Notes FSML II*

Tobías Chavarría

DSTI | DSBD2-001

Contents

Introduction

Statistics notation:

1. If $X_1, \ldots, X_n$ are random variables (r.v).
2. $x_1, \ldots, x_n$ are observations.
3. If we write *i.i.d* means that the r.v are independent and identically distributed.

First aim: To propose a model for a random variable.

Generalization to multi-dimensional case:

- Y: response variable.
- $X^{(1)}, \ldots, X^{(p)}$: explanatory variables.

Aim: To find a functional link between Y and the explanatory variables.

To find this functional link , the method to apply depends on the nature of the r.v's.

Y	Model
Numeric	Linear model
Qualitative (labels)	Classification

Linear model

A linear model is given by:

$Y_i = \beta_0 + \beta_1 X_i^1 + \ldots + \beta_p X_i^p + \varepsilon_i$

where:

- $\beta_0, \ldots, \beta_p$ are unknown *fixed* parameters that can be estimated by two methods:
  - Point estimation

*Replication files are available on the author's Github account (http://github.com/svmiller/svm-r-markdown-templates).

- Confidence interval
- ε is the noise and also a random variable.

Chapter 1: Estimation for one parameter

Previous Knowledge

- Random Variable:
- The notion of distribution.
- The expectation and variance
- The distribution function
- The classical distributions (in particular the Gaussian)
- The Law of Large numbers and the Central Limit theorem

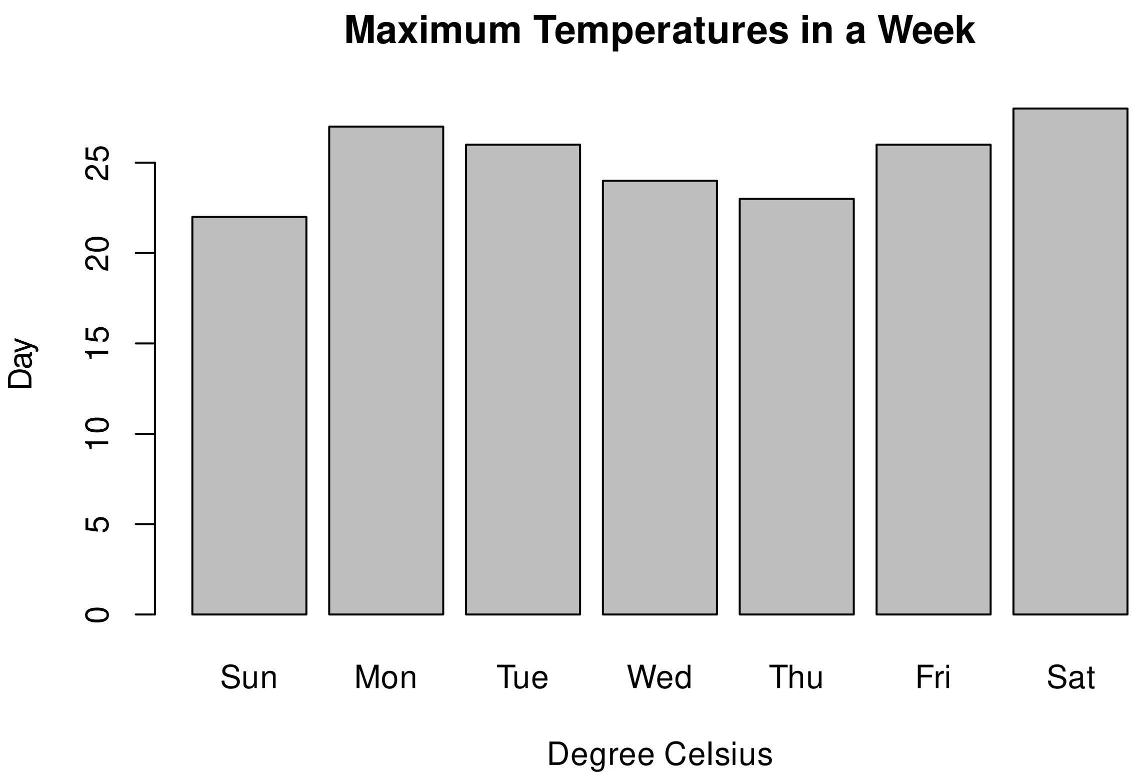
Introduction

Given $x_1, \ldots, x_n$ numeric observations, to try to find a correct parametric model, we can use 2 graphs:

Plot type	Variable type	Density
Bar plot	Discrete	count/n
Histogram	Continuous	count for a bin/ n x (length of the bin)

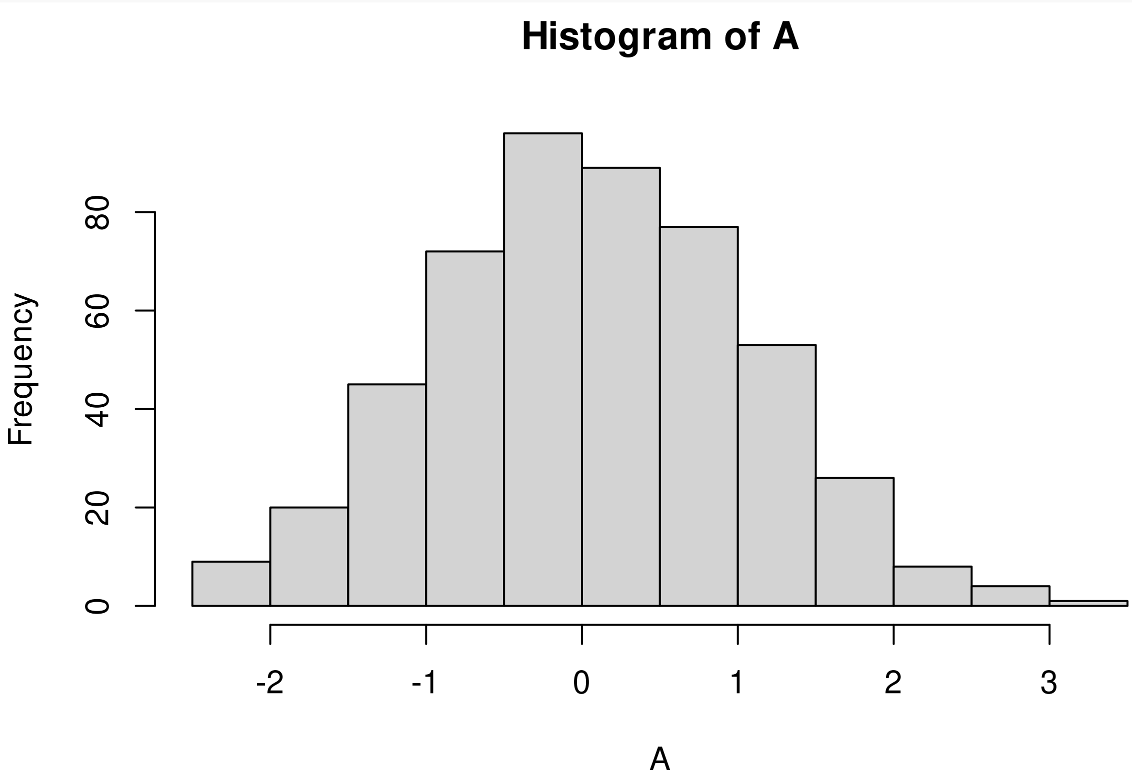
Barplot for discrete variables

```
max.temp <- c(22, 27, 26, 24, 23, 26, 28)

barplot(max.temp,
main = "Maximum Temperatures in a Week",
xlab = "Degree Celsius",
ylab = "Day",
names.arg = c("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"))
```

