\sqrt{2\pi}})^k \prod_{p=1}^{k} exp(-\frac{(y_p - \mu_1)^2}{2}) (\frac{1}{\sqrt{2\pi}})^{n-k} \prod_{q=k+1}^{n} exp(-\frac{(y_q - \mu_2)^2}{2}) \\ &= \frac{k}{20\pi c} (\frac{1}{\sqrt{2\pi}})^n exp(-\frac{\mu_1^2 + \mu_2^2}{20}) \prod_{p=1}^{k} exp(-\frac{(y_p - \mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q - \mu_2)^2}{2}) \\ &= \frac{k}{10c(2\pi)^{\frac{n}{2}+1}} exp(-\frac{\mu_1^2 + \mu_2^2}{20}) \prod_{p=1}^{k} exp(-\frac{(y_p - \mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q - \mu_2)^2}{2}) \\ & \propto k * exp(-\frac{\mu_1^2 + \mu_2^2}{20}) \prod_{p=1}^{k} exp(-\frac{(y_p - \mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q - \mu_2)^2}{2}) \end{aligned}$$

We write the joint posterior up to a normalising constant Z(define later),

$$f_1^{(post)}(\mu_1, \mu_2, k|\mathbf{y}) = \frac{k * exp(-\frac{\mu_1^2+\mu_2^2}{20}) \prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})}{Z}$$

Set $k > \left\lfloor \frac{n-1}{2} \right\rfloor$, then prior distribution of k is $P(k) = \frac{n-k}{c}$,

then the joint posterior is,

$$
\begin{aligned}
& f_2^{(post)}(\mu_1, \mu_2, k|\mathbf{y}) \\
& \propto f^{prior}(\mu_1, \mu_2) * P(k) * l(\mathbf{y}|\mu_1, \mu_2, k) \\
= \frac{n-k}{c}\frac{1}{\sqrt{20\pi}}exp(-\frac{\mu_1^2}{20})\frac{1}{\sqrt{20\pi}}exp(-\frac{\mu_2^2}{20})\prod_{p=1}^{k}\frac{1}{\sqrt{2\pi}}exp(-\frac{(y_p-\mu_1)^2}{2})\prod_{q=k+1}^{n}\frac{1}{\sqrt{2\pi}}exp(-\frac{(y_q-\mu_2)^2}{2}) \\
&= \frac{n-k}{20\pi c}exp(-\frac{\mu_1^2+\mu_2^2}{20})\prod_{p=1}^{k}\frac{1}{\sqrt{2\pi}}exp(-\frac{(y_p-\mu_1)^2}{2})\prod_{q=k+1}^{n}\frac{1}{\sqrt{2\pi}}exp(-\frac{(y_q-\mu_2)^2}{2}) \\
&= \frac{n-k}{10c(2\pi)^{\frac{n}{2}+1}}exp(-\frac{\mu_1^2+\mu_2^2}{20})\prod_{p=1}^{k}exp(-\frac{(y_p-\mu_1)^2}{2})\prod_{q=k+1}^{n}exp(-\frac{(y_q-\mu_2)^2}{2}) \\
&\propto (n-k) * exp(-\frac{\mu_1^2+\mu_2^2}{20})\prod_{p=1}^{k}exp(-\frac{(y_p-\mu_1)^2}{2})\prod_{q=k+1}^{n}exp(-\frac{(y_q-\mu_2)^2}{2})
\end{aligned}
$$

We write the joint posterior up to a normalising constant Z(define later),

$$
f_2^{(post)}(\mu_1, \mu_2, k|\mathbf{y}) = \frac{(n-k) * exp(-\frac{\mu_1^2+\mu_2^2}{20})\prod_{p=1}^{k}exp(-\frac{(y_p-\mu_1)^2}{2})\prod_{q=k+1}^{n}exp(-\frac{(y_q-\mu_2)^2}{2})}{Z}
$$

Lastly, we define the normalising constant Z as,

$$
Z = 10c(2\pi)^{\frac{n}{2}+1}\sum_k P^{(prior)}(k)\int_{\mu_1}\int_{\mu_2} f^{prior}(\mu_1, \mu_2) l(\mathbf{y}|\mu_1, \mu_2, k)\, d\mu_1\mu_2
$$

Finally, we write our joint posterior of μ_1, μ_2, k given a sequence of observations $\mathbf{y}$ up to a noemalising constant Z as,

$$
f^{(post)}(\mu_1, \mu_2, k|\mathbf{y}) = \begin{cases} \frac{k*exp(-\frac{\mu_1^2+\mu_2^2}{20})\prod_{p=1}^{k}exp(-\frac{(y_p-\mu_1)^2}{2})\prod_{q=k+1}^{n}exp(-\frac{(y_q-\mu_2)^2}{2})}{Z} & k \le \lfloor\frac{n-1}{2}\rfloor \\ \frac{(n-k)*exp(-\frac{\mu_1^2+\mu_2^2}{20})\prod_{p=1}^{k}exp(-\frac{(y_p-\mu_1)^2}{2})\prod_{q=k+1}^{n}exp(-\frac{(y_q-\mu_2)^2}{2})}{Z} & k > \lfloor\frac{n-1}{2}\rfloor \end{cases}
$$

where $Z = 10c(2\pi)^{\frac{n}{2}+1}\sum_k P^{(prior)}(k)\int_{\mu_1}\int_{\mu_2} f^{prior}(\mu_1, \mu_2) l(\mathbf{y}|\mu_1, \mu_2, k)\, d\mu_1\mu_2$.

