Computational Statistics Lab3 Group 29

Jin Yan (jinya425), Yaning Wang (yanwa579)

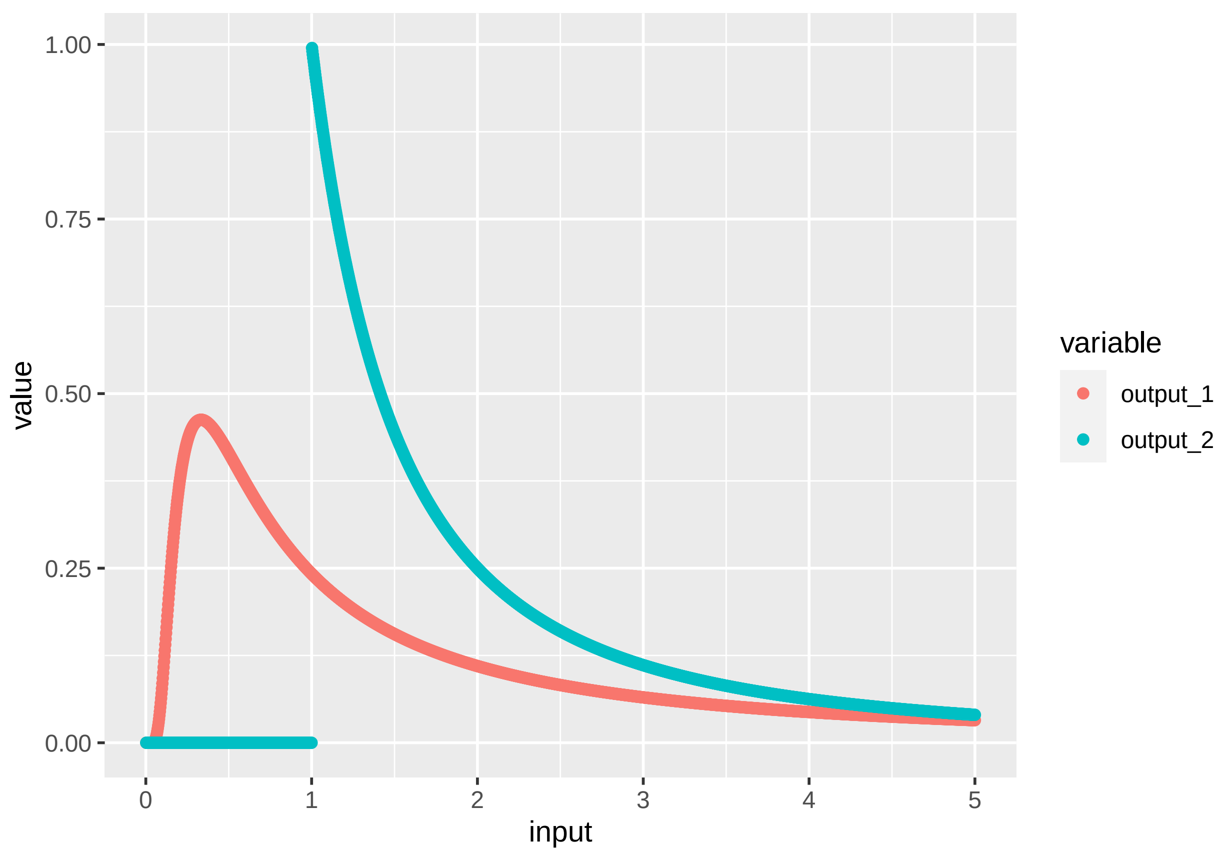
2022-11-27

Question 1: Stable distribution

1.

The plots of f(x) and fp(x) are shown here.

The parameters are alpha = 2, c = 1, Tmin = 1 respectively.



From the plot we can see that for the fp(x),when the values of inputs are in the region between 0 and Tmin, the outputs are zero, which means we cannot only use PDF of the power-low distribution as the majorizing density. This is because fp(x) in the next steps will be used for sampling. For this purpose, when multiplied with a constant, the values of outputs in fp(x) should be larger than the outputs of inputs in f(x). Obviously, zeros cannot reach this goal. In order to derive a appropriate majorizing density, we have to make some changes. One way is to make the outputs of inputs in the region between zero and Tmin equal to the output of the Tmin in fp(x) and then make the new piecewise function into a real density function. So, we can define the function for the region between 0 to Tmin as:

$f_{part1}(x) = \frac{\alpha-1}{Tmin}$

Note that the integral of fpart1(x) in the region between 0 and Tmin is $\alpha - 1$ and obviously, as the density function the integral of fp(x) is 1. That means the sum of the integrals of the two functions is α. We have to do some adjustments. According to the properties of integrals, we can divide the two PDFs byα. In this way, the integral of them all is 1. The final function looks like this.

