Homework 01 - Statistical Methods for Data Science

Letícia Negrão Pinto

DSSC - 2019

DAAG - Chapter 1

Exercise 11

Run the following code:

```
gender <- factor(c(rep("female", 91), rep("male", 92)))
table(gender) #1
```

```
## gender
## female   male
##     91     92
```

```
gender <- factor(gender, levels=c("male", "female"))
table(gender) #2
```

```
## gender
##   male female
##     92     91
```

```
gender <- factor(gender, levels=c("Male", "female"))
# Note the mistake: "Male" should be "male"
table(gender) #3
```

```
## gender
##   Male female
##      0     91
```

```
table(gender, exclude=NULL) #4
```

```
## gender
##   Male female   <NA>
##      0     91     92
```

```
rm(gender) # Remove gender
```

- Explain the output from the successive uses of table().

Answer: At first a vector named 'gender' was constructed with two categories: 'male' and 'female'.

(#1) The first table() returns the number of males and females as defined in the assignement of 'gender'.

(#2) The second table() returns the same as the previous one, but this time the vector was redefined to be the levels of 'male' and 'female' of vector 'gender'.

(#3) The third table() will not return the results for 'male' because of the typo ('Male' instead of 'male') when redefining the vector 'gender'. In this case the vector will have only 'female' and '', so the table() will return 0 Male.

(#4) Finally, when we type 'table(gender, exclude=NULL)' it will return the amount correspondent to 'Male', 'female', and '', as discussed previously.

Exercise 15

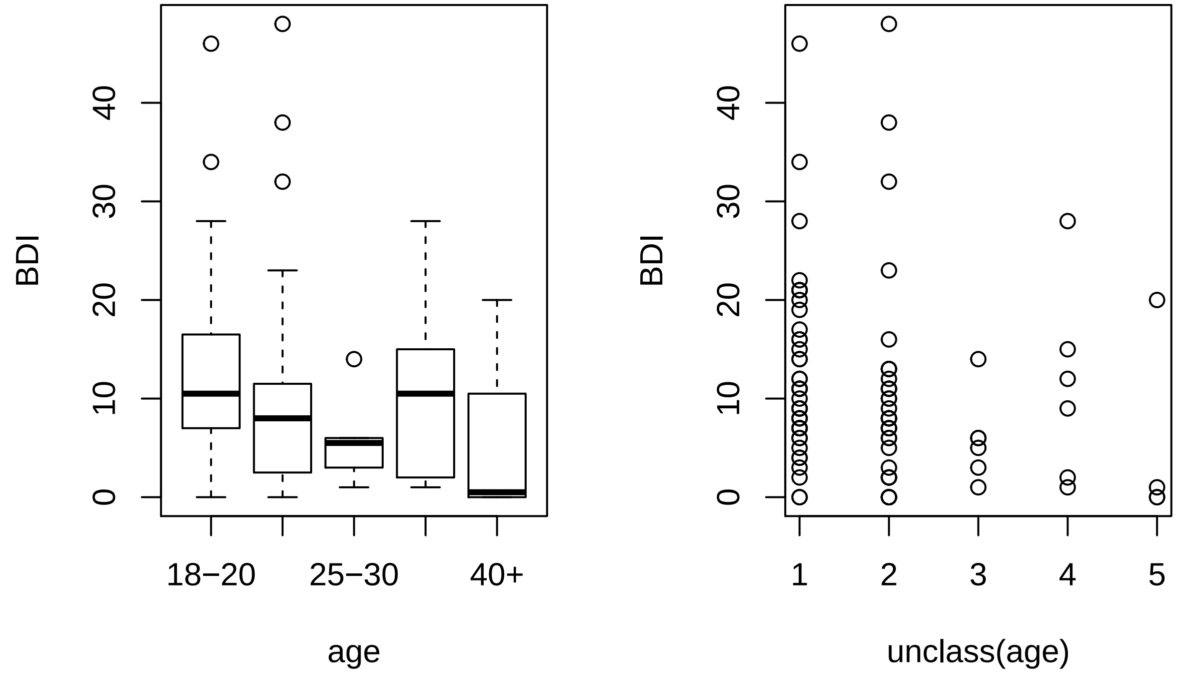
The data frame socsupport (DAAG) has data from a survey on social and other kinds of support, for a group of university students. It includes Beck Depression Inventory (BDI) scores. The following are two alternative plots of BDI against age:

```
library(DAAG)
```

```
## Warning: package 'DAAG' was built under R version 3.5.3
```

```
## Loading required package: lattice
```

```
par(mfrow=c(1,2))

plot(BDI ~ age, data=socsupport)
plot(BDI ~ unclass(age), data=socsupport)
```



- For examination of cases where the score seems very high, which plot is more useful? Explain.