(b)

First, we compute the full conditional distribution of μ_1. For simplicity, whether it is k or $n-k$ in the joint posterior will be removed so we can write two posterior corresponding to different range of k to just one posterior for the full conditional distribution of μ_1 (same for μ_2),

$$
\begin{aligned}
& f(\mu_1|\mathbf{y},\mu_2,k) \\
& \propto exp(-\frac{\mu_1^2}{20})\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \\
& = exp(-\frac{\mu_1^2}{20})exp(-\frac{\sum_{p=1}^{k}(y_p-\mu_1)^2}{2}) \\
& = exp(-\frac{\mu_1^2}{20})exp(-\frac{\sum_{p=1}^{k}(y_p^2-2y_p\mu_1+\mu_1^2)}{2}) \\
& = exp(-\frac{\mu_1^2}{20})exp(-\frac{\sum_{p=1}^{k}y_p^2-2\sum_{p=1}^{k}y_p\mu_1+k\mu_1^2}{2}) \\
& \propto exp(-\frac{\mu_1^2}{20})exp(\frac{2\sum_{p=1}^{k}y_p\mu_1-k\mu_1^2}{2}) \\
& = exp(\frac{-\frac{\mu_1^2}{10}}{2}+\frac{2\sum_{p=1}^{k}y_p\mu_1-k\mu_1^2}{2}) \\
& = exp(-\frac{-2\sum_{p=1}^{k}y_p\mu_1+(k+\frac{1}{10})\mu_1^2}{2}) \\
& = exp(-\frac{\mu_1^2-2\frac{\sum_{p=1}^{k}y_p}{(k+\frac{1}{10})}\mu_1}{\frac{2}{k+\frac{1}{10}}}) \\
& = exp(-\frac{(\mu_1-\frac{\sum_{p=1}^{k}y_p}{k+\frac{1}{10}})^2}{\frac{2}{k+\frac{1}{10}}})*const
\end{aligned}
$$

therefore we can obtain that,

$$
\mu_1|\mathbf{y},\mu_2,k\sim N(\frac{\sum_{p=1}^{k}y_p}{k+\frac{1}{10}},\frac{1}{k+\frac{1}{10}})
$$

First, we compute the full conditional distribution of μ_2.

$$
\begin{aligned}
& f(\mu_2|\mathbf{y},\mu_1,k) \\
& \propto exp(-\frac{\mu_2^2}{20})\prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2}) \\
& \propto exp(-\frac{\mu_2^2}{20})exp(\frac{2\sum_{q=k+1}^{n}y_q\mu_2-(n-k)\mu_2^2}{2}) \\
& = exp(-\frac{\frac{\mu_2^2}{10}}{2}-\frac{-2\sum_{q=k+1}^{n}y_q\mu_2+(n-k)\mu_2^2}{2}) \\
& = exp(-\frac{-2\sum_{q=k+1}^{n}y_q\mu_2+(n-k+\frac{1}{10})\mu_2^2}{2}) \\
& = exp(-\frac{\mu_2^2-2\frac{\sum_{q=k+1}^{n}y_q}{(n-k+\frac{1}{10})}\mu_2}{\frac{2}{n-k+\frac{1}{10}}}) \\
& = exp(-\frac{(\mu_2-\frac{\sum_{q=k+1}^{n}y_q}{n-k+\frac{1}{10}})^2}{\frac{2}{n-k+\frac{1}{10}}})*const
\end{aligned}
$$

therefore we can obtain that,

$$\mu_2|\mathbf{y},\mu_1,k \sim N(\frac{\sum_{q=k+1}^{n} y_q}{n-k+\frac{1}{10}},\frac{1}{n-k+\frac{1}{10}})$$

Finally, we compute the full conditional distribution of k.

Set $k \le \lfloor\frac{n-1}{2}\rfloor$,

$$f_1(k|\mathbf{y},\mu_1,\mu_2) \propto k * \prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})$$

Set $k > \lfloor\frac{n-1}{2}\rfloor$,

$$f_2(k|\mathbf{y},\mu_1,\mu_2) \propto (n-k) * \prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})$$

Define a normalising constant C as,

$$C = \sum_{k} \left\{ \Big(\mathbf{1}_{\{k\le\lfloor\frac{n-1}{2}\rfloor\}} * k + \mathbf{1}_{\{k>\lfloor\frac{n-1}{2}\rfloor\}} * (n-k)\Big) * \prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2}) \right\}$$

where $\mathbf{1}_{\{\cdot\}}$ is the indicator function.

then we can obtain the full conditional distribution of k.

$$f(k|\mu_1,\mu_2,\mathbf{y}) = \begin{cases} \frac{k*\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})}{C} & k \le \lfloor\frac{n-1}{2}\rfloor \\ \frac{(n-k)*\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})}{C} & k > \lfloor\frac{n-1}{2}\rfloor \end{cases}$$

And the full conditional distribution of μ_1, μ_2.