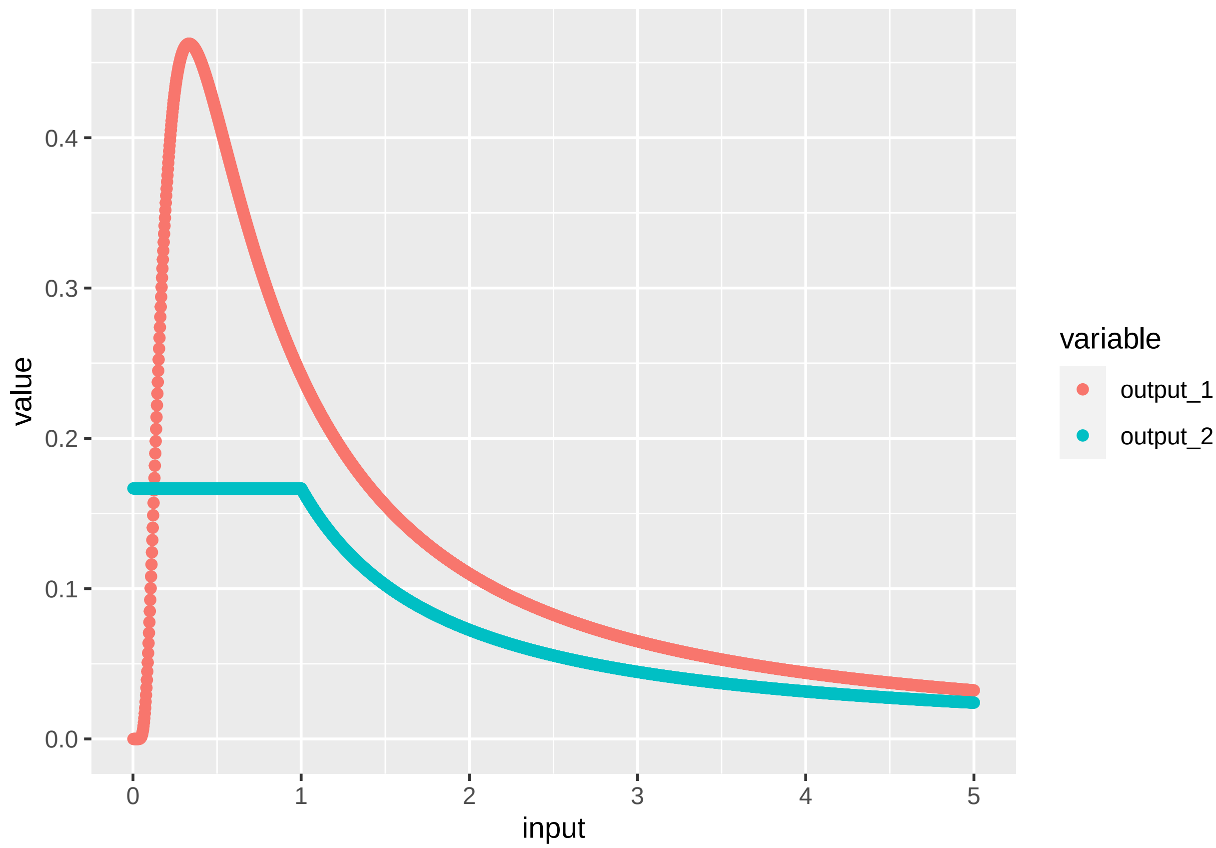
$f_{final}(x) = \frac{\alpha-1}{\alpha Tmin}$ $x \in (0, Tmin]$

$f_{final}(x) = \frac{\alpha-1}{\alpha Tmin} * \left(\frac{x}{Tmin}\right)^{-\alpha}$ $x \in (Tmin, \infty)$

The plot for the f(x) and $f_{final}(x)$ is:

The parameters are alpha = 1.2, c = 1, Tmin = 1 respectively.

The reasons for choosing these three parameters are given in the answer of second question.



2. We know if and only if $\frac{f(x)}{M*f_{final(x)}} \geq u$, the number in the sample can be accepted($u \sim Uniform(0, 1)$). For this check, we have to know M. According to the properties of M, M is positive and $0 < \frac{f(x)}{M*f_{final(x)}} \leq 1$ in the domain.This means $M \geq \frac{f(x)}{f_{final(x)}}$ in the domain. Note that $f_{final(x)}$ is a piecewise function, so $\frac{f(x)}{f_{final(x)}}$ is a piecewise function too. In order to describe much easier, we can define the first part of $\frac{f(x)}{f_{final(x)}}$ as $g_1(x)$, and the second part is $g_2(x)$. So, in the first part of domain, $M \geq g_1(x)$ and for the second, $M \geq g_2(x)$. In this way, only if the M we get is larger than or equal to any output of $g_1(x)$ and $g_2(x)$, the M is what we want. In other words, M should be larger than or equal to the bigger one between the maximum of $g_1(x)$ and maximum of $g_2(x)$. Another property we need to notice is M has to be as small as possible to increase the rate of acceptance$\frac{1}{M}$. So, the most satisfactory M is equal to the bigger one between the maximum of $g_1(x)$ and maximum of $g_2(x)$. In fact, we cannot get these maximums easily in any case. The only situation where we can get the maximums of $g_1(x)$ and $g_2(x)$ easily is that the x corresponding to the maximum of $g_1(x)$ $x \in (-\infty, \infty)$ is in the first part of the domain and the x corresponding to the maximum of $g_2(x)$ $x \in (-\infty, \infty)$ is in the second part of the domain.This is because only in this situation, can we get maximums of two functions by setting the derivatives of them to zero. Therefore, to get the concrete M, we have to choose some special values of parameters to guarantee that we can get the maximums of $g_1(x)$ and $g_2(x)$ in this way.

