HW3

Jiang Bian & Haoyu Yin

01 March 2018

Q1

a

For $p_{ij}^{(k+1)}$, based on the lecture notes, we have the following:

$$Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(k)}) = \sum_{i=1}^{n}\sum_{j=1}^{m} p(z_{ij}|y_i, \boldsymbol{\Psi}^{(k)}) \cdot \ln p(z_{ij} = 1, y_i|\boldsymbol{\Psi}^{(k)})$$

denote $p_{ij} = p(z_{ij} = 1|y_i, \boldsymbol{\Psi})$, then by Bayes Rule, we can compute

$$p_{ij}^{(k+1)} = \frac{p(z_{ij} = 1, y_i|\boldsymbol{\Psi}^{(k)})}{p(y_i|\boldsymbol{\Psi}^{(k)})} = \frac{p(z_{ij} = 1, y_i|\boldsymbol{\Psi}^{(k)})}{\sum_{j=1}^{m} p(z_{ij} = 1, y_i|\boldsymbol{\Psi}^{(k)})}$$

And

$$\begin{aligned} p(z_{ij} = 1, y_i|\boldsymbol{\Psi}^{(k)}) &= p(z_{ij} = 1|\boldsymbol{\Psi}^{(k)}) \cdot p(y_i|z_{ij} = 1, \boldsymbol{\Psi}^{(k)}) \\ &= \pi_j^{(k)} \cdot \phi(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)}; 0, \sigma^2) \end{aligned}$$

Thus we have

$$p_{ij}^{(k+1)} = \frac{\pi_j^{(k)} \cdot \phi(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)}; 0, \sigma^{2^{(k)}})}{\sum_{j=1}^{m} \pi_j^{(k)} \phi(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)}; 0, \sigma^{2^{(k)}})}$$

b

From a, we have

$$\begin{aligned} Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(k)}) &= \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} \cdot \ln p(z_{ij} = 1, y_i|\boldsymbol{\Psi}^{(k)}) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} \cdot \ln\,[\pi_j^{(k)} \cdot \phi(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)}; 0, \sigma^2)] \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} \ln \pi_j^{(k)} + \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} \ln\,(\frac{1}{\sqrt{2\pi} \cdot \sigma}) + \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} [-\frac{(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)})^2}{2\sigma^2}] \end{aligned}$$

Therefore, we can set:

$$I_1 = \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} \ln \pi_j^{(k)}$$
$$I_2 = \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} \ln (\frac{1}{\sqrt{2\pi}\cdot\sigma})$$
$$I_3 = \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} [-\frac{(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)})^2}{2\sigma^2}]$$

As $\sum_{j=1}^{m} \pi_j = 1$, we can construct the Lagrangian Function:

$$L(\pi_1^{(k)}, ..., \pi_m^{(k)}; \lambda) = Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(k)}) - \lambda(\sum_{j=1}^{m} \pi_j^{(k)} - 1)$$

Set $\frac{\partial L}{\partial \pi_j^{(k)}} = 0$ with $(j = 1, 2, ..., m)$, we have

$$\sum_{i=1}^{n} p_{ij}^{(k+1)} \frac{1}{\pi_j^{(k+1)}} - \lambda = 0 \ (j = 1, 2, ..., m)$$
$$\Rightarrow \pi_j = \frac{\sum_{i=1}^{n} p_{ij}^{(k+1)}}{\lambda}$$
$$\Rightarrow \sum_{j=1}^{m} \pi_j = \frac{\sum_{j=1}^{m}\sum_{i=1}^{n} p_{ij}^{(k+1)}}{\lambda} = \frac{n}{\lambda} = 1$$
$$\Rightarrow \lambda = n$$

Thus we have $\pi_j^{(k+1)} = \frac{\sum_{i=1}^{n} p_{ij}^{(k+1)}}{n}$, which is **(2.a)**

