Computer Lab 3

Thomas Zhang

2016 M02 18

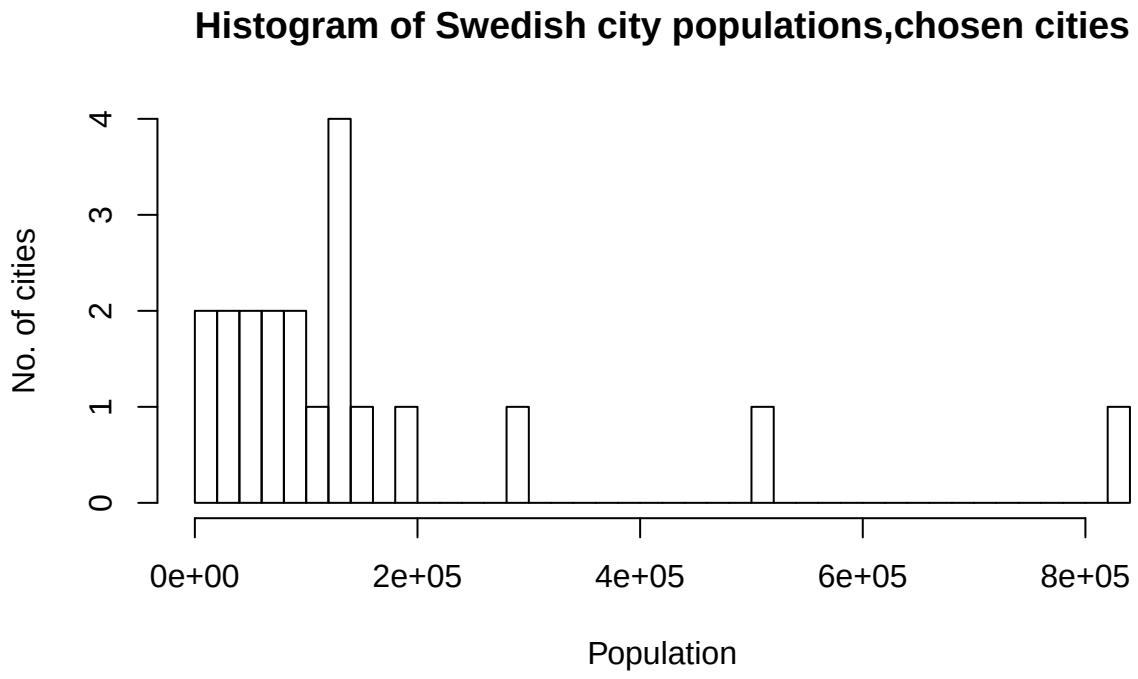
Assignment 1

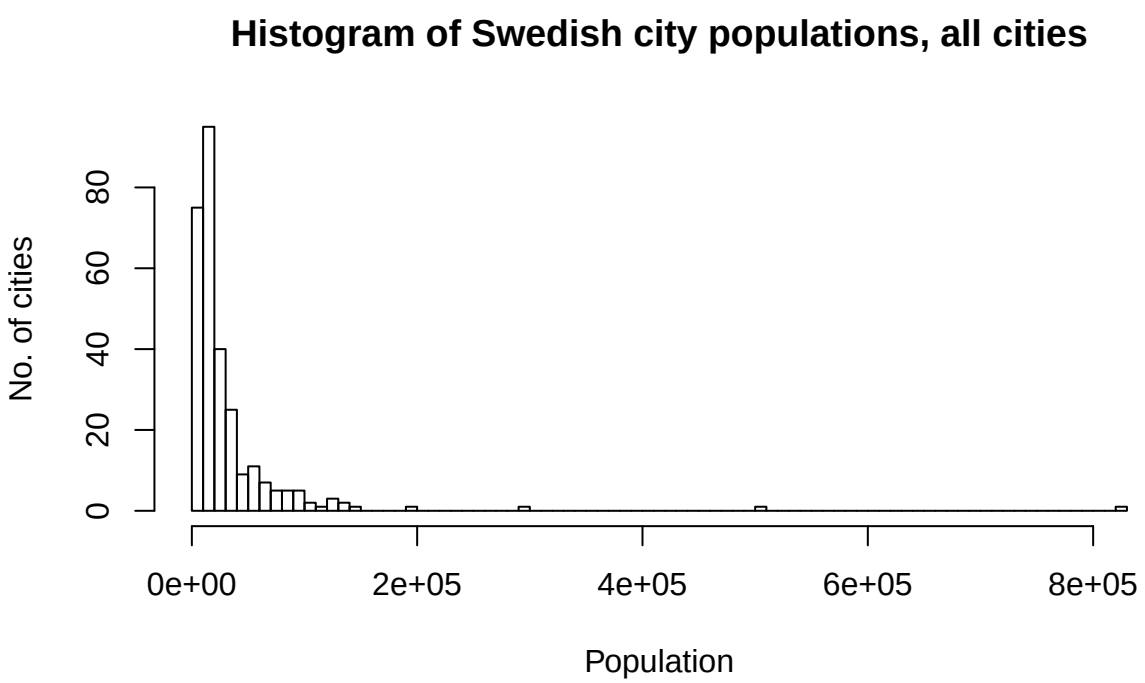
This task asks for a sampler of twenty opinion polling cities in Sweden without replacement where the probability of a city being chosen is proportional to the population of the city. In a large city the different municipalities (kommuner) of Sweden count as individual cities. We implement this sampler in `R`.

```
##  [1] "Västerås"    "Landskrona"  "Uppsala"     "Borås"       "Helsingborg"
##  [6] "Sala"        "Munkedal"    "Motala"      "Stockholm"   "Linköping"
## [11] "Värmdö"      "Halmstad"    "Jönköping"   "Luleå"       "Malmö"
## [16] "Växjö"       "Täby"        "Kinda"       "Göteborg"    "Norrköping"
```