$g_1(x) = \frac{\alpha Tmin}{\alpha-1} * c(\sqrt{2\pi})^{-1} * exp(\frac{c^2}{-2x}) * x^{\frac{-3}{2}}$

$g_2(x) = \frac{\alpha Tmin^{1-\alpha}}{\alpha-1} * c(\sqrt{2\pi})^{-1} * exp(\frac{c^2}{-2x}) * x^{\frac{-3}{2}+\alpha}$

the derivatives of them are:

$x \in (0, Tmin]$ $\left(\frac{f(x)}{f_{final}(x)}\right)' = \frac{\alpha Tmin}{\alpha-1} * c(\sqrt{2\pi})^{-1} * exp(\frac{c^2}{-2x}) * (\frac{c^2}{2}x^{-3.5} - 1.5x^{-2.5})$

$x \in (Tmin, \infty]$$\left(\frac{f(x)}{f_{final}(x)}\right)' = \frac{\alpha Tmin^{1-\alpha}}{\alpha-1} * c(\sqrt{2\pi})^{-1} * exp(\frac{c^2}{-2x}) * (\frac{c^2}{2}x^{-3.5+\alpha} - (1.5-\alpha)x^{-2.5+\alpha})$

If we want to get the situation we mentioned above we have to get an alpha between 1 and 1.5. Only values in this region can let us get the maximum of $g_2(x)$ in the second part of domain. We also ensure c and Tmin are larger than zero. so we can let alpha = 1.2, c = 1, Tmin = 1.

When we set upper $\left(\frac{f(x)}{f_{final}(x)}\right)'$ zero, we can find the $x = \frac{1}{3}$ and the corresponding maximum is 2.775246.

When we set lower $\left(\frac{f(x)}{f_{final}(x)}\right)'$ zero, we can find the $x = \frac{5}{3}$ and the corresponding maximum is 1.521312.

Obviously, 2.775246> 1.521312, so M is 2.775246.

The integral of the first part of $f_{final}(x)$ is $\frac{\alpha-1}{\alpha}$. As for the second part, the integral is $\frac{1}{\alpha}$. Based on this, we know that if we sample from the distribution, the proportion of the numbers from first part should be $\frac{\alpha-1}{\alpha}$, and the numbers from the second part should account for $\frac{1}{\alpha}$.

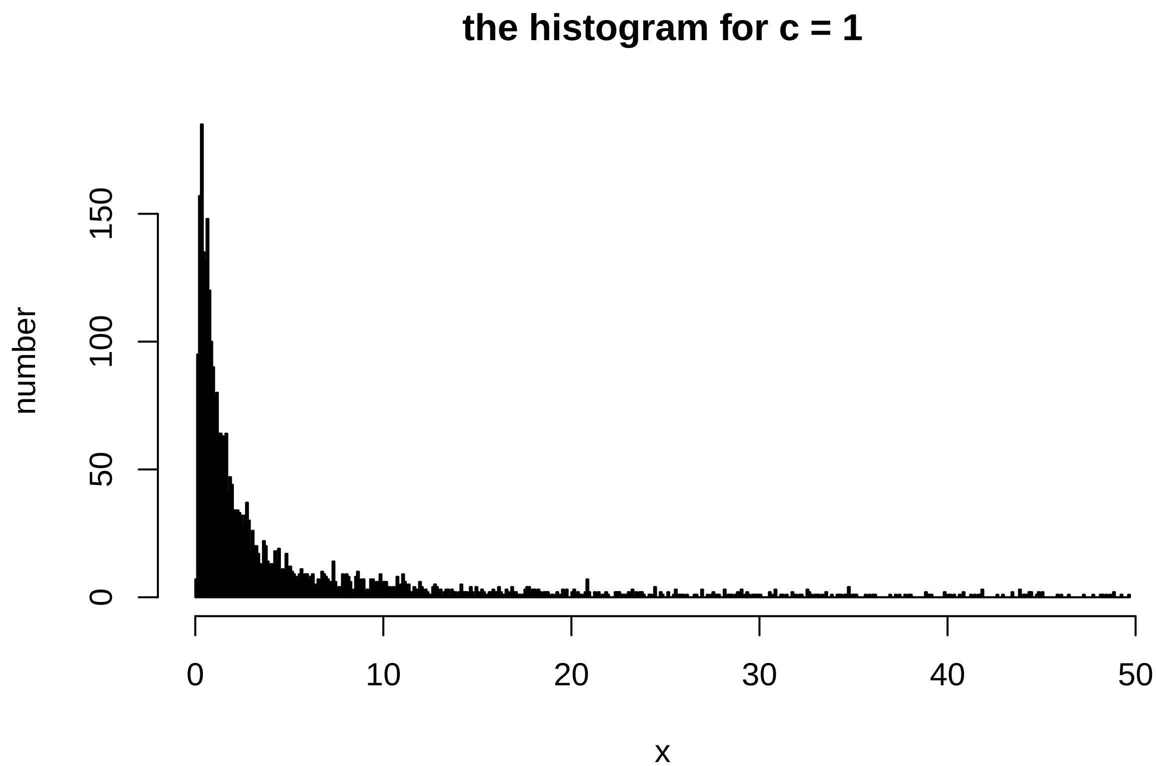
3. For this question, c is 1, 1.2, 1.5, 1.7.

c	1	1.2	1.5	1.7
rejection rate	0.6416	0.4854	0.4392	0.4635
variance	2.265593e+08	3.003488e+14	8.445804e+11	8.104468e+11
mean	680.2713	254165.6891	16803.2909	21445.5837