$$\mu_1|\mathbf{y},\mu_2,k \sim N(\frac{\sum_{p=1}^{k} y_p}{k+\frac{1}{10}},\frac{1}{k+\frac{1}{10}})$$
$$\mu_2|\mathbf{y},\mu_1,k \sim N(\frac{\sum_{q=k+1}^{n} y_q}{n-k+\frac{1}{10}},\frac{1}{n-k+\frac{1}{10}})$$

(c)

we use the function provided in the assignments sheet and let me give an explanation why it works,

In the part of code: log(c(1:(floor((n-1)/2)),n-(ceiling((n/2)+1):n-1))), it allocated each k corresponding prior without normalising constant c, also this is same as the k and $n-k$ in $f(k|\mu_1,\mu_2,\mathbf{y})$ we derived above refers to different range of k.

then in the for loop part, the code aims to give each corresponding k $log(\frac{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2})}{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_2)^2}{2})})$. This computation is alternate to $log(\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2}))$ because here we have fixed μ_1 and μ_2, and observed data $\mathbf{y}$, then $\prod_{q=1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})$ is a constant, we can assume $K = \prod_{q=1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})$, then,

$$\frac{1}{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_2)^2}{2})} = \frac{\prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})}{K}$$

so

$$\frac{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2})}{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_2)^2}{2})} = \frac{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})}{K}$$

$$log(\frac{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2})}{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_2)^2}{2})}) = log(\frac{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})}{K})$$

$$log(\frac{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2})}{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_2)^2}{2})}) = log(\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2})) - log(K)$$

$$e^{log(\frac{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2})}{\prod_{p=1}^{k} exp(-\frac{(y_p-\mu_2)^2}{2})})} = \prod_{p=1}^{k} exp(-\frac{(y_p-\mu_1)^2}{2}) \prod_{q=k+1}^{n} exp(-\frac{(y_q-\mu_2)^2}{2}) \cdot \frac{1}{K}$$

for the $\frac{1}{K}$ in this form, we can normalise it in the later steps because it is a multiplier.

And we minus max(logpost.k) in order to avoid NA output here for numerical reasons.

Lastly, normalise by post = post / sum (post) to get the exact posterior distribution of k. And use sample(n-1,size =1,prob=post) to sample from this distribution. So this function exactly this function returns samples from the distribution which we derived in part b.

```
gibbs.k = function(mu1,mu2,y){
n = length (y)

logprior.k = log(c(1:(floor((n-1)/2)),n-(ceiling((n/2)+1):n-1)))
logpost.k = logprior.k

cr = 0
for (i in 1:(n-1)){
   cr = cr + dnorm (y[i], mean = mu1 , sd =1 , log = TRUE )
   cr = cr - dnorm (y[i], mean = mu2 , sd =1 , log = TRUE )
   logpost.k[i] = logpost.k[i] + cr
}

post = exp(logpost.k - max(logpost.k))
post = post / sum (post)

sample(n-1 , size =1 , prob = post )
}
```

```r
pmix.gibbs=function(k,mu1,mu2,y,t){
  n = length (y)
  r=array(NA,c(t+1,3))
  r[1,]=c(k,mu1,mu2)
  for (i in 1:t) {
    r[i+1,2]=rnorm(1,mean=(sum(y[1:r[i,1]])/(r[i,1]+0.1)),sd=(1/(r[i,1]+0.1)))
    r[i+1,3]=rnorm(1,mean=(sum(y[(r[i,1]+1):n])/(n-r[i,1]+0.1)),sd=(1/(n-r[i,1]+0.1)))
    r[i+1,1]=gibbs.k(r[i+1,2],r[i+1,3],y)
  }
  r
}
```

(d)

i.

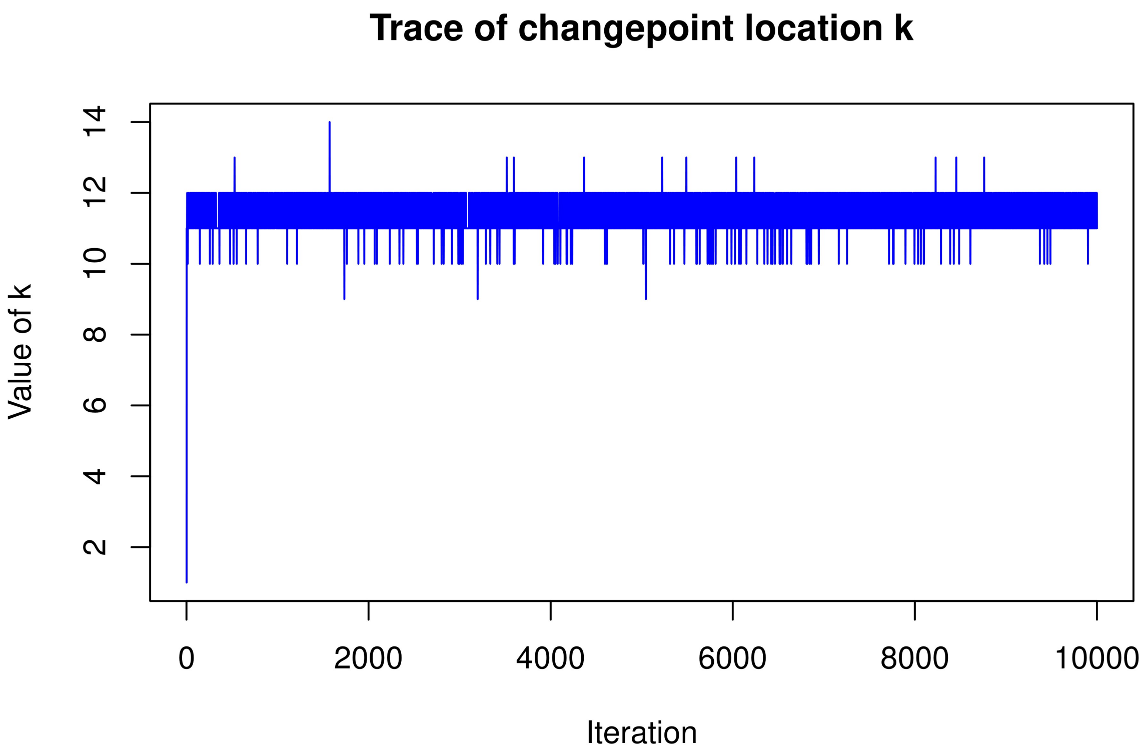
Run Gibbs sampler on the first data series with initial values $X^{(0)} = (k^{(0)}, \mu_1^{(0)}, \mu_2^{(0)}) = (1, 1, 1)$ and generate 10000 samples,

```r
l1=pmix.gibbs(1,1,1,dataset1,10000)
```