Recall the equation above, only I_3 contains $\boldsymbol{\beta}_j^{(k)}$, so in order to maximize I_3, we can minimize $I_3^* = \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)}(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)})^2$, and $I_{3j}^* = \sum_{i=1}^{n} p_{ij}^{(k+1)} \cdot (y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)})^2$ $(j = 1, 2, ..., m)$
In order to minimize $I_{3j}^{(*)}$, we can solve:

$$\frac{\partial I_{3j}^{(*)}}{\partial \boldsymbol{\beta}_j} = 0$$
$$\iff 2\sum_{i=1}^{n} p_{ij}^{(k+1)} \mathbf{x}_i (y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k+1)}) = 0$$
$$\iff \sum_{i=1}^{n} p_{ij}^{(k+1)} \mathbf{x}_i y_i = \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k+1)} p_{ij}^{(k+1)}$$
$$\iff \boldsymbol{\beta}_j^{(k+1)} = (\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T p_{ij}^{(k+1)}))^{-1} \cdot (\sum_{i=1}^{n} \mathbf{x}_i p_{ij}^{(k+1)} y_i), \ j = 1, ..., m$$

which is **(2.b)**

Last, only I_2 and I_3 contain σ^2, and if we denote $I_2^* = \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} \ln \sigma^2$, also denote $I_3^{**} = \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}^{(k+1)} [\frac{(y_i - \mathbf{x}_i^T \beta_j^{(k)})^2}{\sigma^2}]$.

$$s_j = \sum_{i=1}^{n} p_{ij}^{(k+1)} \ln \sigma^2 + \sum_{i=1}^{n} p_{ij}^{(k+1)} [\frac{(y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k)})^2}{\sigma^2}]$$

Thus we only need to find σ^2 to minimize each s_j where s_j $(j = 1, 2, ..., m)$. Now that $\mathbf{x}_i^T \boldsymbol{\beta}_j^{(k+1)}$ is equal to the weighted mean of y_i, to minimize s_j, also from the property of sample variance, σ^2 must be the sample variance of the weighted sample $y_1, y_2, ..., y_n$.
Therefore,

$$\sigma_j^{2(k+1)} = \frac{\sum_{i=1}^{n} p_{ij}^{(k+1)} (y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k+1)})^2}{\sum_{i=1}^{n} p_{ij}^{(k+1)}}$$

According to the given condition, $\sigma_1^{2(k+1)} = \sigma_2^{2(k+1)} = ... = \sigma_m^{2(k+1)} \equiv \sigma^{2(k+1)}$
So we have:

$$\begin{aligned}
&\sigma^{2(k+1)} \sum_{i=1}^{n} p_{ij}^{(k+1)} = \sum_{i=1}^{n} p_{ij}^{(k+1)} (y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k+1)})^2 \\
\iff\ &\sigma^{2(k+1)} \sum_{j=1}^{m} \sum_{i=1}^{n} p_{ij}^{(k+1)} = \sum_{j=1}^{m} \sum_{i=1}^{n} p_{ij}^{(k+1)} (y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k+1)})^2 \\
\iff\ &\sigma^{2(k+1)} = \frac{\sum_{j=1}^{m} \sum_{i=1}^{n} p_{ij}^{(k+1)} (y_i - \mathbf{x}_i^T \boldsymbol{\beta}_j^{(k+1)})^2}{n}
\end{aligned}$$

Which is **(2.c)**

Q2

a

Known $g(x) \propto (2x^{\theta-1} + x^{\theta-1/2})e^{-x}$, $x > 0$.
And the constant C such that $C \int_0^{\infty} (2x^{\theta-1} + x^{\theta-1/2})e^{-x} dx = 1$.