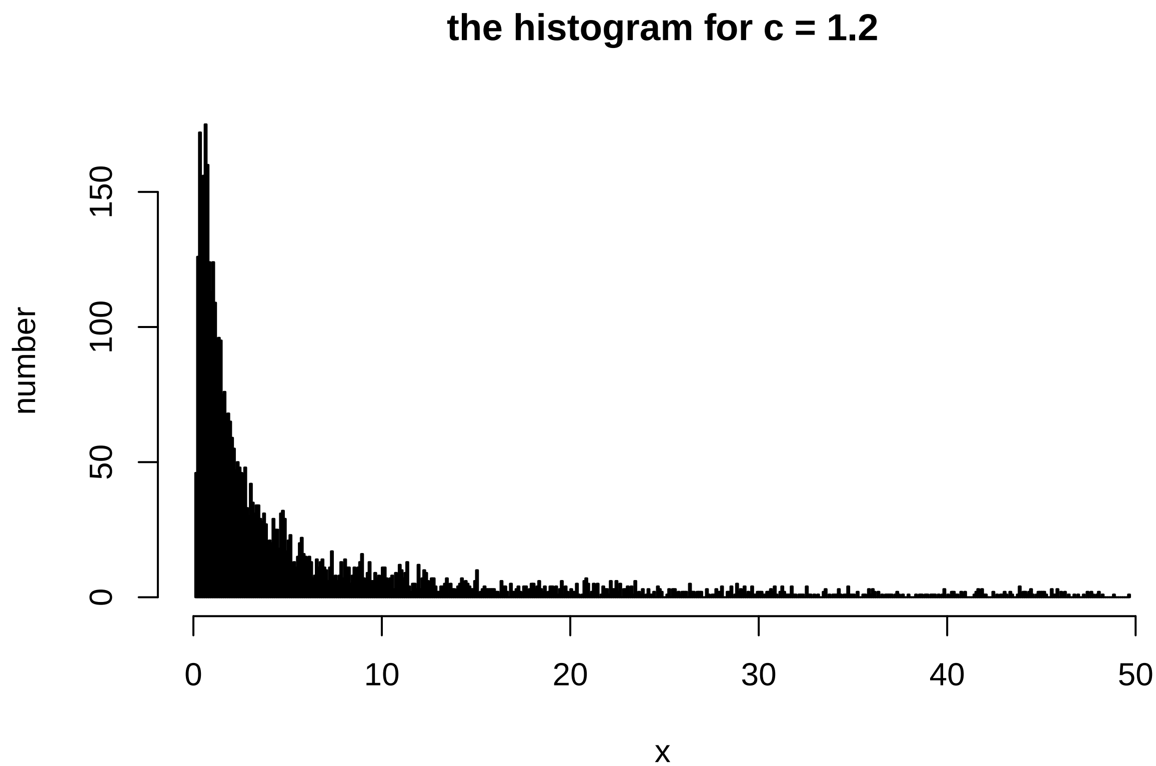
From the values we can see when the value of c increases, rejection rate increase in general. Mean and varance do not reflect the obvious trend.