ii.

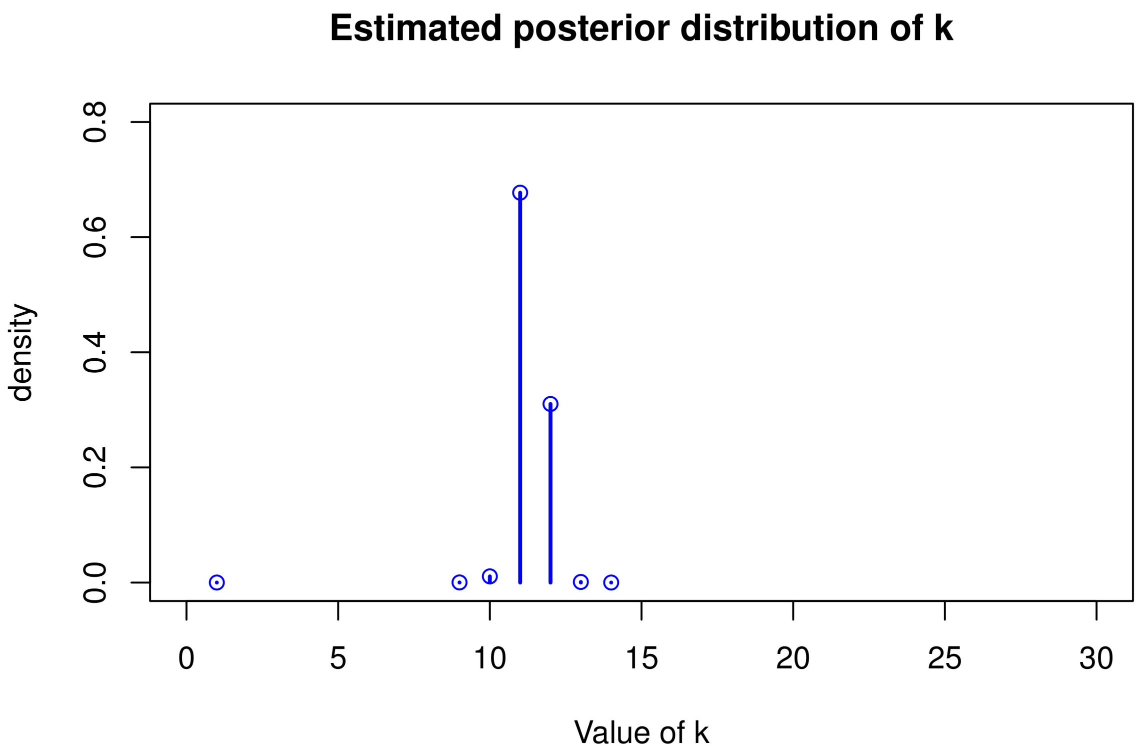
First, we look at the trace of changepoint location for k,

```r
plot(l1[,1],type = 'l',col='blue',main = 'Trace of changepoint location k',
     xlab = 'Iteration',ylab = 'Value of k')
```