We can solve C as:
$2C \cdot \Gamma(\theta) + C \cdot \Gamma(\theta + \frac{1}{2}) = 1$, and receive $C = \frac{1}{2 \cdot \Gamma(\theta) + \Gamma(\theta + \frac{1}{2})}$
Therefore,

$$\begin{aligned}
g(x) &= \frac{1}{2\Gamma(\theta) + \Gamma(\theta + \frac{1}{2})} \cdot 2x^{\theta-1} e^{-x} + \frac{1}{2\Gamma(\theta) + \Gamma(\theta + \frac{1}{2})} \cdot x^{\theta-1/2} e^{-x} \\
&= \frac{2\Gamma(\theta)}{2\Gamma(\theta) + \Gamma(\theta + \frac{1}{2})} \cdot \frac{1}{\Gamma(\theta)} x^{\theta-1} e^{-x} + \frac{\Gamma(\theta + \frac{1}{2})}{2\Gamma(\theta) + \Gamma(\theta + \frac{1}{2})} \cdot \frac{1}{\Gamma(\theta + \frac{1}{2})} x^{\theta-1/2} e^{-x}
\end{aligned}$$

So, g is a mixture of Gamma(θ,1) and Gamma($\theta + \frac{1}{2}$,1) with the weights $\frac{2\Gamma(\theta)}{2\Gamma(\theta)+\Gamma(\theta+\frac{1}{2})}$ and $\frac{\Gamma(\theta+\frac{1}{2})}{2\Gamma(\theta)+\Gamma(\theta+\frac{1}{2})}$.

b

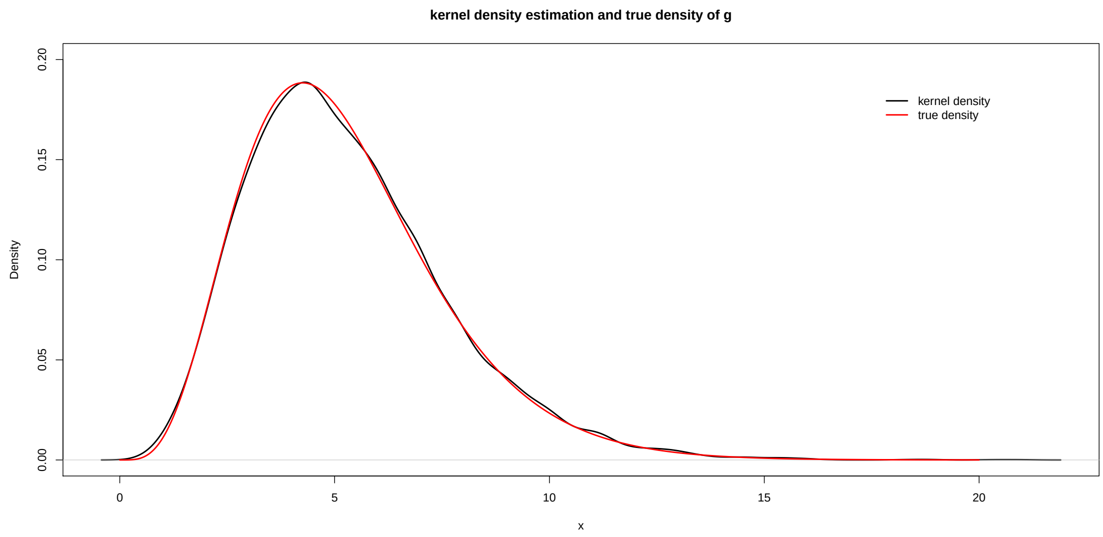
pseudo code
Input θ, n,
let $c1 \leftarrow \frac{2\Gamma(\theta)}{2\Gamma(\theta)+\Gamma(\theta+\frac{1}{2})}$;
$x \leftarrow x \sim Unif(0, 1)$;
for i from 1 to n,
if $x_i < c1$, *then*
$x_i \leftarrow Gamma(\theta, 1)$;
if $x_i \geq c1$, *then*
$x_i \leftarrow Gamma(\theta + \frac{1}{2}, 1)$;
Output x.