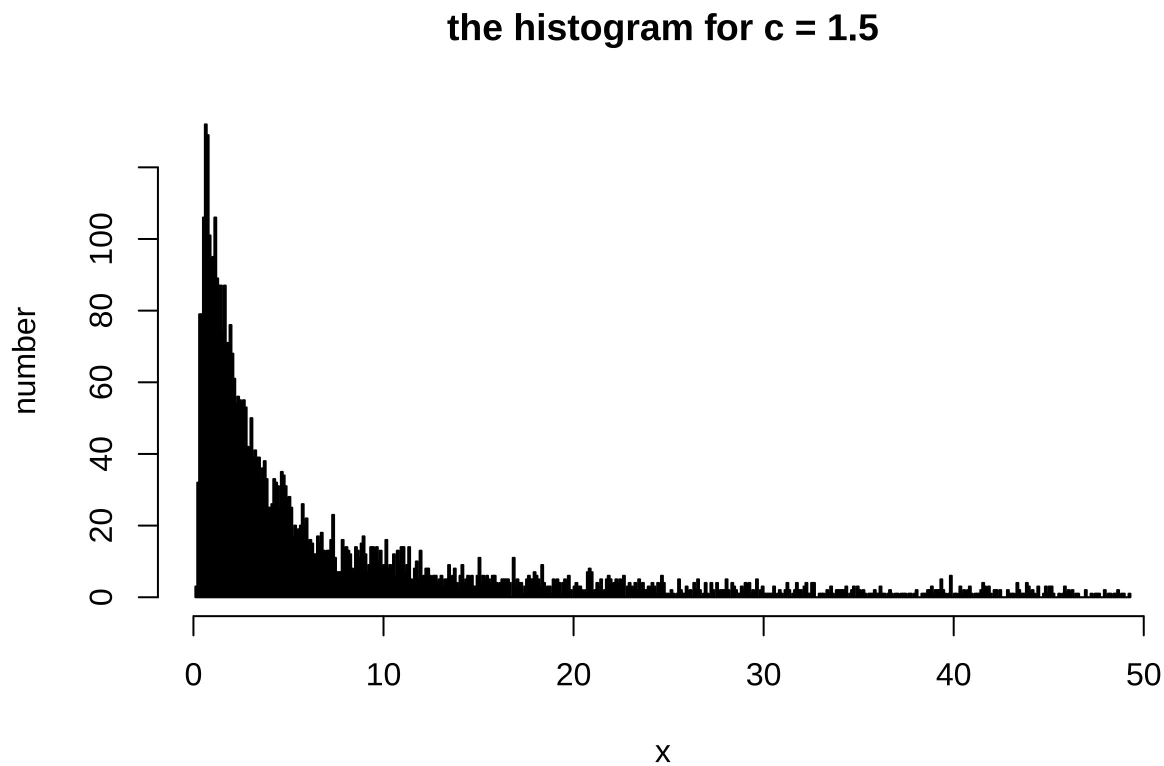
Four histograms are shown below

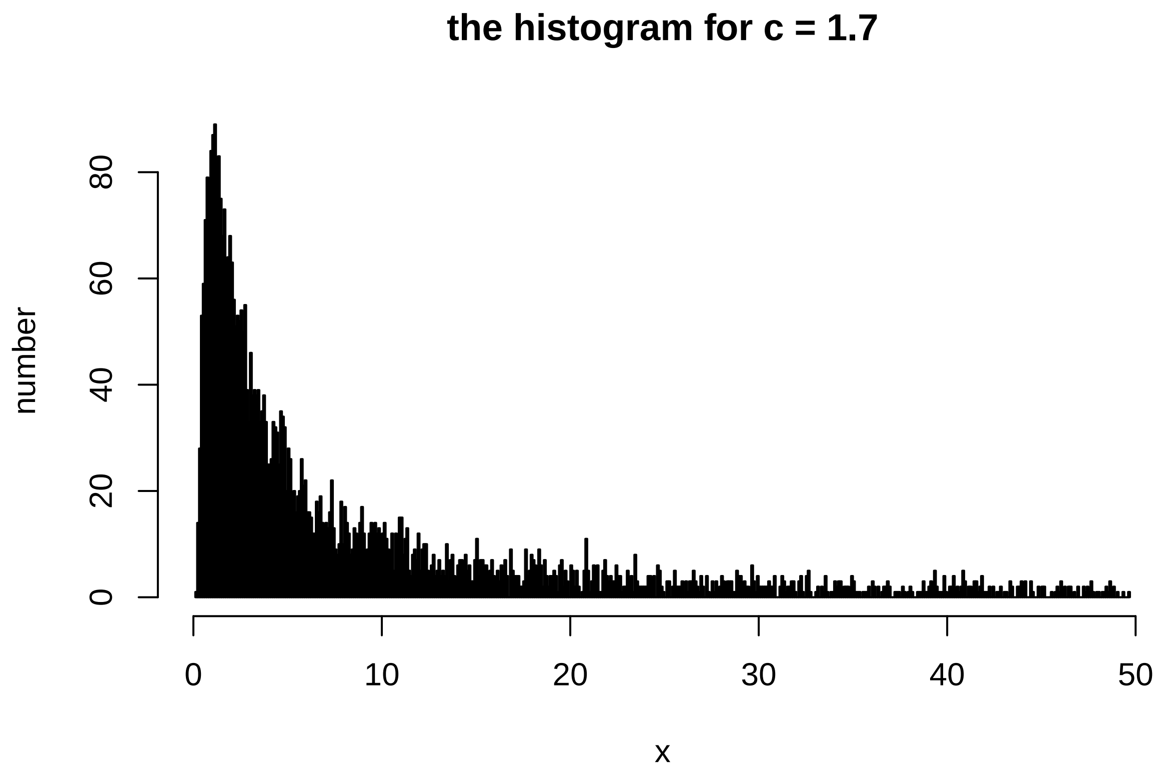
```
## [1] 1.3
```



the histogram for c = 1
number
0
50
100
150
0
10
20
30
40
50
x


the histogram for c = 1.2
number
0
50
100
150
0
10
20
30
40
50
x


the histogram for c = 1.5
number
0
20
40
60
80
100
0
10
20
30
40
50
x



Question 2: Laplace distribution

1.

The Laplace distribution $DE(\mu, \alpha)$ is:

$y = \frac{\alpha}{2} exp(-\alpha|x - \mu|)$

When $\mu = 0$,$\alpha = 1$, the formula is:

$y = \frac{1}{2} exp(-|x|)$

This is the density function of DE(0,1).

We should calculate the CDF function now.

for x<0 The density function is $y = \frac{1}{2} exp(x)$.The CDF function is: $\int_{-\infty}^{x} \frac{1}{2} exp(x)dx = \frac{1}{2} exp(x)$