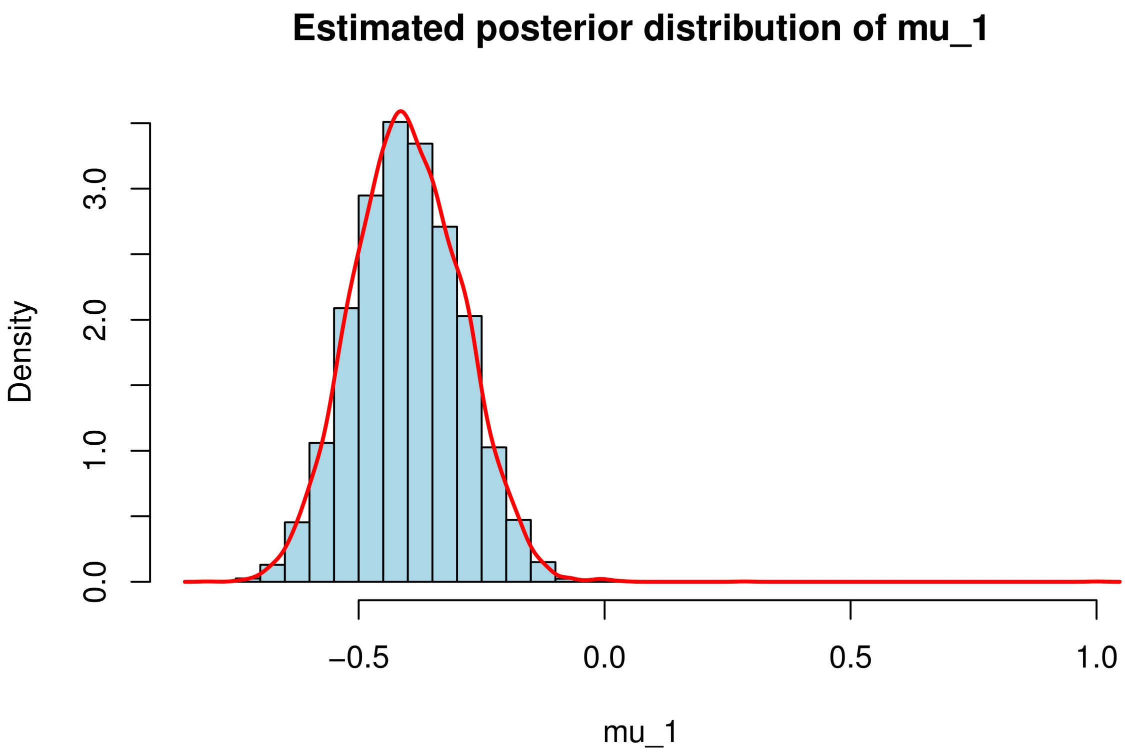
Also, we can plot the estimated posterior distribution of k,

```
dk1=as.data.frame(table(l1[,1]))
plot(as.vector(dk1$Var1),as.vector(dk1$Freq/sum(dk1$Freq)),type='h',
     xlim=c(0,30),ylim = c(0,0.8),lwd=2,col='blue',
     main = 'Estimated posterior distribution of k',xlab = 'Value of k',
     ylab = 'density')
points(as.vector(dk1$Var1),as.vector(dk1$Freq/sum(dk1$Freq)),col='blue',lwd=1)
```