The R function is shown below:

```
code2b <- function(theta,n){
  c1 <- (2*gamma(theta))/(2*gamma(theta)+gamma(theta+1/2))
  x <- runif(n,0,1)
  for (i in 1:n){
    if (x[i]<c1){
      x[i] <- rgamma(1, shape = theta, scale = 1)
    }
    else{
      x[i] <- rgamma(1, shape = theta+0.5, scale = 1)
    }
  }
  x
}
gt <- function(x,theta){
  1/(2*gamma(theta)+gamma(theta+0.5))*(2*x^(theta-1)+x^(theta-0.5))*exp(-x)
}
```

Suppose $\theta = 5$ and $n = 10000$, we can plot the kernel density estimates and true density below:

```
plot(density(code2b(5,10000)), ylim = c(0, 0.2), xlab = c("x"),
     main = c("kernel density estimation and true density of g"),lty = 1,lwd = 2)
curve(gt(x,5), from = 0, to = 20, n = 10000, add = T, col="red",lty = 1,lwd = 2)
legend("topright", inset=.1,
       legend = c("kernel density","true density"),
       bty = "n", lty = 1, lwd = 2, col = c("black", "red"))
```



c

Known that $\sqrt{(4+x)}x^{\theta-1}e^{-x} \le (2+\sqrt{x})x^{\theta-1}e^{-x} = (2x^{\theta-1}+x^{\theta-1/2})e^{-x}$.
Using rejection sampling to sample from f using g as the instrumental distribution.
We can design a **pseudo-code** as following:
Input θ, n;
let $c1 \leftarrow \frac{2\Gamma(\theta)}{2\Gamma(\theta)+\Gamma(\theta+\frac{1}{2})}$;
y as an empty set;
for i from 1 to n,
$while\ process = TRUE$,
$x \leftarrow Unif(0,1)$;
$if\ x < c1,\ then$
$x \leftarrow Gamma(\theta,1)$;
$if\ x \ge c1,\ then$
$x \leftarrow Gamma(\theta+\frac{1}{2},1)$;
$test \leftarrow Unif(0,1)\cdot(2x^{\theta-1}+x^{\theta-1/2})e^{-x}$;
$if\ test \le \sqrt{(4+x)}x^{\theta-1}e^{-x}$,
$y_i \leftarrow x$
$process = FALSE$;
Output y.

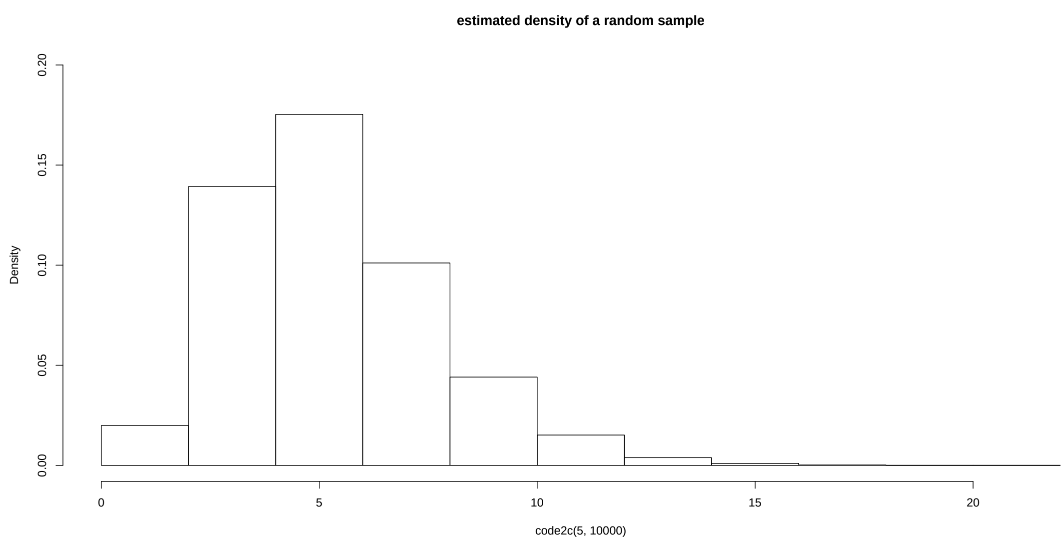
The R function is shown below:

```
code2c <- function(theta,n){
  c1 <- (2*gamma(theta))/(2*gamma(theta)+gamma(theta+1/2))
  y <- rep(0,n)
  for (i in 1:n){
    process <- TRUE
    while(process){
      x <- runif(1,0,1)
      if (x<c1){
```

```
        x <- rgamma(1, shape = theta, scale = 1)
      }
      else{
        x <- rgamma(1, shape = theta+0.5, scale = 1)
      }
      test <- runif(1,0,1) * (2*x^(theta-1)+x^(theta-1/2))*exp(-x)
      if (test<=(sqrt(4+x)*x^(theta-1)*exp(-x))){
        y[i] <- x
        process <- FALSE
      }
    }
  }
  y
}
```