We see that the cities/municipality chosen by the sampler are relatively large cities and there is often at least one city/municipality from one of the regions Stockholm-Göteborg-Malmö present in the list.

Let us plot the population histogram of the chosen cities and compare that to the population histogram of all the swedish cities/municipalities.





We see that the histograms have almost the same shape in both cases. In no way, however, are the cities/counties chosen by the sampler representative of the sizes of all swedish cities. In fact, since the probability of being chosen is proportional to population, larger cities should be overrepresented and vice versa for smaller cities.

Assigment 2

First, we generate a sample from the standard double exponential distribution (the standard Laplace distribution) using the uniformly distributed numbers U using the inverse CDF method.

It is known that the CDF of the double exponential distribution is:

$$\begin{cases} \frac{1}{2}exp(\alpha(x-\mu)) & \text{if } x < \mu \\ 1 - \frac{1}{2}exp(-\alpha(x-\mu)) & \text{if } x \geq \mu \end{cases}$$

In this case since we want a DE(0,1) it becomes:

$$\begin{cases} \frac{1}{2}exp(x) & \text{if } x < 0 \\ 1 - \frac{1}{2}exp(-x) & \text{if } x \geq 0 \end{cases}$$

To find F_X^{-1} we have to solve for x these two equations:

$$\begin{cases} U = \frac{1}{2}exp(x) & \text{if } x < 0 \\ U = 1 - \frac{1}{2}exp(-x) & \text{if } x \geq 0 \end{cases}$$

Solving for the first one:

$$U = \frac{1}{2}exp(x) \text{ implies } 2U = exp(x) \text{ implies } x = \log 2U$$

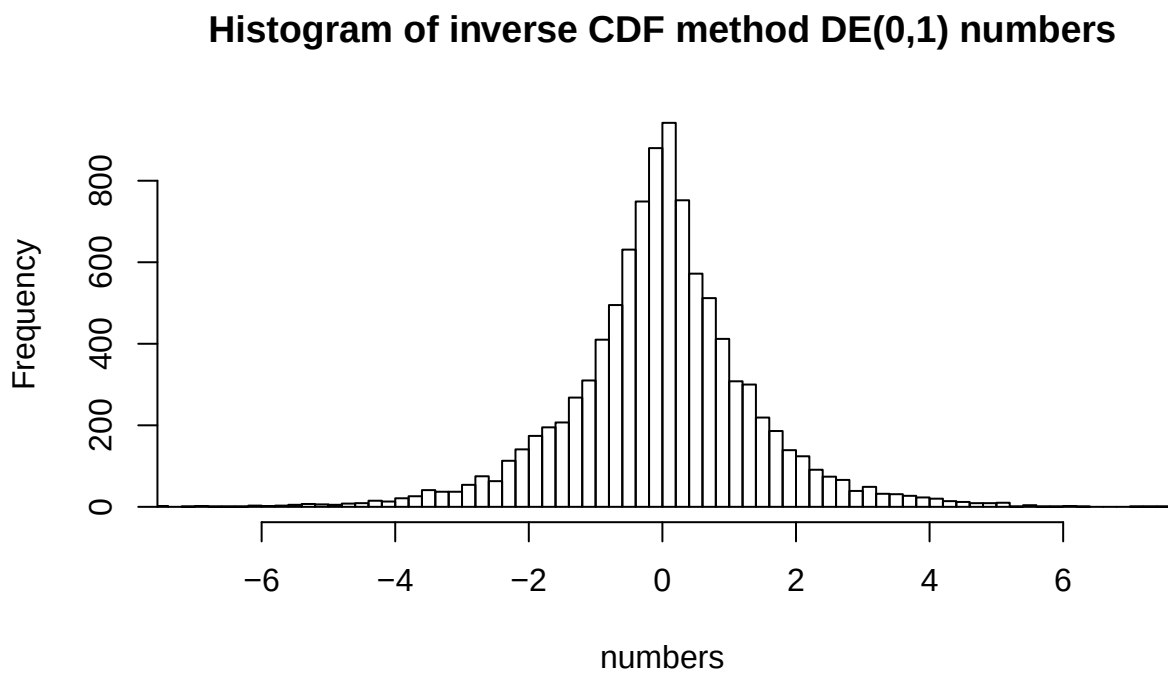
and $x < 0$ implies $\log 2U < 0$ implies $U < \frac{1}{2}$