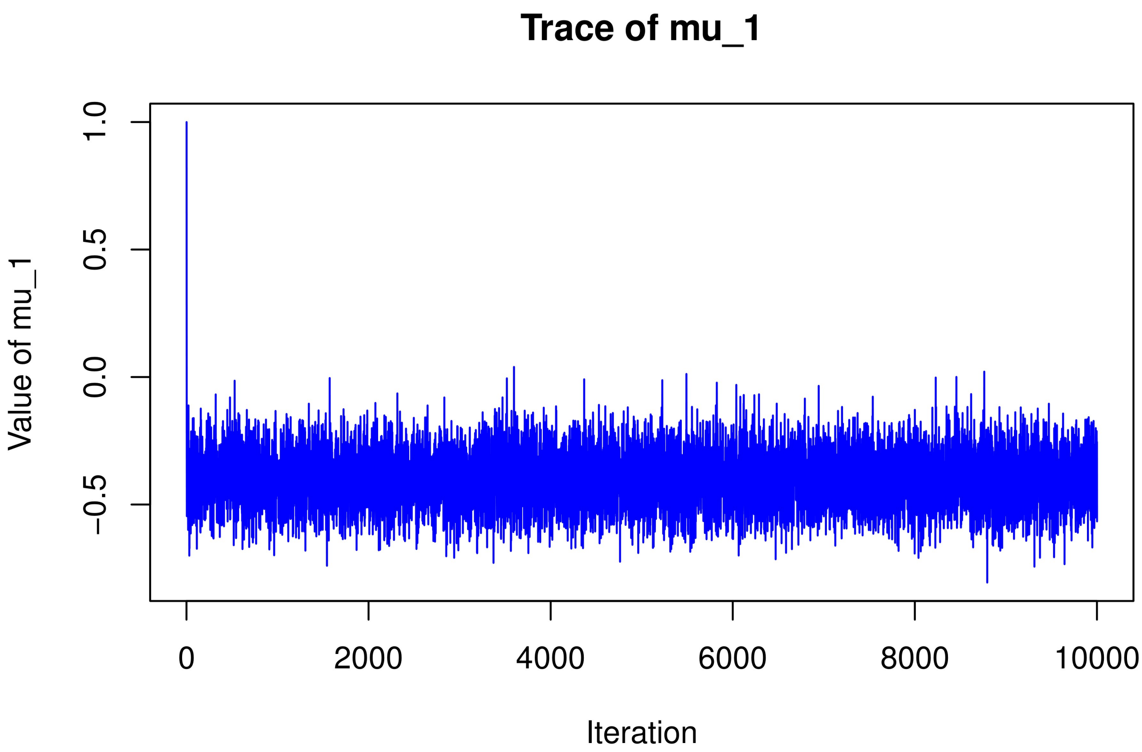


Next, let us visualize the density plot and trace plot of μ_1 and μ_2,

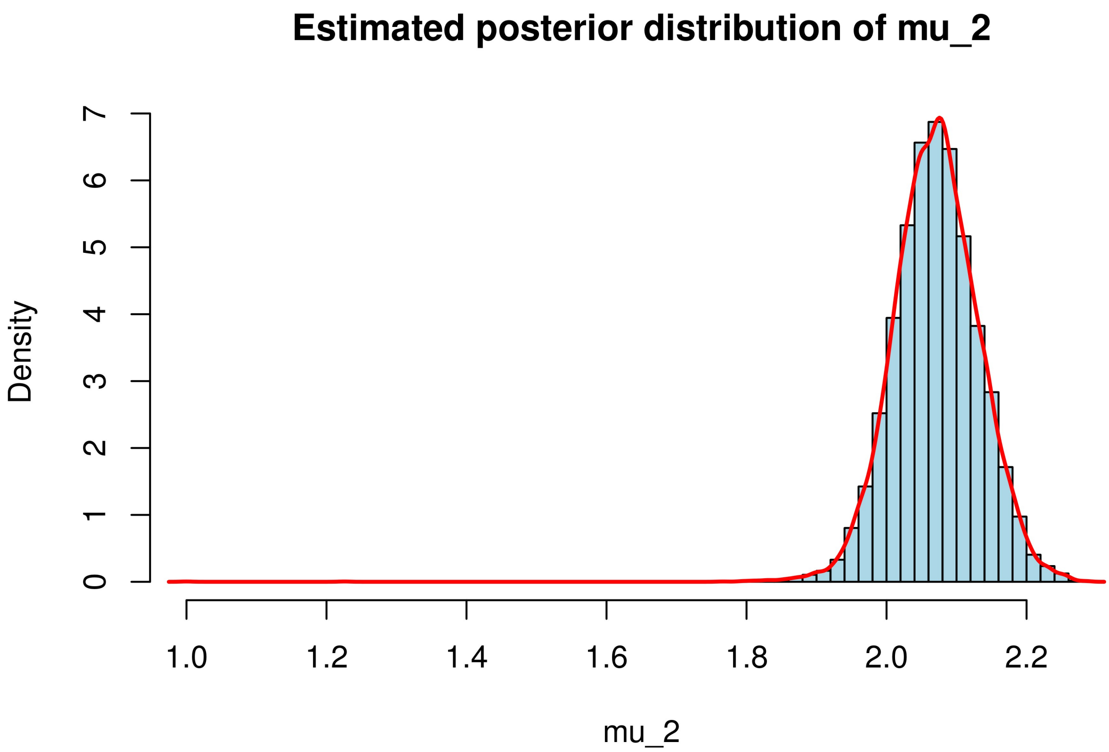
```
hist(l1[,2],freq = FALSE, ,breaks=50, col = 'lightblue',
     main = 'Estimated posterior distribution of mu_1',xlab = 'mu_1')
lines(density(l1[,2]),col='red',lwd=2)
```



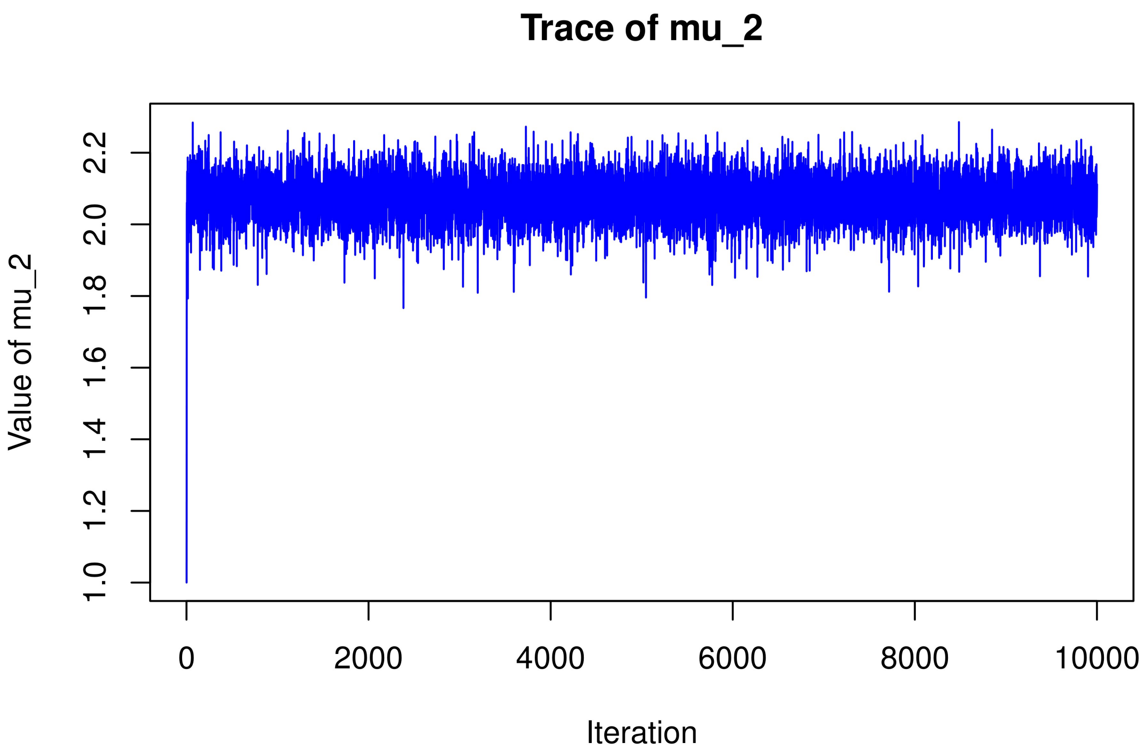

```
plot(l1[,2], col = 'blue',main = 'Trace of mu_1',xlab = 'Iteration',
     ylab = 'Value of mu_1',type='l')
```




```
hist(l1[,3],freq = FALSE, ,breaks=50, col = 'lightblue',
     main = 'Estimated posterior distribution of mu_2',xlab = 'mu_2')
lines(density(l1[,3]),col='red',lwd=2)
```




```
plot(l1[,3], col = 'blue',main = 'Trace of mu_2',xlab = 'Iteration',
     ylab = 'Value of mu_2',type='l')
```