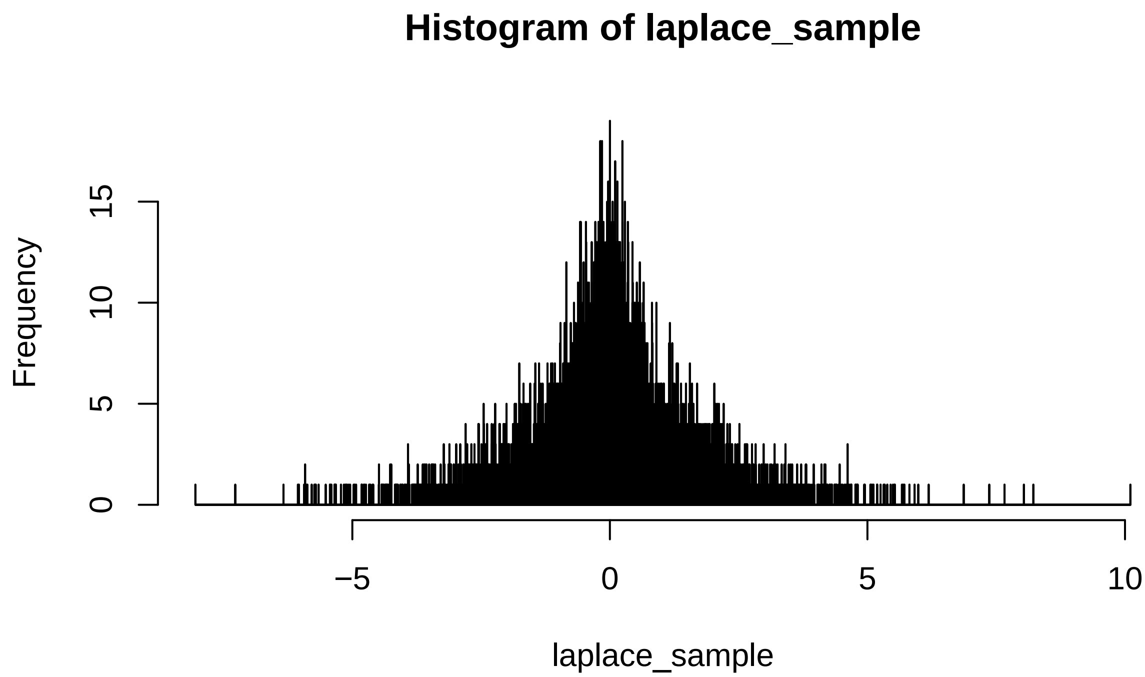
for x>=0 The density function is $y = \frac{1}{2} exp(-x)$.The CDF function is: $\int_{-\infty}^{0} \frac{1}{2} exp(x)dx + \int_{0}^{x} \frac{1}{2} exp(-x)dx$ $= 1 - \frac{1}{2} exp(-x)$

Now we should calculate the inverse CDF function.

for x<0,$y = \frac{1}{2} exp(x)$. So the inverse CDF function is $x = ln2y$,y<0.5

for x>=0,$y = 1 - \frac{1}{2} exp(-x)$. So the inverse CDF function is $x = -ln(2 - 2y)$,y>=0.5

Generate 10000 random numbers from this distribution. Firstly, we generate 10000 random numbers from uniform distribution as y, then get the random numbers from this distribution. Below is the histogram of the 10000 laplace samples.



Q: Explain how you obtained that code step by step. Generate 10000 random numbers from this distribution, plot the histogram and comment whether the result looks reasonable.

From the histogram above and the values we get, we know the mean is about 0.0012, and the variance is about 1.988344 (Because we use the runif function to get the laplace samples. The sample values obtained are different each time).From the formula we know the true mean value is 0,and the true variance is 2.The mean and variance values we get are very close to the true values.

2.

The Acceptance/rejection method must satisfy:

$c * DE(0,1) >= N(0,1)$

We put the density function in this formula:

$c * \frac{1}{2}e^{-|x|} >= \frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}x^2}$

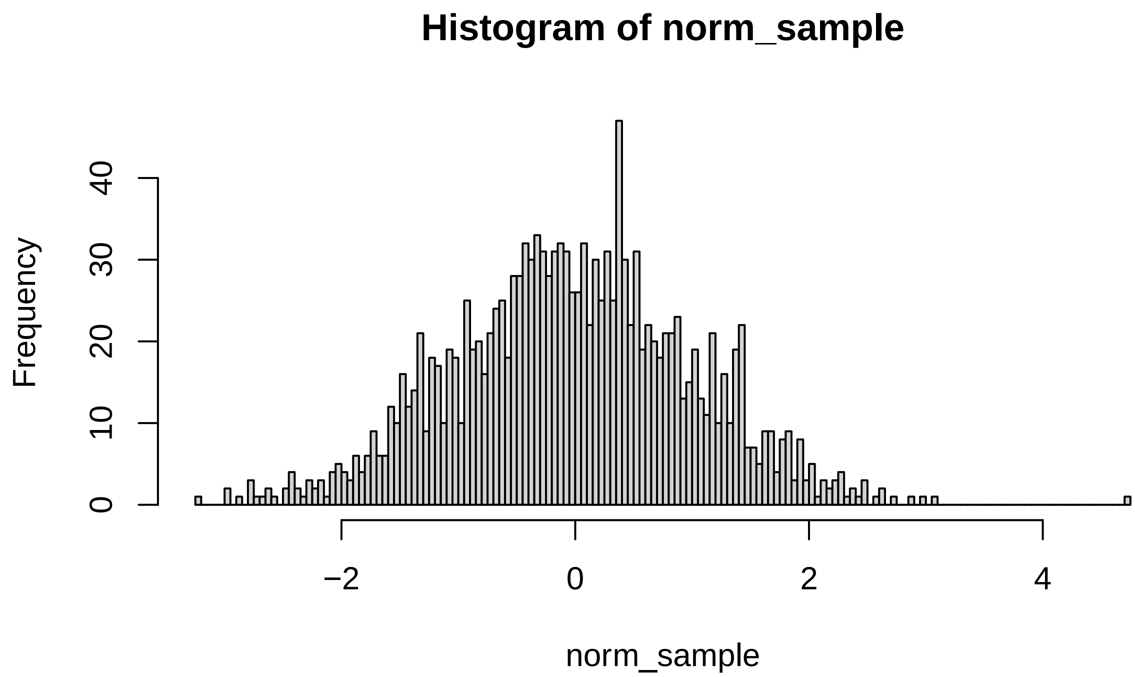
So,we can get c is: $c >= \frac{2e^{-\frac{1}{2}x^2}}{\sqrt{2\pi}e^{-|x|}}$