Solving for the second one:

$$U = 1 - \frac{1}{2}exp(-x) \text{ implies } -2(U-1) = exp(-x) \text{ implies } x = -\log(2-2U)$$

and $x \geq 0$ implies $\log(2-2U) \geq 0$ implies $U \geq \frac{1}{2}$

We generate 10000 numbers x and plot the histogram.



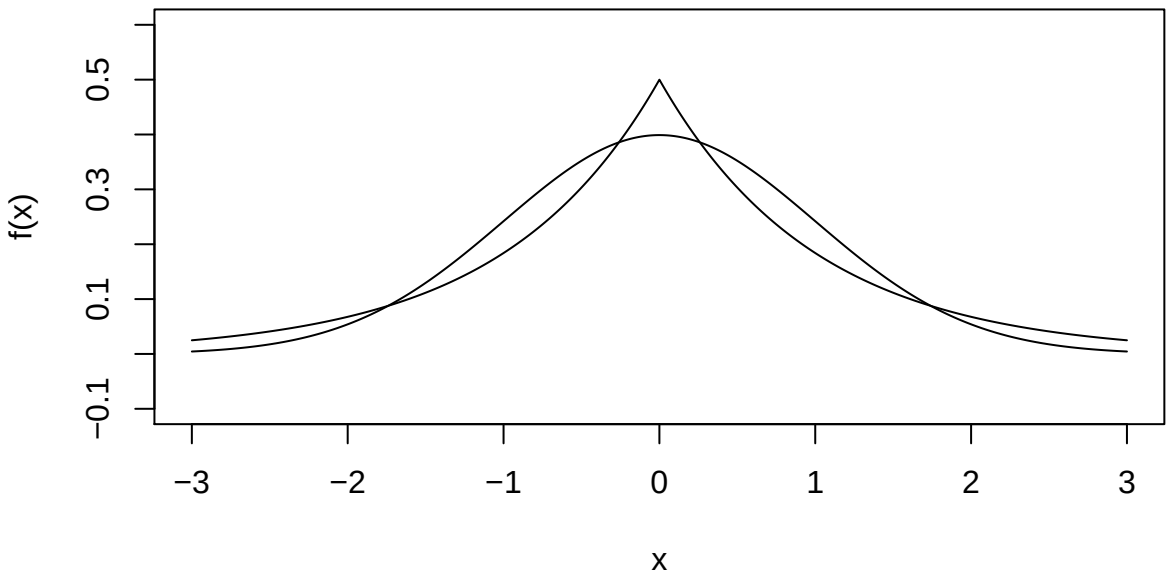
We see that the shape of the histogram is suggestive of the double exponential function, so this is as expected.(In particular, note the second derivative of the slope coming from the left is positive all the way up. A normal density function tend to have an inflection point half-way up the slope and then the second derivative turns negative.)

Next, we are to originate standard normal distributed numbers from double exponentially distributed numbers using the Acceptance/Rejection method. This method involves evaluating the truth value of the statement

$$U \leq \frac{f_X(x)}{cg_X(x)}$$

where U is a observation of a uniformly distributed U(0,1) variable, $f_X(x)$ is the target density from which we wish our numbers to originate from, and $g_X(x)$ is a majorizing density from which all candidate numbers are actually generated from. The constant c is a constant chosen such that $cg_X(x) > f_X(x)$ for all possible x. The gist of the method is that we only accept those numbers x for which the statement above is true. Then the accepted numbers will have density distribution $f_X(x)$.