Answer: In cases where the score seems very high the most useful plot is the first one $plot(BDI \sim age, data = socsupport)$ because the outliars apear explicitly.

- Why is it necessary to be cautious in making anything of the plots for students in the three oldest age categories (25-30, 31-40, 40+)?

Answer: We need to be cautious because the number of students in these categories is very low to make inferences.

```
library(DAAG)
data = socsupport
table(data$age)
```

```
##
## 18-20 21-24 25-30 31-40   40+
##    44    35     6     6     4
```

Core Statistics (CS) - Chapter 1

Exercise 1.1

Exponential random variable, X = 0, has p.d.f. $f(x) = \lambda exp(-\lambda x)$.

1. Find the c.d.f. and the quantile function for X.

Answer: The Cumulative Probability Function, P(x), of a random variable x expresses the probability that x does not exceed the value x, as a function of x. $F(x) = P(X \leq x)\ \forall x$.

In the case of an Exponential distribution, we have the following: $F(x) = \int_{-\infty}^{x} f(x)dx = 1 - exp(-\lambda x)$ for $x \geq 0$, 0 otherwise.

If we have x = 0, so the probability will be $1 - exp(-\lambda.0) = 0$, as expected.

For the p-th quantile, we will have the following expression:

$$p = 1 - exp(-\lambda x)$$

$$1 - p = exp(-\lambda x)$$

$$x = \frac{-ln(1-p)}{\lambda}$$

So, it means that if x = 0 we have p = 0.

2. Find $Pr(X < \lambda)$ and the median of X.

Answer: As discussed previously:

$$P(x < \lambda) = \int_{-\infty}^{\lambda} f(x)dx = 1 - exp(-\lambda^2)$$

The median will be given by p = 0.5, so:

$$median = -\frac{ln(1-0.5)}{\lambda} = -\frac{ln(0.5)}{\lambda}$$

3. Find the mean and variance of X.

Answer: The mean will be given by:

$$mean = E[x] = \int_{0}^{\infty} x\lambda exp(-\lambda x)dx$$

$$\lambda\left[\frac{-xexp(-\lambda x)}{\lambda}\Bigg|_{0}^{\infty} + \frac{1}{\lambda}\int_{0}^{\infty} exp(-\lambda x)dx\right]$$

$$\lambda\left[0 + \frac{1}{\lambda} - \frac{exp(-\lambda x)}{\lambda}\Bigg|_{0}^{\infty}\right] = \lambda\frac{1}{\lambda^2}$$

$$mean = \frac{1}{\lambda}$$

And the variance:

$$Var(x) = E[x^2] - E[x]^2$$

$$E[x^2] = \int_0^\infty x^2 \lambda exp(-\lambda x)dx = \frac{2}{\lambda^2}$$

$$Var(x) = \frac{2}{\lambda^2} - \frac{1}{\lambda^2}$$

$$variance = \frac{1}{\lambda^2}$$

Exercise 1.8

If $log(X) \sim N(\mu, \sigma^2)$, find the p.d.f. of X.