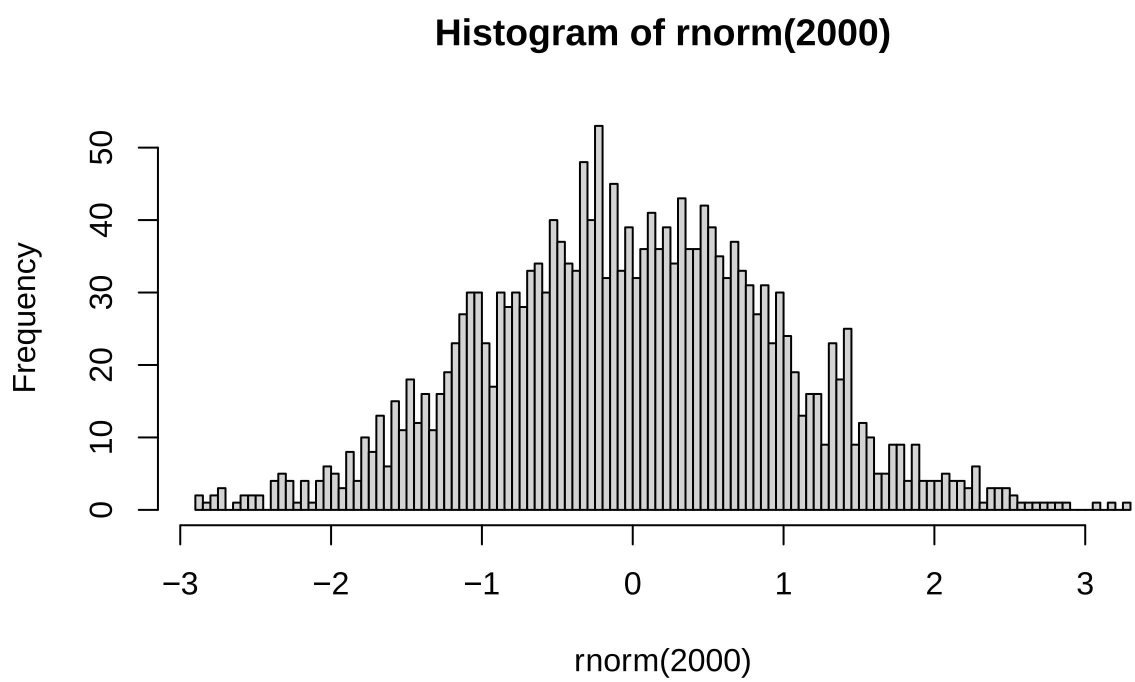
When x<0, $c >= \frac{2e^{-\frac{1}{2}x^2+x}}{\sqrt{2\pi}}$, and we know the biggest value of $-\frac{1}{2}x^2 + x$ is $\frac{1}{2}$, so c>=1.31549.

When x>=0,$c >= \frac{2e^{-\frac{1}{2}x^2-x}}{\sqrt{2\pi}}$, and we know the biggest value of $-\frac{1}{2}x^2 - x$ is also $\frac{1}{2}$, so c>=1.31549.

So we choose c=1.31549.

The first figure below is the histogram of the 2000 samples. The second figure below is the histogram of using the normal distribution function.





Q: Compute the average rejection rate R in the acceptance/rejection procedure. What is the expected rejection rate ER and how close is it to R? Generate 2000 numbers from N(0,1) using standard rnorm() procedure, plot the histogram and compare the obtained two histograms.

In the 2000 samples, we have 485 reject samples. So the average rejection rate is 485/2000=0.2425. Expected rejected rate is 1-1/c=0.239827. The average rejection rate is very close to the expected rejected rate. As we can see from the figures below, the two figures are very similar.