Suppose $\theta = 5$ and $n = 10000$, we can plot estimated density of a random sample below:

```
hist(code2c(5,10000), prob=TRUE, ylim = c(0, 0.2),
     main = c("estimated density of a random sample"))
```



Q3

a

Known that $f(x) \propto \frac{x^{\theta-1}}{1+x^2} + \sqrt{2+x^2}(1-x)^{\beta-1}$, $0 < x < 1$.
Therefore $\frac{x^{\theta-1}}{1+x^2} + \sqrt{2+x^2}(1-x)^{\beta-1} \leq x^{\theta-1} + \sqrt{3}(1-x)^{\beta-1}$,
Set $g(x) \propto x^{\theta-1} + \sqrt{3}(1-x)^{\beta-1}$ and $g(x) = Cg^*(x) = x^{\theta-1} + \sqrt{3}(1-x)^{\beta-1}$,
Thus $C\int_0^1 g^*(x)dx = 1 \Rightarrow C = \frac{1}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}$,
Then,

$$
\begin{aligned}
g(x) &= \frac{1}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}(x^{\theta-1}+\sqrt{3}(1-x)^{\beta-1}) \\
&= \frac{B(\theta,1)}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}\frac{x^{\theta-1}}{B(\theta,1)}+\frac{B(1,\beta)}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}\sqrt{3}\frac{(1-x)^{\beta-1}}{B(1,\beta)} \\
&= \frac{1/\theta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}\frac{x^{\theta-1}}{B(\theta,1)}+\frac{1/\beta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}\sqrt{3}\frac{(1-x)^{\beta-1}}{B(1,\beta)}
\end{aligned}
$$

So, it is a mixture of Beta(θ,1) and Beta(1,β) with the weights $\frac{1/\theta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}$ and $\frac{\sqrt{3}/\beta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}$.

We can design a **pseudo-code** as following:
Input θ, β, n;
let $c1 \leftarrow \frac{1/\theta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}$;
y as an empty set;
for i from 1 to n,
$while\ process = TRUE$,
$x \leftarrow Unif(0,1)$;
$if\ x < c1,\ then$
$x \leftarrow Beta(\theta,1)$;
$if\ x \geq c1,\ then$
$x \leftarrow Beta(1,\beta)$;
$test \leftarrow Unif(0,1)\cdot(2x^{\theta-1}+x^{\theta-1/2})e^{-x}$;
$if\ test \leq (x^{\theta-1}+\sqrt{3}*(1-x)^{\beta-1})$,
$y_i \leftarrow x$
$process = FALSE$;
Output y.