Answer: This distribution is known as Lognormal distribution. Considering: $ln(X) = Y$; $Y {\sim} N(\mu, \sigma^2)$

We have:

$$f_Y(Y) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(Y-\mu)^2}{2\sigma^2})$$

Doing:

$$X = e^Y = g(Y)$$

$$Y = ln(X) = g^{-1}(X)$$

We get that:

$$\frac{dg^{-1}(X)}{dX} = \frac{1}{X}$$

And the following relation is true:

$$f_Y(Y) = f_X[g^{-1}(Y)] \left| \frac{dg^{-1}(Y)}{dY} \right|$$

or: $f_Y dY = f_X(X)dX$

Then, we finally have the p.d.f. of X:

$$f_X(X) = \frac{1}{X\sqrt{2\pi\sigma^2}} exp(-\frac{(ln(X)-\mu)^2}{2\sigma^2})$$

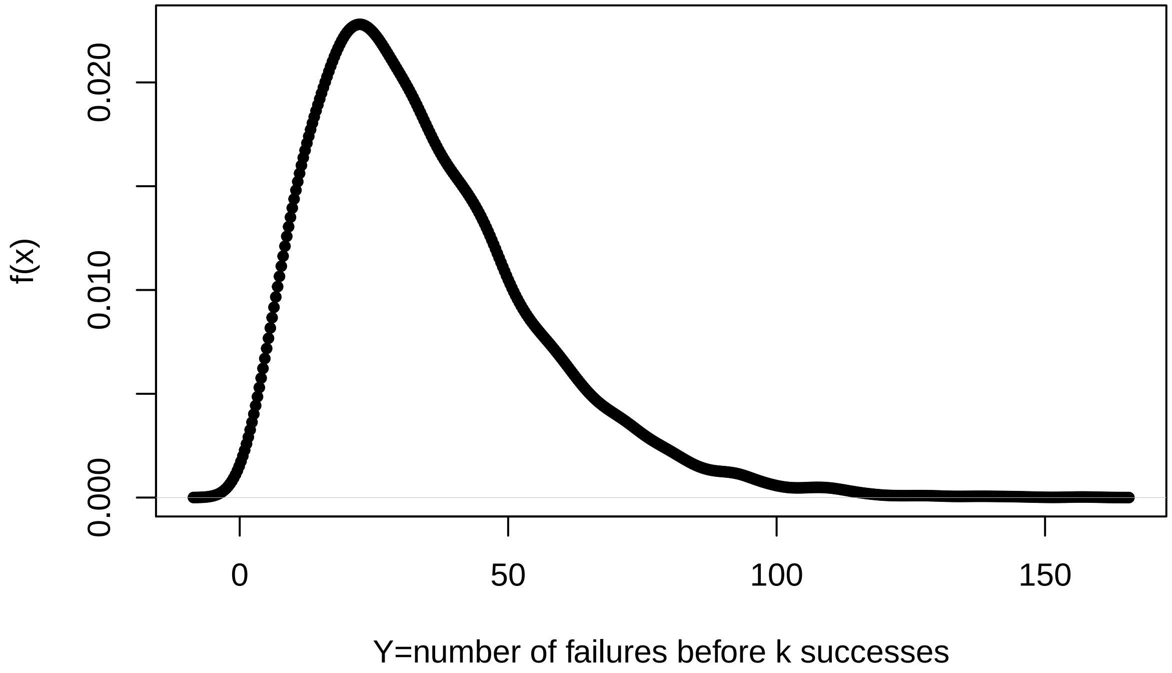
Lab Exercises

Exercise 2

Generate in R the same output, but using rgeom() for generating the random variables. Hint: generate n times three geometric distribution X1,. . . ,X3 with p=0.08, store them in a matrix and compute then the sum Y.

```r
set.seed(2019)

# Matrix with geometric distributions:
n <- 10000
prob <- 0.08

Xmatrix = matrix(nrow=n, ncol=3)

# Each column corresponds to X1, X2 and X3:
for (i in 1:3)
```

```
{
  Xmatrix[,i]=c(rgeom(n,prob))
}

# Y=X1+X2+X3:
Y=c(rowSums(Xmatrix))

# Graph:
plot(density(Y), type = "p", col = "black", lwd = 1, pch=20,
xlab="Y=number of failures before k successes", ylab="f(x)", main="")
```