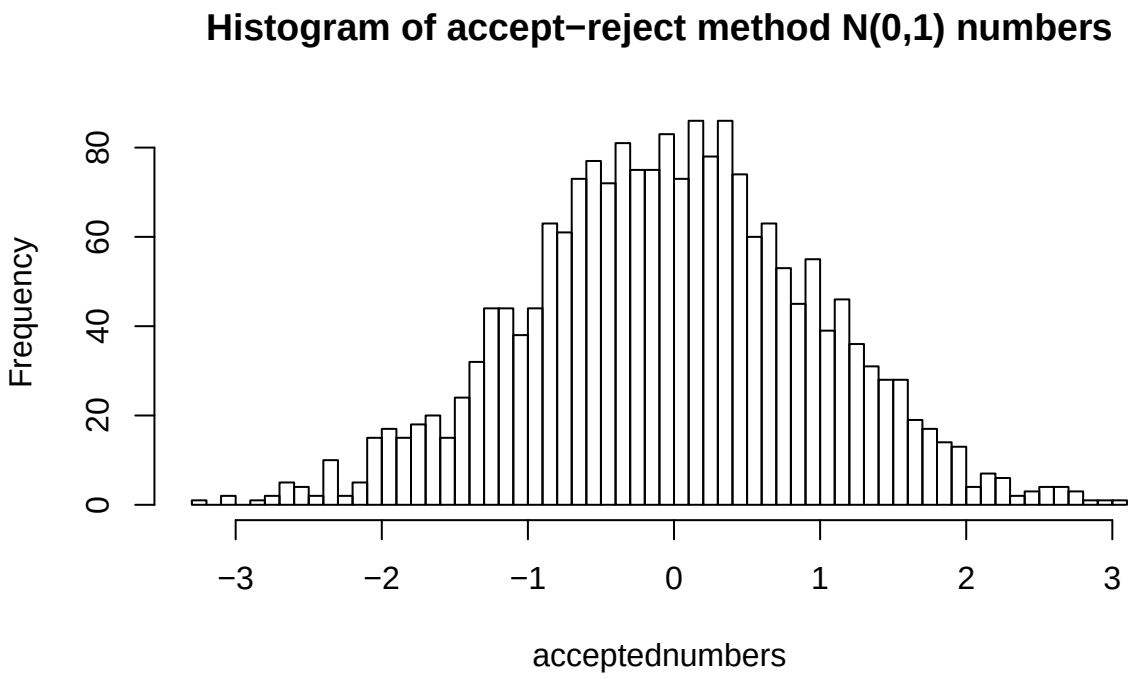
In order to find constant c, we plotted the two density functions and found c as the maximum ratio between the standard normal function and the standard double exponential function. We found that $c \approx 1.315$.