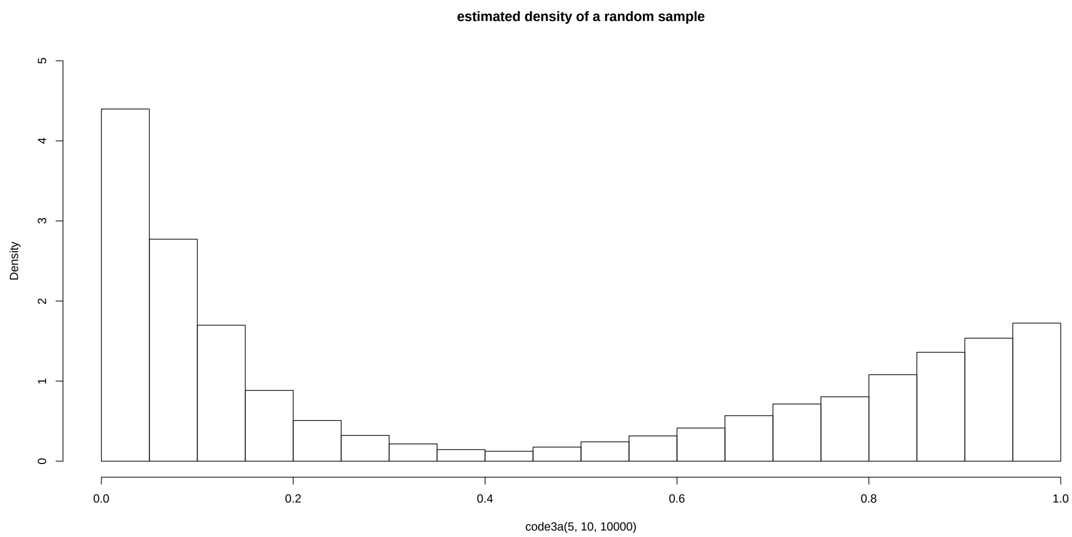
The R function is shown below:

```
code3a <- function(theta,beta,n){
  c1 <- 1/theta/(1/theta+sqrt(3)/beta)
  y <- rep(0,n)
  for (i in 1:n){
    process <- TRUE
    while(process){
      x <- runif(1,0,1)
      if (x<c1){
        x <- rbeta(1,theta,1)
      }
      else{
        x <- rbeta(1,1,beta)
      }
      test <- runif(1,0,1) * (x^(theta-1)+sqrt(3)*(1-x)^(beta-1))
      if (test<=(x^(theta-1)/(1+x^2)+sqrt(2+x^2)*(1-x)^(beta-1))){
        y[i] <- x
        process <- FALSE
      }
```

```
    }
  }
  y
}
```

Suppose $\theta = 5$, $\beta = 10$ and $n = 10000$, we can graph the estimated density of a random sample below:

```
hist(code3a(5,10,10000),xlim = range(0:1), ylim = range(0:5),prob=TRUE,
     main = c("estimated density of a random sample"))
```



b

Define $g_1(x) = x^{\theta-1}$, $g_2(x) = \sqrt{3}(1-x)^{1-\beta}$,
$p_1 \int_0^1 g_1(x)dx = 1$ and $p_2 \int_0^1 g_2(x)dx = 1 \Rightarrow p_1 = \theta, p_2 = \frac{\beta}{\sqrt{3}}$
$\int_0^1 g_1(x)dx = \frac{1}{p_1}$, $\int_0^1 g_2(x)dx = \frac{1}{p_2}$
$c_1 = \frac{\frac{1}{p_1}}{\frac{1}{p_1}+\frac{1}{p_2}} = \frac{p_2}{p_1+p_2} = \frac{1/\theta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}$
And $c_2 = \frac{\frac{1}{p_2}}{\frac{1}{p_1}+\frac{1}{p_2}} = \frac{p_1}{p_1+p_2} = \frac{\sqrt{3}/\beta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}$

We can design a **pseudo-code** as following:
Input θ, β, n;
let $c1 \leftarrow \frac{1/\theta}{\frac{1}{\theta}+\frac{\sqrt{3}}{\beta}}$;
y as an empty set;
for i from 1 to n,
$while\ process = TRUE$,
$x \leftarrow Unif(0,1)$;
$if\ x < c1,\ then$
$x \leftarrow Beta(\theta, 1)$,
$test \leftarrow Unif(0,1) \cdot x^{\theta-1}$;