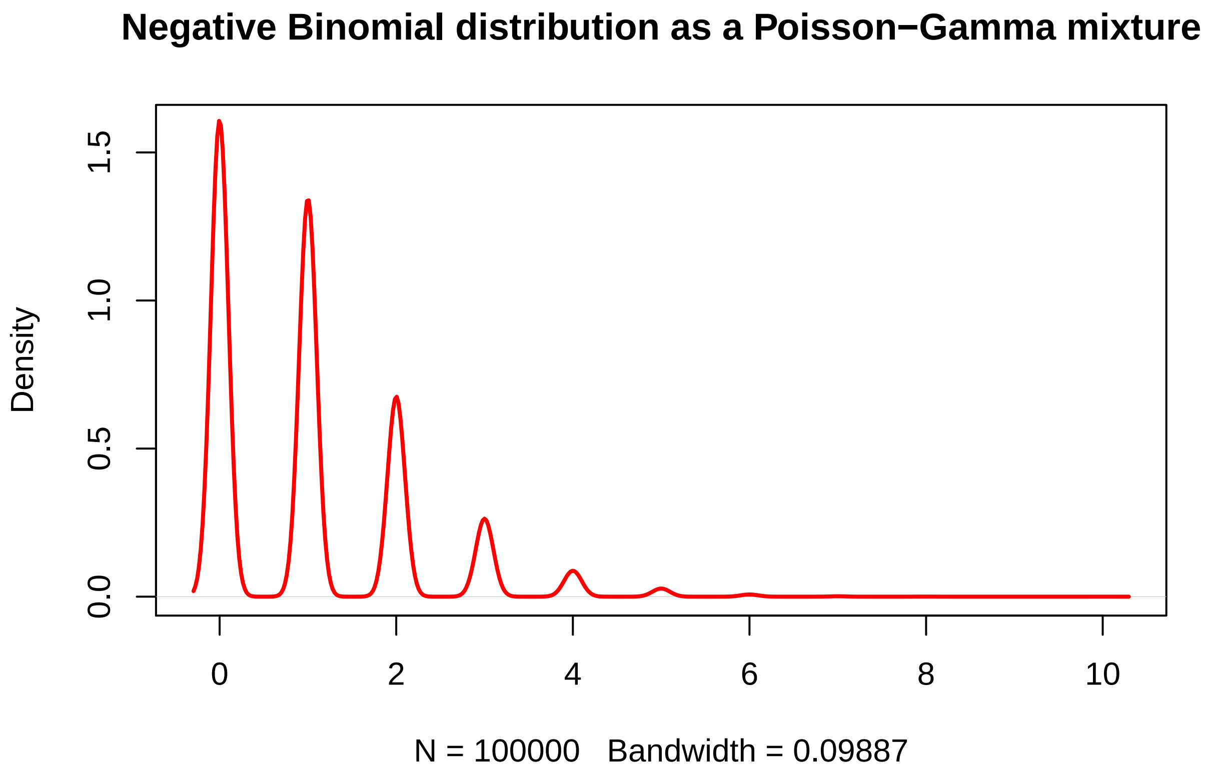


Exercise 5

Analogously, show with a simple R function that a negative binomial distribution may be seen as a mixture between a Poisson and a Gamma. In symbols: X|Y~P(Y), Y~Gamma(α, β), then X~. . . .

Answer: The negative binomial distribution can be viewed as a Poisson distribution where the Poisson parameter λ is itself a random variable, distributed according to a Gamma distribution.

```
set.seed(2019)

# Mixture of Poisson and Gamma distributions:
mixture <- function(df, n)
{
  Lambda = rgamma(n, df, df)
  X = rpois(n, Lambda)
  return(X)
}

df <- 5
n <- 100000

# Plot of the mixture distribution - PoissonGamma - in black:
PoissonGamma <- mixture(df,n)
plot( density(PoissonGamma), col="black", lwd=1,
main="Negative Binomial distribution as a Poisson-Gamma mixture")

# Negative Binomial distribution superposition - in red:
Nbinom <- c(rnbinom(n, size=df, prob=df/(df+1)))
lines( density(Nbinom), col="red", lwd=2 )
```



```
# Comparison between the mean and variance:

# Mean PoissonGamma:
mean(PoissonGamma)
```

```
## [1] 1.00681
```

```
# Mean Negative Binomial:
mean(Nbinom)
```

```
## [1] 1.00469
```

```
# Variance PoissonGamma:
var(PoissonGamma)
```

```
## [1] 1.206816
```