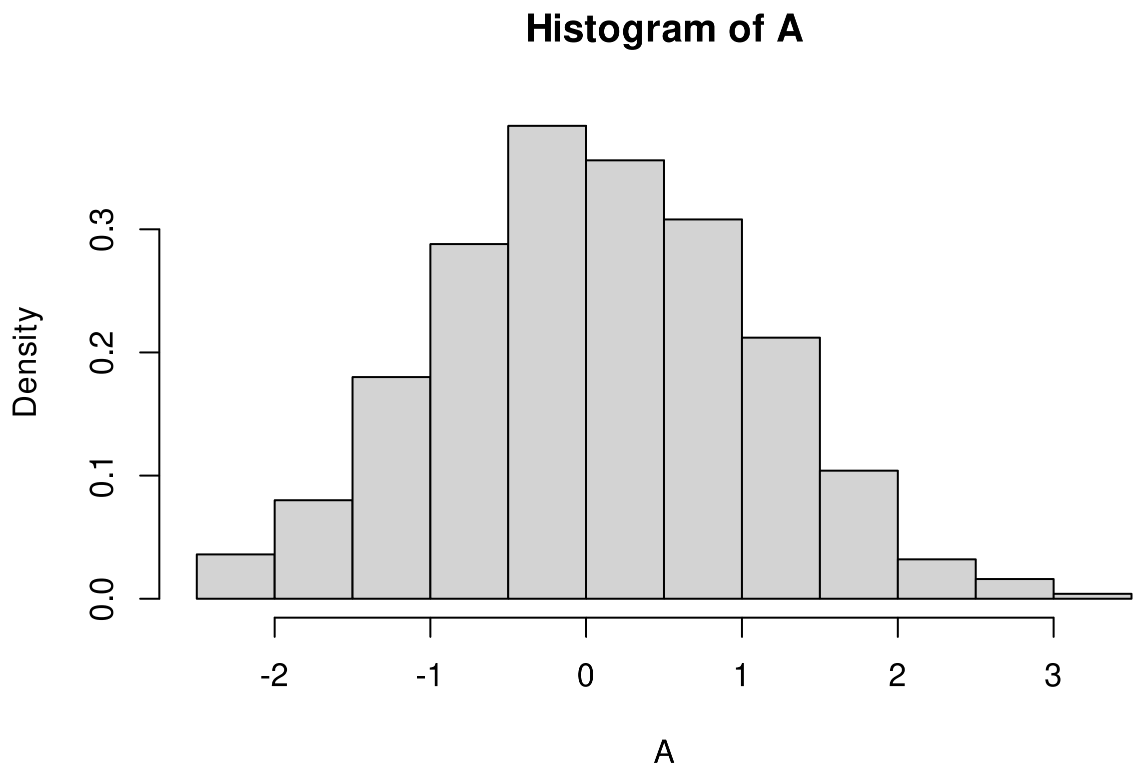


Histogram for continuous variables

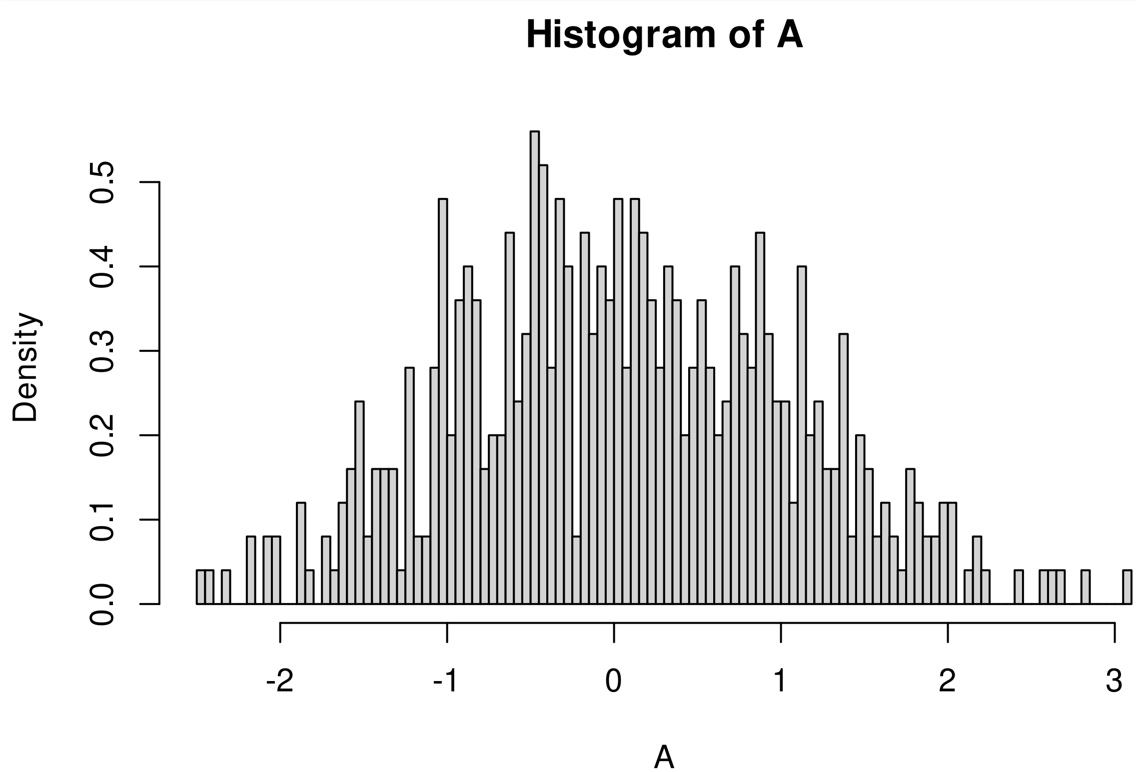
```
A <- rnorm(500, 0, 1)

## The default execution of this function doesn't generate a density:
hist(A)
```