Appendix

Code in Question 1

```
#1
#generate 10000 random numbers from uniform distribution
uniform_sample<-runif(n=10000,min=0,max=1)
laplace_sample<-c()
for(i in 1:10000){
  ifelse(uniform_sample[i]<0.5,
         laplace_sample[i]<-log(2*uniform_sample[i]),
         laplace_sample[i]<- -log(2-2*uniform_sample[i]))
}
#plot the histogram
hist(laplace_sample,breaks = 10000)
Mean_laplace_sample<-mean(laplace_sample)
Var_laplace_sample<-var(laplace_sample)
#From the histogram above and the value we get,we know the mean is about 0.0012,
#and the variance is about 1.988344(because we use the runif function to get the
#samples,the samples are different every time).From the formula we know the
#actual mean value is 0,and the variance is 2.The values we get are very close
#to the actual values.


#2
#Use the Acceptance/rejection method with DE(0,1) as a majorizing density
# to generate N(0,1) variables.

#Firstly,there must  exist a constant c,for each x:c*DE(0,1)>=N(0,1)

#Generate 2000 random numbers DE(0,1)
uniform_sample2000<-runif(n=2000,min=0,max=1)
laplace_sample2000<-c()
for(i in 1:2000){
  ifelse(uniform_sample2000[i]<0.5,
         laplace_sample2000[i]<-log(2*uniform_sample2000[i]),
         laplace_sample2000[i]<- -log(2-2*uniform_sample2000[i]))
}
#the Acceptance/rejection method
c<-1.31549
num.reject<-0
norm_sample<-c()
for(j in 1:2000){
  u<-runif(1)
  fX_Y<-dnorm(laplace_sample2000[j])
  fY_Y<-1/2*exp(-abs(laplace_sample2000[j]))
  #find the accept samples
  if(u<=fX_Y/(c*fY_Y)){
    norm_sample[j]<-laplace_sample2000[j]
```

```
  }
  else{
    num.reject<-num.reject+1
  }
  norm_sample<-norm_sample[!is.na(norm_sample)]
}
hist(norm_sample,breaks = 200)
hist(rnorm(2000),breaks = 200)
```

```
### Code in Question 2
# 1.1
# the code for ploting f(x) and f_p_x
library(ggplot2)
f_x <- function(c,x){
  res <- c * (sqrt(2 * pi))^-1 * exp((-c^2)/(2 * x)) * x ^ (-3/2)
  return(res)
}

f_p_x <- function(alpha,Tmin,x){
  x_part2 <- x[which(x > Tmin)]
  y_part2 <- (alpha - 1) / Tmin * (x_part2 / Tmin)^-alpha
  x_part1 <- x[-which(x > Tmin)]
  y_part1 <- rep(0,length(x_part1))
  res <- c(y_part1,y_part2)
  return(res)
}

# x <- c(1:2000)/400
# c <- 1
# alpha <- 2
# Tmin <- 1
y1 <- f_x(1,c(1:2000)/400)
y2 <- f_p_x(2,1,c(1:2000)/400)
data1 <- data.frame(input = c(1:2000)/400, output_1 = y1, output_2 = y2 )
data2 <- reshape2::melt(data1,id.var = "input")
ggplot(data2,aes(x = input, y = value )) + geom_point(aes(color = variable))
```

```
# the code for ploting f(x) and f_final_x
f_final_x <- function(alpha,Tmin,x){
  x_part2 <- x[which(x > Tmin)]
  y_part2 <- (alpha - 1) / Tmin / alpha * (x_part2 / Tmin)^-alpha
  x_part1 <- x[-which(x > Tmin)]
  y_part1 <- rep((alpha - 1) / Tmin / alpha,length(x_part1))
```

```
  res <- c(y_part1,y_part2)
}

# x <- c(1:2000)/400
# c <- 1
# alpha <- 1.2
# Tmin <- 1
y3 <- f_final_x(1.2,1,c(1:2000)/400)
data3 <- data.frame(input = c(1:2000)/400, output_1 = y1, output_2 = y3 )
data4 <- reshape2::melt(data3,id.var = "input")
ggplot(data4,aes(x = input, y = value )) + geom_point(aes(color = variable))




########################################################################
# 1.2

library(poweRlaw)
library(stats)
# the code for calculating M

g1_x <- function(x,alpha,Tmin,c){
  return( alpha * Tmin / (alpha - 1) * c * (sqrt(2 * pi))^-1 * exp((-c^2)/
    (2 * x)) * x ^ (-3/2))
}

g2_x <- function(x,alpha,Tmin,c){
  return(alpha * Tmin^(1 - alpha) / (alpha - 1) * c * (sqrt(2 * pi))^-1 *
    exp((-c^2)/(2 * x)) * x ^ (-3/2 + alpha))
}




# the code for sampling from the distribution with the majorizing density
sample_1 <- function(number,Tmin,alpha){
  number_1 <- round(number * (alpha - 1) / alpha)
  number_2 <- number - number_1
  sample_first <- runif(number_1,min = 0,max = Tmin)
  sample_second <- rplcon(number_2,Tmin,alpha)
  return(c(sample_first,sample_second))

}

sample_accepted <- function(M,sample,alpha,Tmin,c){
  accepted_sample <- c()
  num <- length(sample)
  for(i in 1:num){
    f_x <- c * (sqrt(2*pi))^-1 * exp(-c^2 / 2 /sample[i]) * sample[i] ^ (-3 / 2)
    if(sample[i] >= Tmin){
      f_final_x <- (alpha - 1) / (alpha * Tmin) *(sample[i] / Tmin) ^ -alpha
    }else if(sample[i] < Tmin){
```

```
      f_final_x <- (alpha - 1) / (alpha * Tmin)
    }
    u <- runif(1)
    if(f_x / (M * f_final_x) >= u){
      accepted_sample <- c(accepted_sample,sample[i])
    }
  }
  return(accepted_sample)

}

################################################################################
1.3
c <- c(1,1.2,1.5,1.7)
alpha <- 1.2
Tmin <- 1
mean <- c()
variance <- c()
rejection_rate <- c()
sample_original <- sample_1(10000,Tmin,alpha)

for(i in c){
  #get the value of M
  M1 <- g1_x(i**2/3,alpha,Tmin,i)
  M2 <- g2_x(i**2/(3-2*alpha),alpha,Tmin,i)
  M <- max(M1,M2)

  #get the accepted sample
  accepted_sample <- sample_accepted(M,sample_original,alpha,Tmin,i)
  rejection_rate <- c(rejection_rate,
                (length(sample_original) -
                   length(accepted_sample)) / length(sample_original) )
  variance <- c(variance,var(accepted_sample))
  mean <- c(mean,mean(accepted_sample))
  hist(accepted_sample[which(accepted_sample < 50)],xlab = "x",
       ylab = "number",main = paste("the histogram for c =",i,sep = " "),
       breaks =500 )
}
```