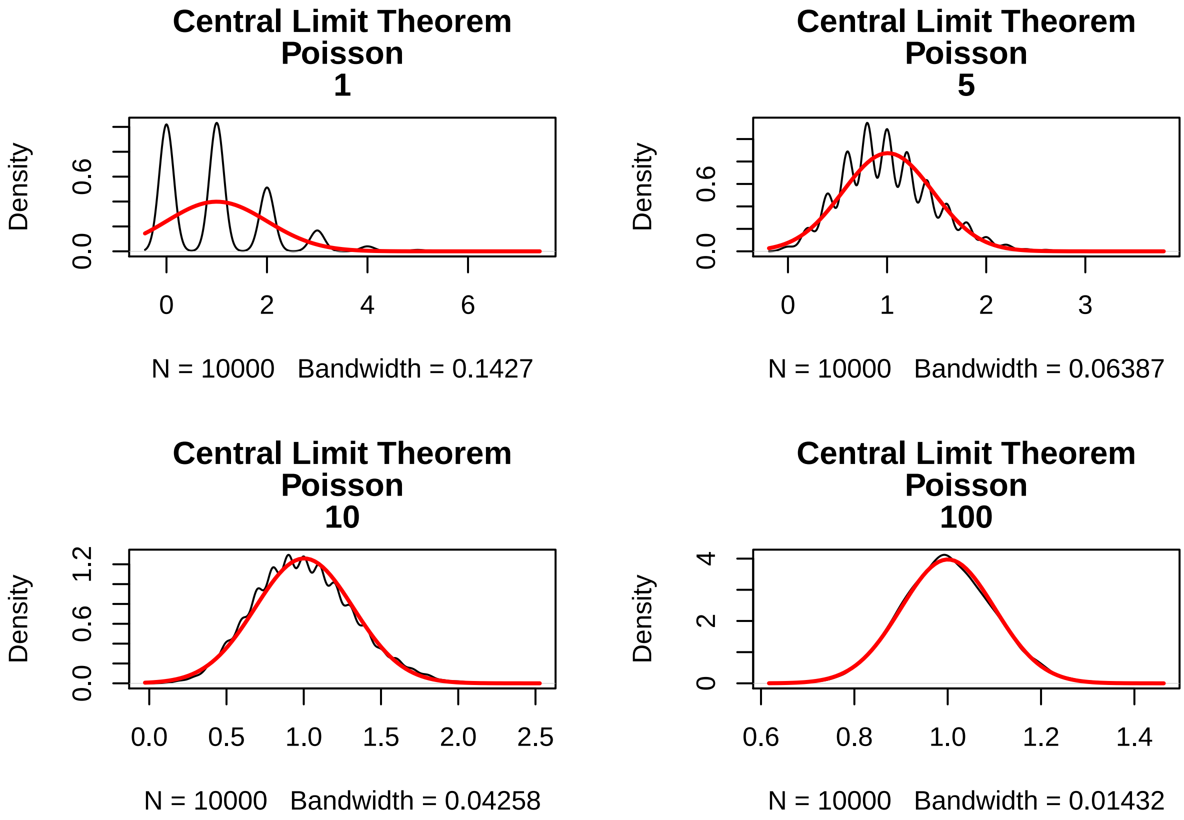
```r
# Variance Negative Binomial:
var(Nbinom)
```

```
## [1] 1.20212
```