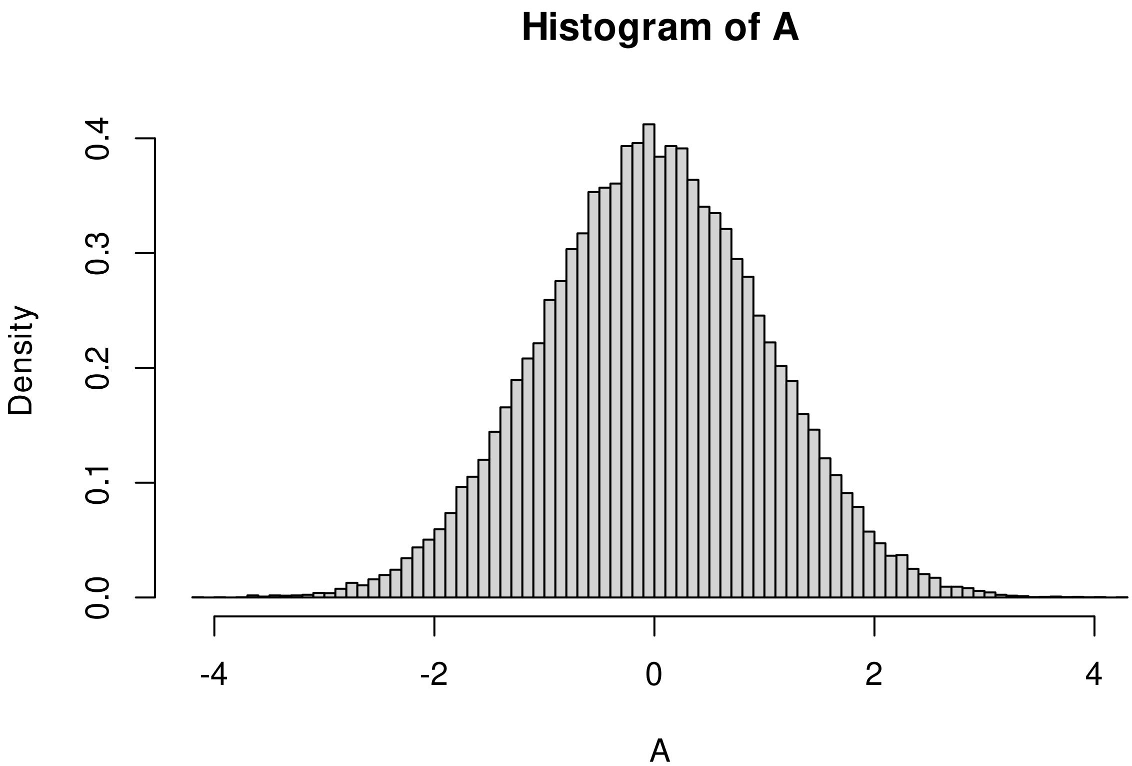
```r
## You need to set freq = FALSE:
hist(A, freq = FALSE)
```




```r
## You can set the numbers of bins that you want to use:
hist(A, freq = FALSE, breaks = 100)
```



```r
# But in order to create more breaks, you need to increase the
# numbers of observations:


A <- rnorm(50000, 0, 1)
hist(A, freq = FALSE, breaks = 100)
```