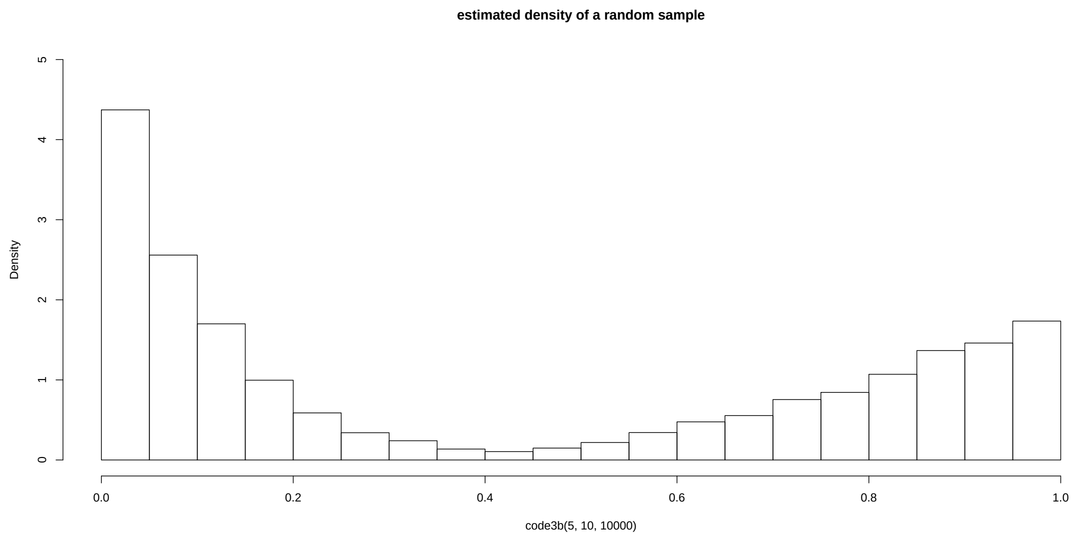
$if\ test \leq x^{theta-1}/(1+x^2)$,
$y_i \leftarrow x$
$process = FALSE$;
$if\ x \geq c1,\ then$
$x \leftarrow Beta(1,\beta)$;
$test \leftarrow Unif(0,1) \cdot \sqrt{3} * (1-x)^{\beta-1}$;
$if\ test \leq \sqrt{2+x^2} * (1-x)^{\beta-1}$,
$y_i \leftarrow x$
$process = FALSE$;
Output y.

The R function is shown below:

```
code3b <- function(theta,beta,n){
  c1 <- 1/theta/(1/theta+sqrt(3)/beta)
  y <- rep(0,n)
  for (i in 1:n){
    process <- TRUE
    while(process){
      x <- runif(1,0,1)
      if (x<c1){
        x <- rbeta(1,theta,1)
        test <- runif(1,0,1)*(x^(theta-1))
        if (test<=(x^(theta-1)/(1+x^2))){
          y[i] <- x
          process <- FALSE
        }
      }
      else{
        x <- rbeta(1,1,beta)
        test <- runif(1,0,1)*(sqrt(3)*(1-x)^(beta-1))
        if (test<=(sqrt(2+x^2)*(1-x)^(beta-1))){
          y[i] <- x
          process <- FALSE
        }
      }
    }
  }
  y
}
```

Suppose $\theta = 5$, $\beta = 10$ and $n = 10000$, we can graph the estimated density of a random sample below:

```
hist(code3b(5,10,10000),xlim = range(0:1), ylim = range(0:5),prob=TRUE,
     main = c("estimated density of a random sample"))
```


estimated density of a random sample
Density
0
1
2
3
4
5
0.0
0.2
0.4
0.6
0.8
1.0
code3b(5, 10, 10000)