Then, we calculate the effective sample size of our output using the autocorrelation of the changepoint location,

```
#get the autocorrelation of changepoint location k
auto_cf_l1_k = acf(l1[,1],plot = FALSE)
#get the first order sample autocorrelation of changepoint location k
acf1 = auto_cf_l1_k$acf[2]
acf1
```

```
## [1] 0.05885733
```

so we can obtain that $\rho(X_1^{(t-1)}, X_1^{(t)}) = 0.05868507$, the by the formula of ESS

$$T_{ESS} = \frac{1-\rho}{1+\rho} \cdot T$$

```
T_ESS = round(10000*((1-acf1)/(1+acf1)))
print(paste("Thus the effective sample size is",T_ESS))
```

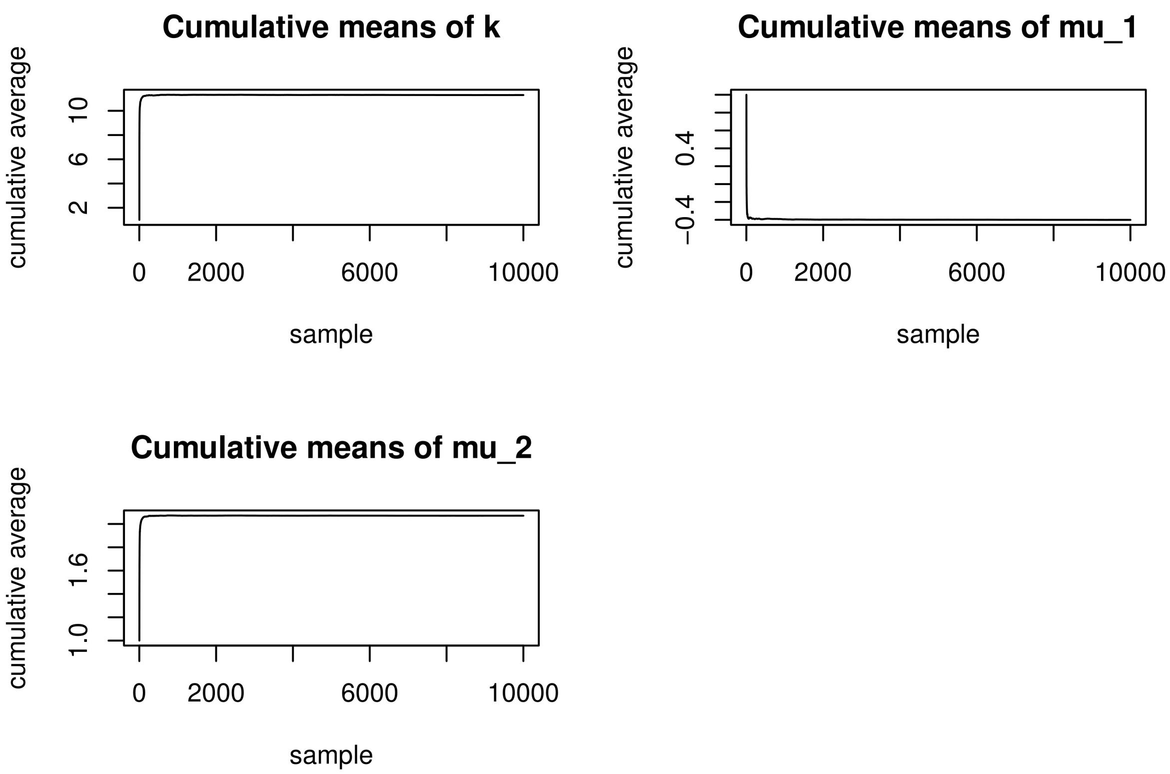
```
## [1] "Thus the effective sample size is 8888"
```

This ESS size is big enough to converge. Therefore, we don't need to run the sampler again for a longer period because the effective sample size is high. This means comparing to i.i.d. sampling, there are not much information loss in the sample from the Markov Chain.