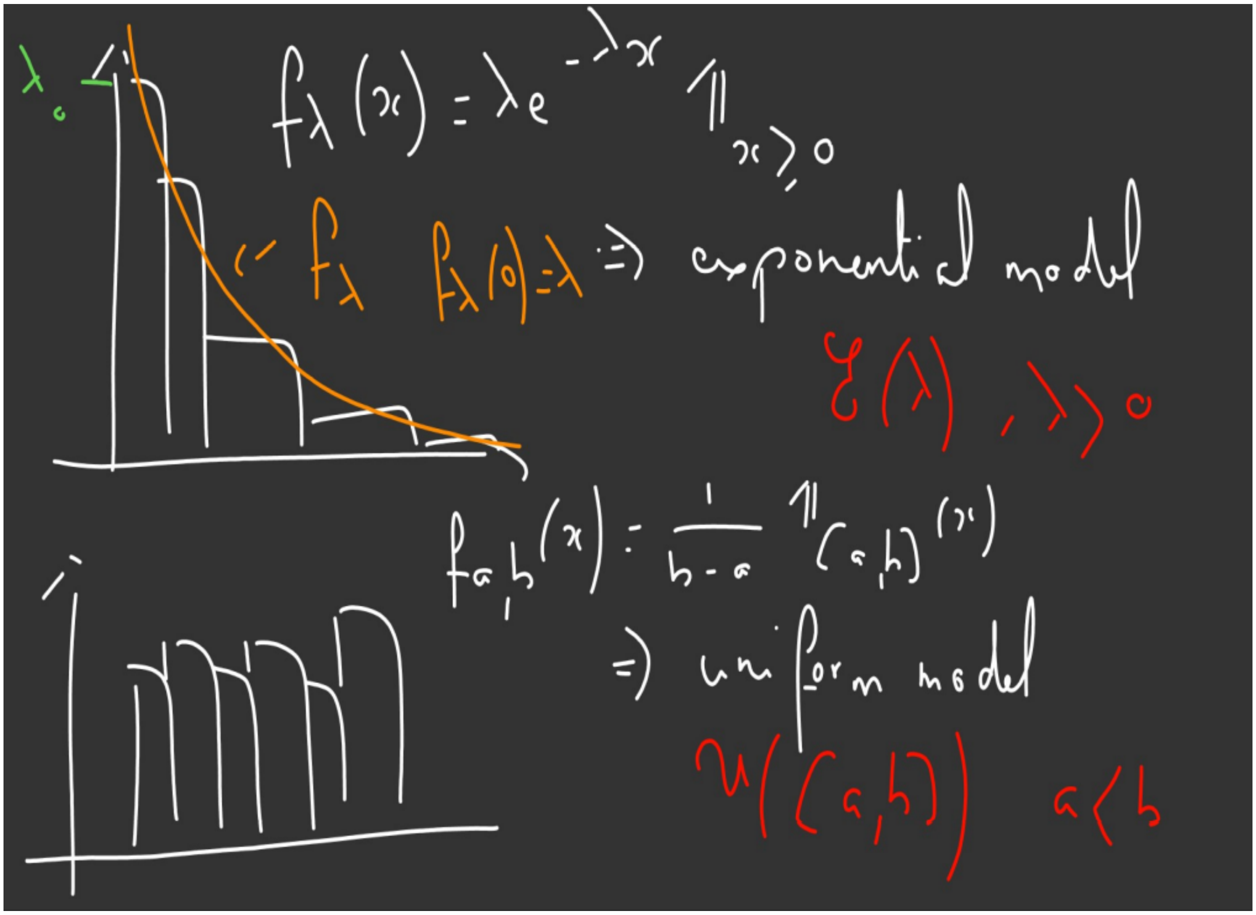


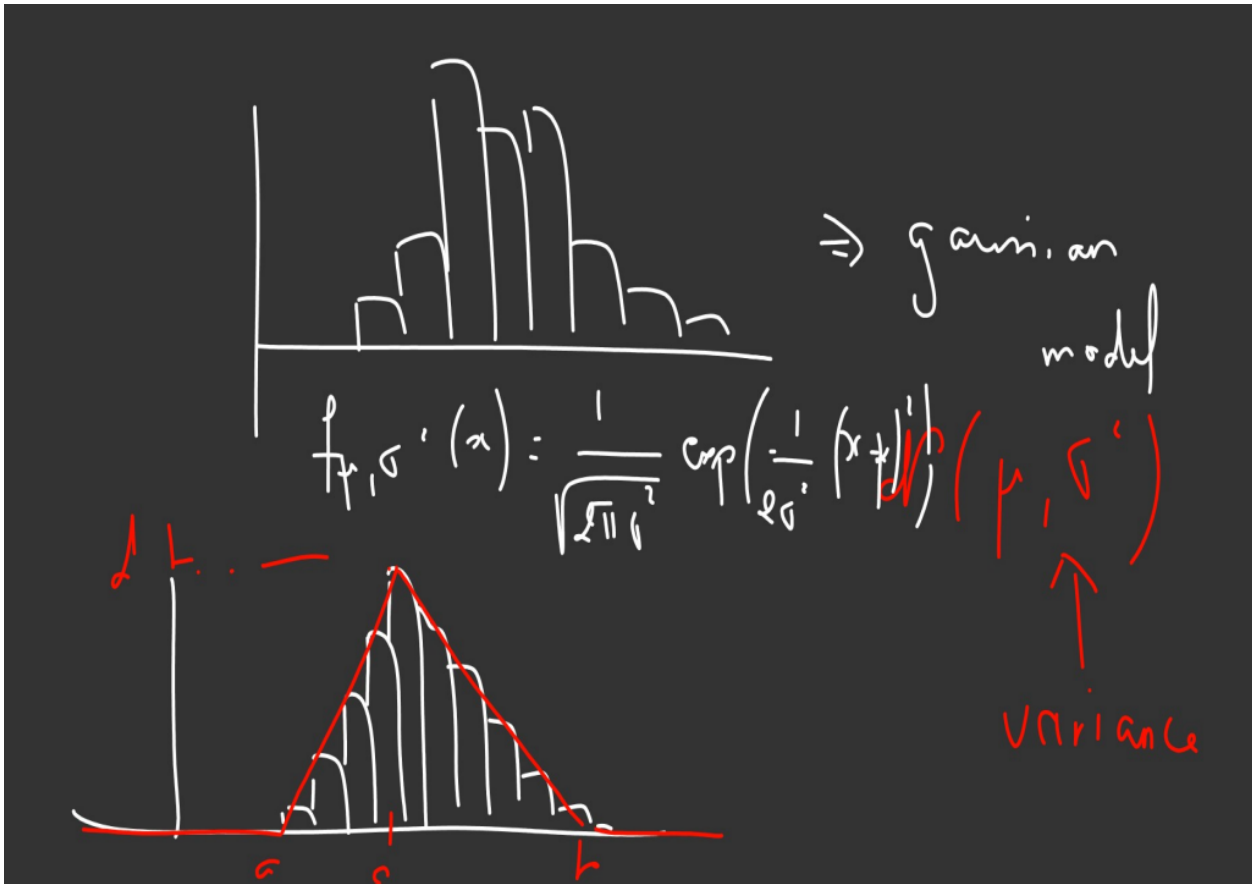
To propose a parametric model:

1. Make a graphical representation of the observations.
2. Guess a theoretical model by looking the previous graphic.

Examples:

Pasarlos a R!!

Example 2: 
$f_\lambda(x) = \lambda e^{-\lambda x} \mathbb{1}_{x \geq 0}$
λ_0
f_λ $f_\lambda(0) = \lambda$ => exponential model
$\mathcal{E}(\lambda)$, $\lambda > 0$
$f_{a,b}(x) = \frac{1}{b-a} \mathbb{1}_{[a,b]}(x)$
=> uniform model
$\mathcal{U}([a,b])$ $a < b$

Example 2: 

Question: with a representation, we can guess a parametric family of models, denoted by $\{P_\Theta, \theta \in \Theta\}$. How to guess a correct value for θ thanks to the observations?