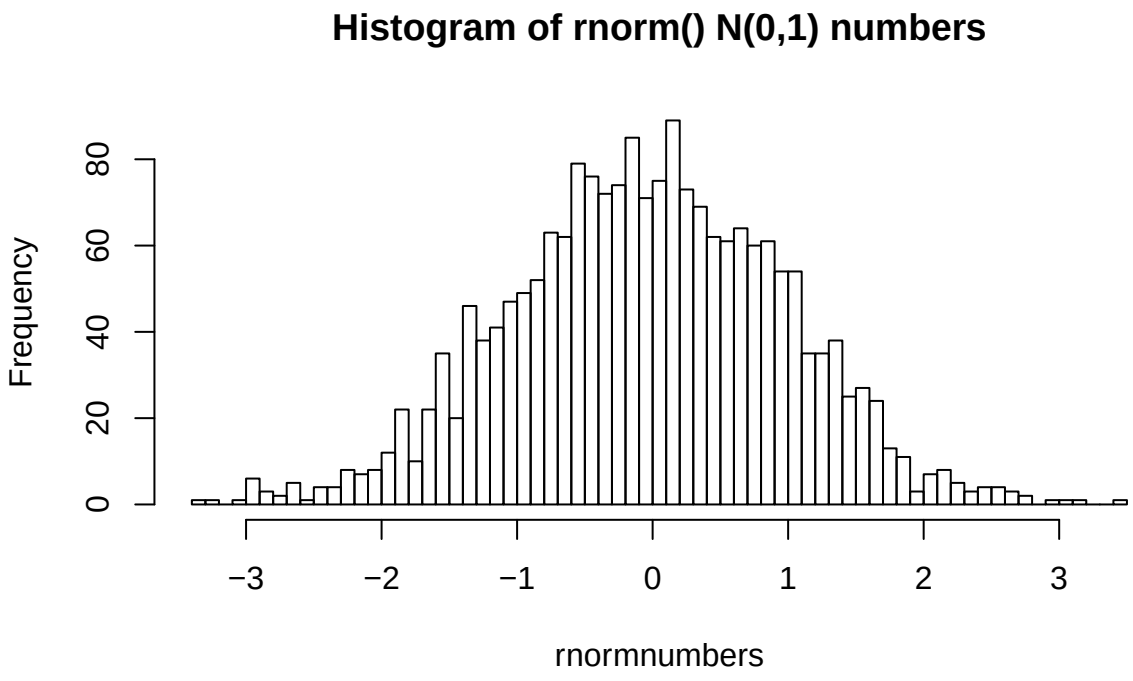


Let us generate 2000 numbers and plot the histogram and compare that to the histogram of 2000 numbers generated by the R function call `rnorm(2000)`.

```
## [1] "Theoretical Rejection rate: 0.24"
```

```
## [1] "Actual Rejection rate: 0.245"
```





We see that the two histograms are reminiscent of the normal density function, which is good. We also find that the actual rejection rate is close to the theoretical rejection rate.

Appendix

R code

```
## Assignment 1
library(XLConnect)
wb = loadWorkbook(paste0(getwd(),"/population.xls"))
data = readWorksheet(wb, sheet = "Table", header = TRUE)
data <- data[-which(nchar(data$Statistics.Sweden) == 2),]
data <- data[-(1:4),c(2,4)]
data[,2] <- as.numeric(data[,2])

propcitypicker <- function(cities){
  citypopulations <- data[match(cities,data[,1]),2]
  totalcitypop <- sum(citypopulations)
  binborders <- cumsum(citypopulations / totalcitypop)
  roll <- runif(1)
  if( roll < binborders[1]){
    return(cities[1])
  }
  for(i in 1:(length(cities)-1)){
    if( binborders[i] < roll && roll <= binborders[i+1]){
      return(cities[i+1])
    }
  }
}

cities <- data[,1]
listofcities <-c()
counter <- 1
while(counter <= 20){
  chosencity <- propcitypicker(cities)
  cities <- cities[-match(chosencity,cities)]
  listofcities <- c(listofcities,chosencity)
  counter <- counter + 1
}

listofcitysizes <- data[match(listofcities,data[,1]),2]
listofcities
hist(listofcitysizes,breaks = 50,main="Histogram of Swedish city populations,chosen cities",
     xlab="Population",ylab = "No. of cities")
hist(data[,2],breaks=100,main="Histogram of Swedish city populations, all cities",
     xlab="Population",ylab = "No. of cities")
## Assignment 2
unifinput <- runif(10000)
norminput <- rnorm(10000)
DEnumbers <- function(x){
  res <- c()
  for(i in 1:length(x)){
    if(x[i] < 0.5){
      res <- c(res,log(2*x[i]))
    } else{
      res <- c(res,-log(2*(1 - x[i])))
```

```
    }
  }
  return(res)
}
DEoutput <- DEnumbers(unifinput)
hist(DEoutput,main="Histogram of inverse CDF method DE(0,1) numbers",xlim=c(-7,7),xlab = "numbers",breal
doubleexp <- function(x){
  return(1/2 * exp(-abs(x)))
}
xordered <- seq(from=-3,to=3,by=0.001)
plot(xordered,dnorm(xordered),type="l",ylim=c(-0.1,0.6), ylab="f(x)",xlab ="x",
     main= "Std. Normal and Double exponential functions")
lines(xordered,doubleexp(xordered))
c <- dnorm(xordered)[which.max(dnorm(xordered)/doubleexp(xordered))] /
  doubleexp(xordered)[which.max(dnorm(xordered)/doubleexp(xordered))]

unifnumbers <- runif(8000)
DEnumbers <- sample(DEoutput,8000)
normnumbers <- sample(norminput,8000)
j <- 1

acceptednumbers <-c()

repeat{
  if(unifnumbers[j] <= dnorm(DEnumbers[j]) / (c * doubleexp(DEnumbers[j]))){
    acceptednumbers <- c(acceptednumbers,DEnumbers[j])
  }
  if(length(acceptednumbers) >= 2000){
    break
  }
  j <- j + 1
}
paste("Theoretical Rejection rate:",round(1 - 1/c,3))
paste("Actual Rejection rate:",round((j - 2000) / j,3))
hist(acceptednumbers,main="Histogram of accept-reject method N(0,1) numbers",
     breaks = 50)
hist(sample(norminput,2000),main="Histogram of rnorm() N(0,1) numbers",xlab="rnormnumbers",breaks = 50)
## NA
```