Alternately, we can assess the convergence by cumulative mean average, and the plots showed that the average converges very fast.

```r
par(mfrow = c(2, 2))
cum_avg1 <- cummean(l1[,1])
cum_avg2 <- cummean(l1[,2])
cum_avg3 <- cummean(l1[,3])

plot(cum_avg1,type='l',xlab = 'sample',ylab = 'cumulative average',
     main = 'Cumulative means of k')
plot(cum_avg2,type='l',xlab = 'sample',ylab = 'cumulative average',
     main = 'Cumulative means of mu_1')
plot(cum_avg3,type='l',xlab = 'sample',ylab = 'cumulative average',
     main = 'Cumulative means of mu_2')
```



iii.

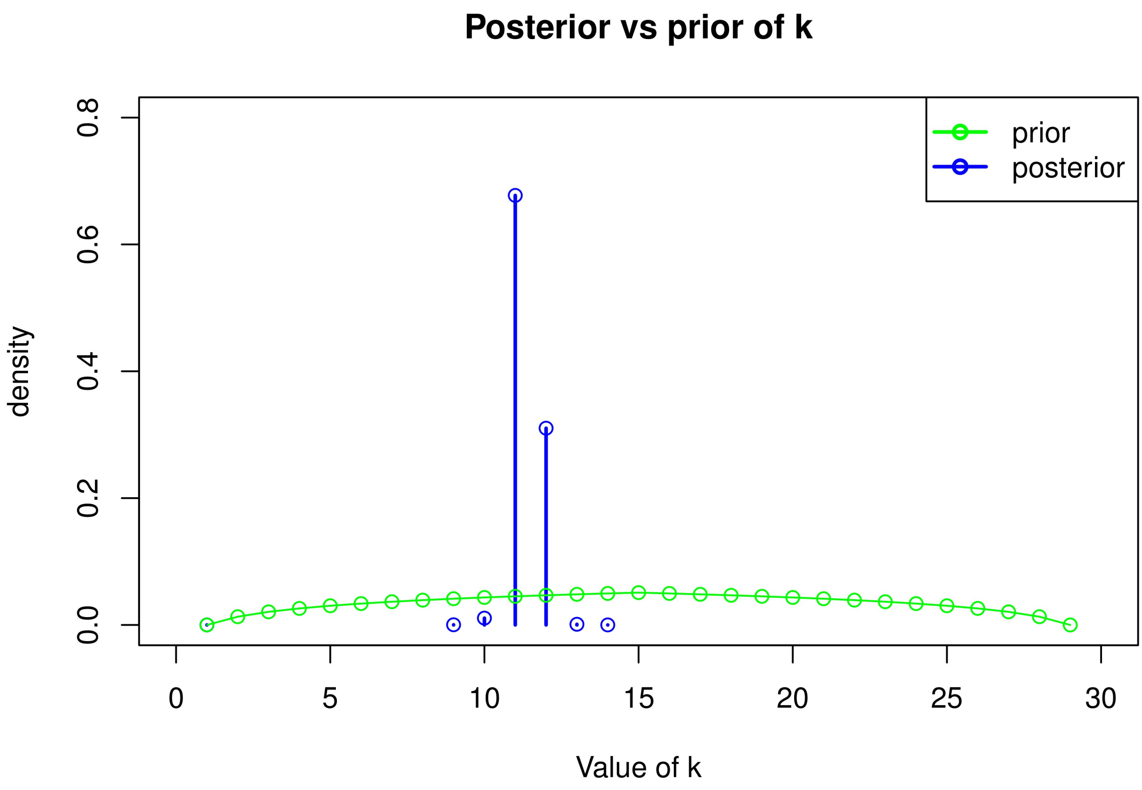
Compare the prior and estimated posterior distributions of the three parameters of interest (k, μ_1, μ_2).

First, write a function to get the prior distribution of k,

```r
prior.k=function(y){
  n=length(y)
  logprior.k = log(c(1:(floor((n-1)/2)),n-(ceiling((n/2)+1):n-1)))
  prior=logprior.k/sum(logprior.k)
  prior
}
```

then plot the posterior vs prior of k, the prior of k is also discrete but we plot it as a line with points in order to visualize more clearly,

```
plot(as.vector(dk1$Var1),as.vector(dk1$Freq/sum(dk1$Freq)),type='h',
     xlim=c(0,30),ylim = c(0,0.8),lwd=2,col='blue',
     main = 'Posterior vs prior of k',xlab = 'Value of k',ylab = 'density')
points(as.vector(dk1$Var1),as.vector(dk1$Freq/sum(dk1$Freq)),col='blue',lwd=1)
lines(prior.k(dataset1),col='green')
points(prior.k(dataset1),col='green')
legend(x = "topright",
       legend = c("prior", "posterior"),
       pch = c(1,1),
       col = c("green", "blue"),
       lwd = 2)
```



plot the posterior vs prior of μ_1,

```
#generate prior of mu_1
y=rnorm(10000,0,sqrt(10))

# Build dataset with different distributions
data1_1 <- data.frame(
  type = c( rep("posterior", 10001), rep("prior", 10000) ),
  value = c(l1[,2], y)
)
```