Answer: *Estimation*.

Point estimation

Let x_i an observation of a r.v X_i we assume that $X_1, \ldots, X_n$ are *i.i.d* with common distribution P_θ.

Estimator **Definition:** An estimator of Θ is just a function of $X_1, \ldots X_n$ that **does not depend onto others unknown parameters.**

Remark: An estimator is a random variable!

Estimation **Definition:** An estimation is the value of an estimator computed thanks to the observations.

Example

Consider $X_1, \ldots, X_n$ exponential distributed and *i.i.d*, an *estimator* of λ is $\hat{\lambda}_n = \frac{n}{\sum X_i}$ an *estimation* is $\hat{\lambda}_n = \frac{n}{\sum x_i}$.

Distribution	Parameter	Estimator	Estimation
Exponential $\xi(\lambda)$	λ	$\frac{n}{\sum X_i}$	$\frac{n}{\sum x_i}$

Bias (for univariate parameter) **Definition:** Let consider $\hat{\theta}_n$ an estimator of θ.

The bias of $\hat{\theta}_n$ is defined by:

$$b(\hat{\theta}_n) := \mathbb{E}(\hat{\theta}_n) - \theta$$

- We say that $\hat{\theta}_n$ is an unbiased estimator if

$$\forall n \in \mathbb{N}^+ \quad b(\hat{\theta}_n) = 0$$

- We say that $\hat{\theta}_n$ is *asymptotic unbiased* estimator if:

$$b(\hat{\theta}_n) \to 0 n \to +\infty$$

How to construct estimator?

- Method of moments
  - less computations
  - based on the Law of large numbers
- Maximum likelihood

Method of moments Let θ a parameter to estimated, parameter which is associate to $X_1, ..., X_n$ *i.i.d* r.v.

Let consider $k \in \mathbb{N}^*$:

- the moment of order k : $\mathbb{E}[X^k]$
- the centered moment of order k: $\mathbb{E}[X - \mathbb{E}[X]]^k$

If there exist a value k such that:

- (a) $\mathbb{E}[X^k] = g(\theta)$
- (b) $\mathbb{E}[[X - \mathbb{E}[X]]^k] = h(\theta)$

Then the estimator $\hat{\theta}_n$ of θ is solution of:

- (a) $g(\hat{\theta}_n) = \frac{1}{n} \sum X_i^k$
- (b) $h(\hat{\theta}_n) = \frac{1}{n} \sum (X_i - \overline{X}_n)^k$

where

$$\overline{X}_n = \frac{1}{n} \sum X_i$$

is the **empirical mean**.

Remark: Exponential Distribution

If Let $X \sim \xi(\lambda)$, then:

- $f_\lambda(x) = \lambda \exp(-\lambda x) 1_{x \geq 0}$
- $\mathbb{E}[X] = \int_{\mathbb{R}} x \cdot f_\lambda(x) dx$
- $V[X] = \mathbb{E}[X^2] - (\mathbb{E}[X])^2$

Transfer formula:

$$\mathbb{E}[l(X)] = \int_{\mathbb{R}} l(x) \cdot f_\lambda(x) dx$$

Applications

1. Let consider $X_1, \ldots, X_n$ exponential distributed and *i.i.d* compute an estimator of λ using the methods of moments.

Solution:

Let $X \sim \xi(\lambda)$, so:

$$\mathbb{E}[X] = \frac{1}{\lambda} \quad \text{and} \quad V[X] = \frac{1}{\lambda^2}$$

By applying the method of moments ($k = 1$) we get:

$\frac{1}{\hat{\lambda}_{n,1}} = \frac{1}{n} \sum X_i$

Thus:

$$\hat{\lambda}_{n,1} = \frac{n}{\sum X_i}$$

in the same way but using the variance ($k = 2$), we get:

$\frac{1}{\hat{\lambda}^2_{n,2}} = \frac{1}{n} \sum (X_i - \overline{X}_n)^2$

Thus:

$$\hat{\lambda}_{n,2} = \frac{\sqrt{n}}{\sqrt{\sum (X_i - \overline{X}_n)^2}}$$

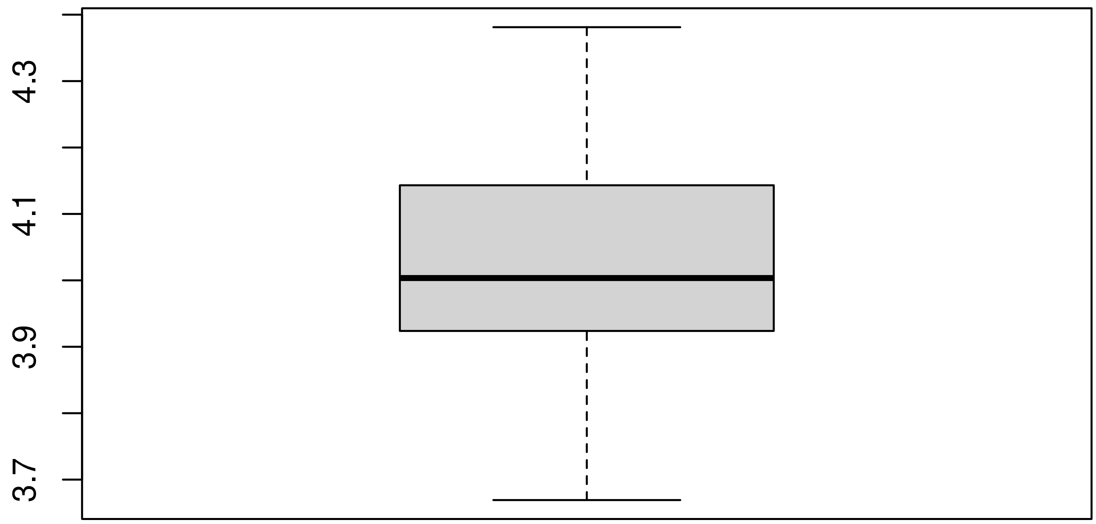
```
A = rexp(500, 4)

1/mean(A)
```

```
## [1] 3.874977
```

```
m = c()

for (i in 1:50) {
  A = rexp(500, 4)
  m[i] <- 1/mean(A)
}


mean(m)
```

```
## [1] 4.022919
```