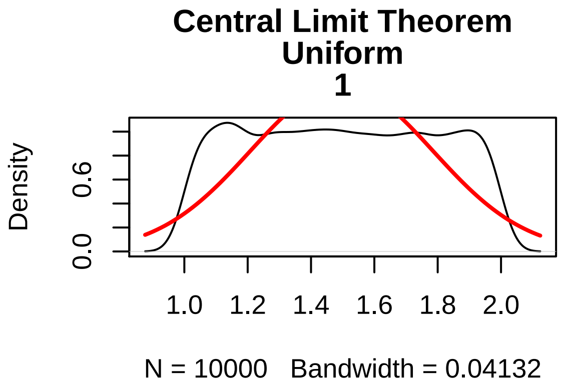
Exercise 8

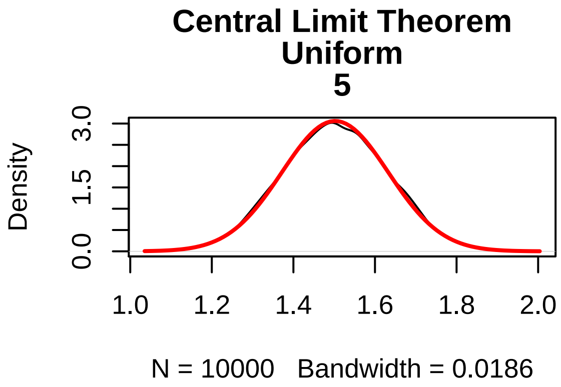
Write a general R function for checking the validity of the central limit theorem. Hint The function will consist of two parameters: clt_function <- function(n, distr), where the first one is the sampe size and the second one is the kind of distribution from which you generate. Use plots for visualizing the results.

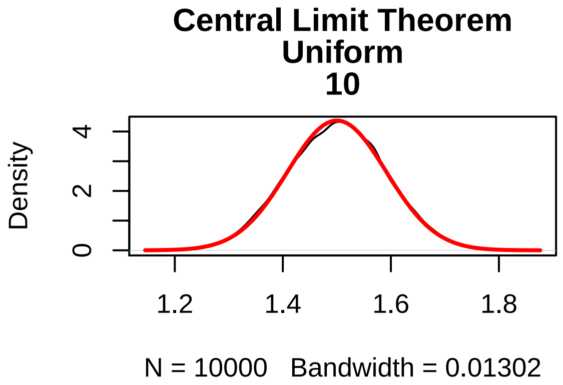
```r
clt_function <- function(n, src.dist = NULL, param1 = NULL, param2 = NULL)
  {

  r <- 10000  # Number of samples

  # matrix r x n. Each row is considered one sample:
  samples <- switch( src.dist,
                     "ChiSqrt"     = matrix( rchisq( n*r,param1 ),r ),
                     "Exponential" = matrix( rexp( n*r,param1 ),r ),
                     "Gamma"       = matrix( rgamma( n*r,param1,param2 ),r ),
                     "Normal"      = matrix( rnorm( n*r,param1,param2 ),r ),
                     "Poisson"     = matrix( rpois( n*r,param1 ),r ),
                     "Uniform"     = matrix( runif( n*r,param1,param2 ),r )
                    )

  means_of_samples <- apply( samples, 1, mean )

  # Distribution:
  plot( density( means_of_samples ), col = "black", lwd = 1,
        main = c( "Central Limit Theorem", src.dist, n ))

  # Gaussian:
  sigma <- sd  ( means_of_samples )
  mu    <- mean( means_of_samples )
  curve( dnorm(x, mu, sigma), add = TRUE, col="red", lwd=2 )

}

# Example 1 parameter - Poisson distribution
par(mfrow=c(2,2))
for (i in c(1,5,10,100))
{
   clt_function(i,src.dist="Poisson",param1=1)
}
```

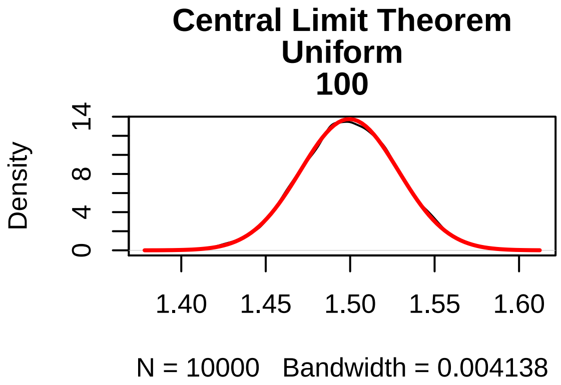


```
# Example 2 parameters - Uniform distribution
for (i in c(1,5,10,100))
{
    clt_function(i,src.dist="Uniform",param1=1, param2=2)
}
```









Answer: Analysing the graphs we can easily note that the distributions tend to a Gaussian with the increase of the sample size.