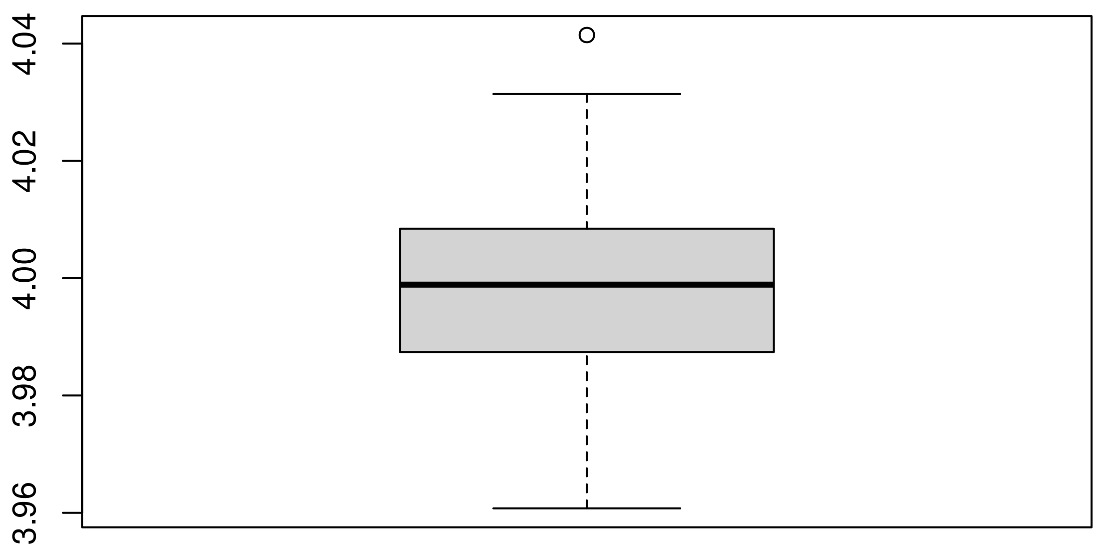
```
boxplot(m)
```




```r
## With more observations we got less variation (500 -> 5000)

## Law of large numbers

m = c()

for (i in 1:50) {
  A = rexp(50000, 4)
  m[i] <- 1/mean(A)
}


mean(m)
```

```
## [1] 3.99764
```

```r
boxplot(m)
```



2. Let consider $X_1, \ldots, X_n$ *i.i.d* $\mathcal{U}([0, \sigma])$.

- Determine an estimator of σ using the methods of moments.

- Let denote $\hat{\sigma}_n = 2\overline{X}_n$. Is $\hat{\sigma}_n$ an unbiased estimator?

Solution:

The Maximum Likelihood

Likelihood **Definition:** Let $X_1, \dots, X_n$ independent random variables, whose distributions are all depending on the same parameter Θ.

Let $x_1, \dots, x_n$ observations of those r.v

$$\mathcal{L}(x_1, \dots, x_n, \Theta) = \begin{cases} \Pi \\ \Pi \end{cases}$$

Def: Estimator thanks to the maximum likelihood.

$\hat{\Theta}_n$, an estimator for Θ, due to the maximum likelihood, is solution of:

$$\mathcal{L}(x_1, \dots, x_n, \Theta) = \max_{\theta} \mathcal{L}(x_1, \dots, x_n, \Theta)$$

Applications

Let consider $X_1, \dots, X_n$ $\xi(\lambda)$ *i.i.d.*

Compute the maximum likelihood estimator.

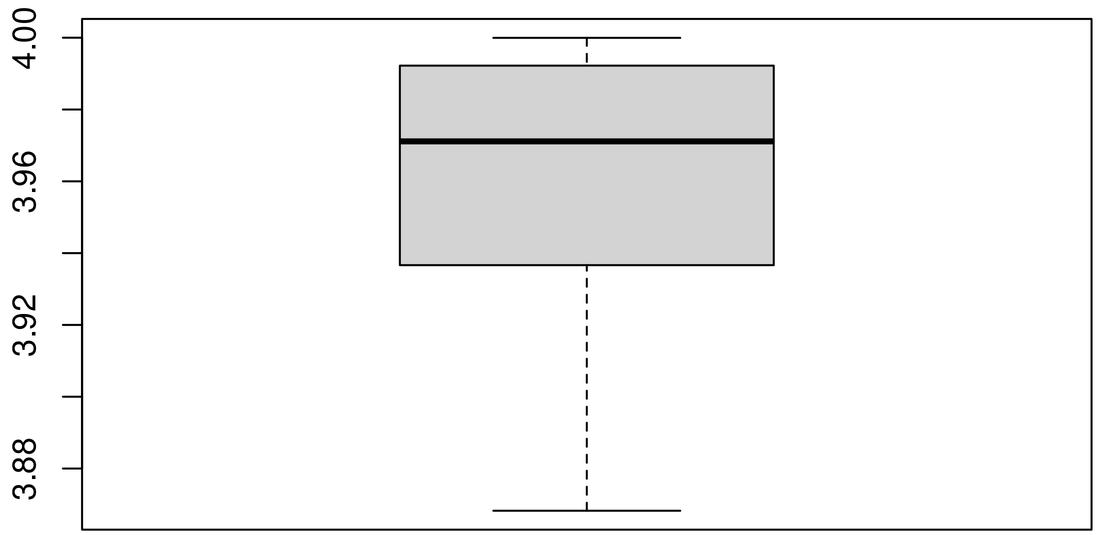
Solution:

…

```
n= 100

U = runif(n, 0, 4)

theta = max(U)


for (i in 1:50) {
  U = runif(n, 0, 4)
  theta = c(theta, max(U))
}


mean(theta)
```

```
## [1] 3.959668
```

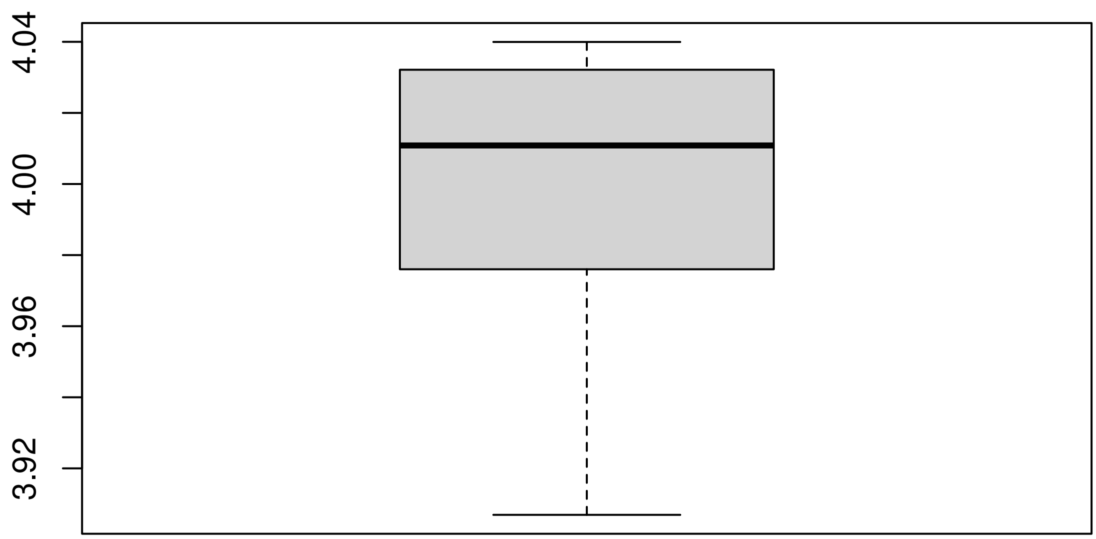
```
boxplot(theta)
```




```r
## Adjust to make the estimator unbiased

thetab = (n+1)/n*theta

mean(thetab)
```

```
## [1] 3.999264
```

```r
boxplot(thetab)
```

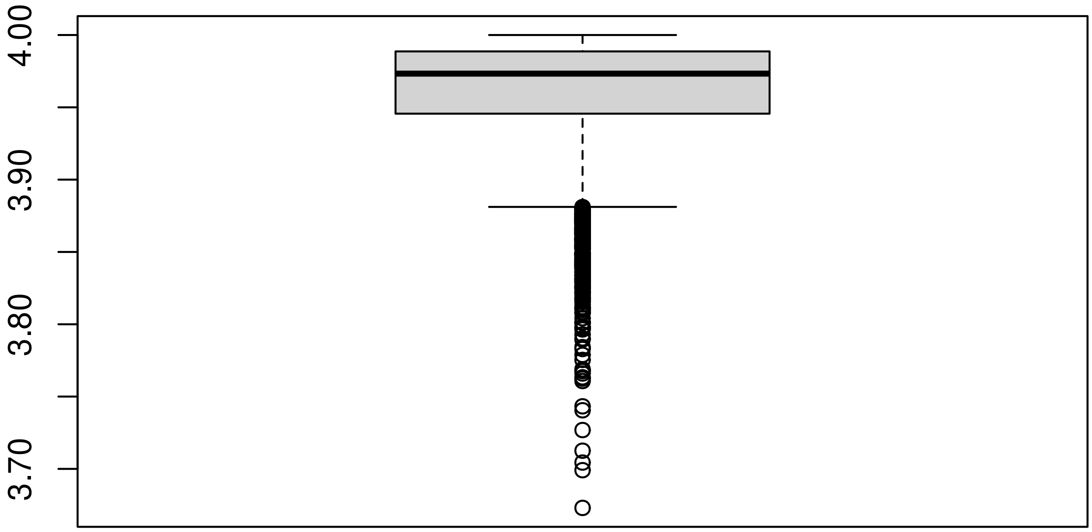



```r
## With more observations

n= 100

U = runif(n, 0, 4)

theta = max(U)


for (i in 1:5000) {
  U = runif(n, 0, 4)
```

```
  theta = c(theta, max(U))
}


mean(theta)
```

```
## [1] 3.960752
```

```
boxplot(theta)
```



Property

Let $X_1, \ldots, X_n$ *i.i.d* r.v. Let $\mu = \mathbb{E}[X_1]$ (unknown) Let $\sigma^2 = V(X_1)$ (unknown)

A classical estimator is:

- μ is $\hat{\mu}_n = \overline{X}_n = \frac{1}{n} \sum X_i$
- σ^2 is $\hat{\sigma}^2_n = \frac{1}{n} \sum (X_i - \overline{X}_n)^2$

Exercise

Show that:

1. $\hat{\mu}$ is unbiased.
2. $\hat{\sigma}^2_n$ is biased and that $\dfrac{n}{n-1}\hat{\sigma}^2_n$ is unbiased.

– Day 2 –

Solution exercise Theorem

Quality of an estimator

def: Let θ an unknown parameter, let $\hat{\theta}_n$ an estimator of θ, mean quadratic error is given by:

Property:

$$MQE = V[\hat{\theta}_n] + ((b(\hat{\theta}_n)))^2$$

Proof:

$$MQE(\hat{\theta}_n) = \ldots \tag{1}$$

##theory##

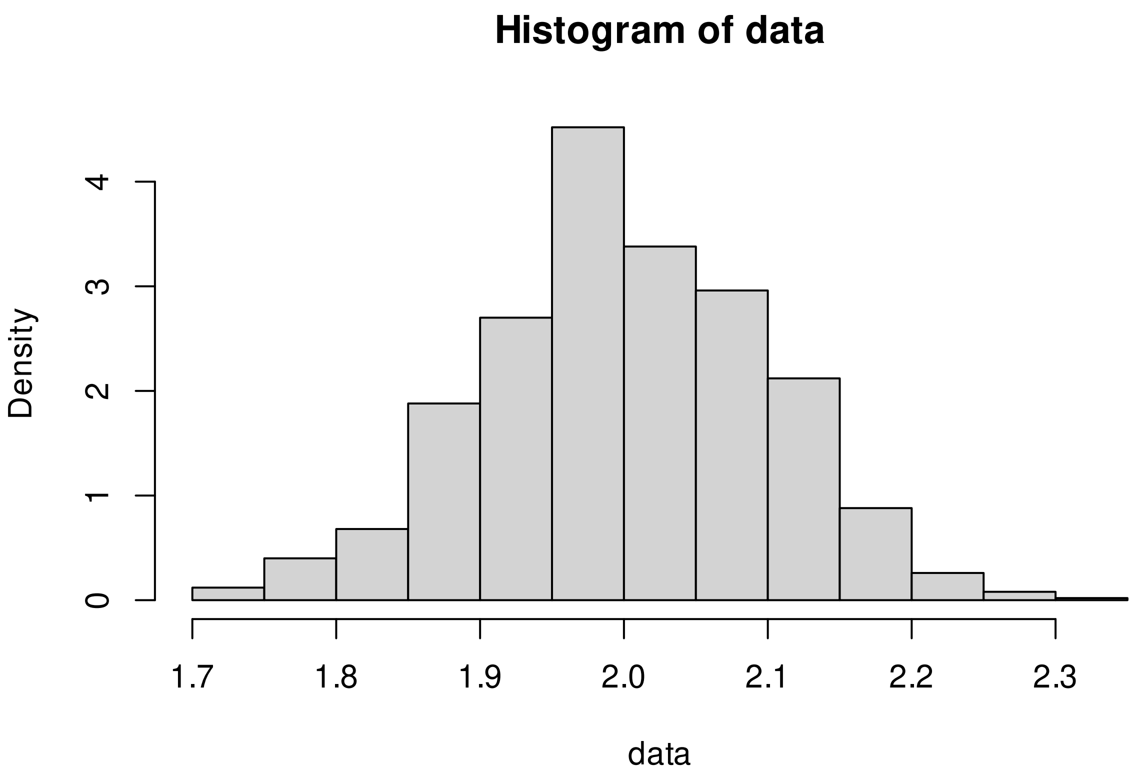
Practical class

1. Load the data in R software.
2. Propose a model for the variables associated to this file.

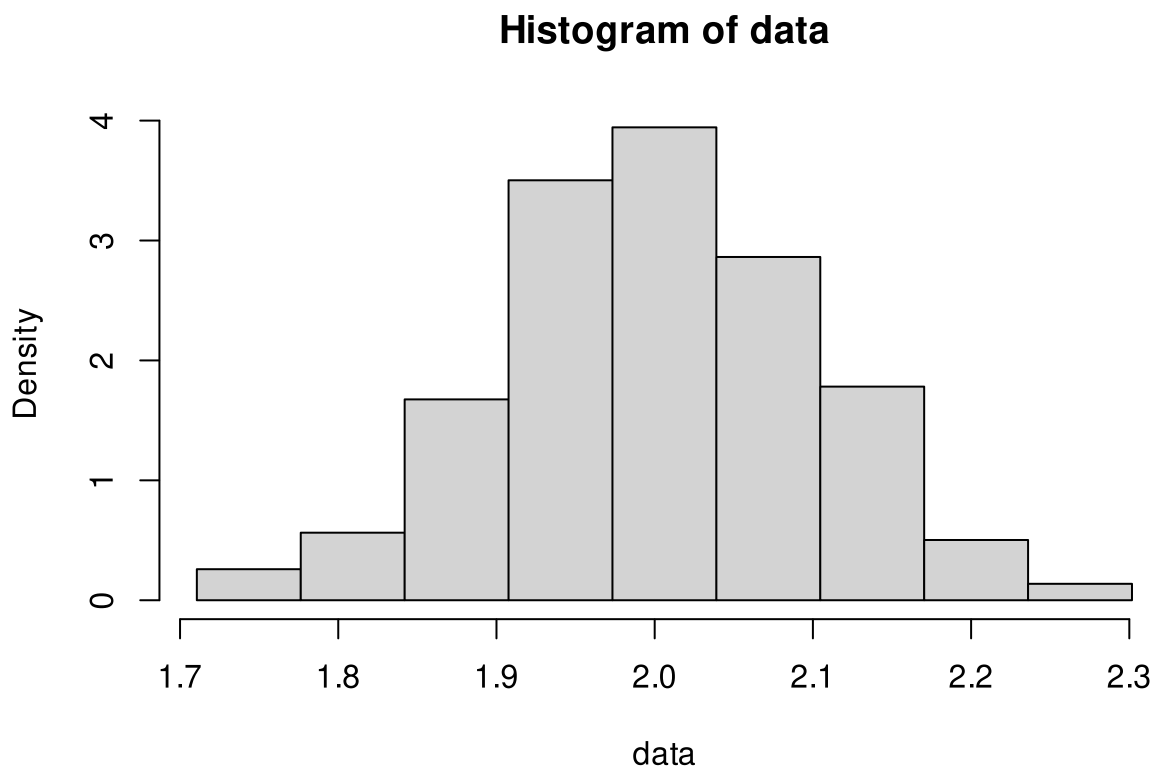
- Make a visualization of this.

```
data <- as.matrix(read.table("/cloud/project/docs/data/data1.txt"))

## Some comments about the data and data types.
```

```
hist(data, freq = FALSE)
```




```
## Be careful about getting conclusions

data1_min = min(data)
data1_max = max(data)

hist(data, freq = FALSE, breaks = seq(data1_min, data1_max, length = 10))
```



We guess a Gaussian distribution.

How to estimate the parameters of the distribution?

1. Compute the empirical mean and statistical variance.
2. Plot the theoretical density that we guess.
3. Test the goodness of fitness.

Compute the empirical mean and statistical variance.

```
mhu <- mean(data)

# When you compute the variance be careful and read the documentation
# in this case we have:
# "The denominator n - 1 is used which gives an unbiased estimator of the (co)variance
# for i.i.d. observations"

sigma2 <- var(data)

## You also can do it manually:

sigma2 <- 1/(nrow(data)-1)*sum((data - mean(data))^2)
```

Plot the theoretical density that we guess.

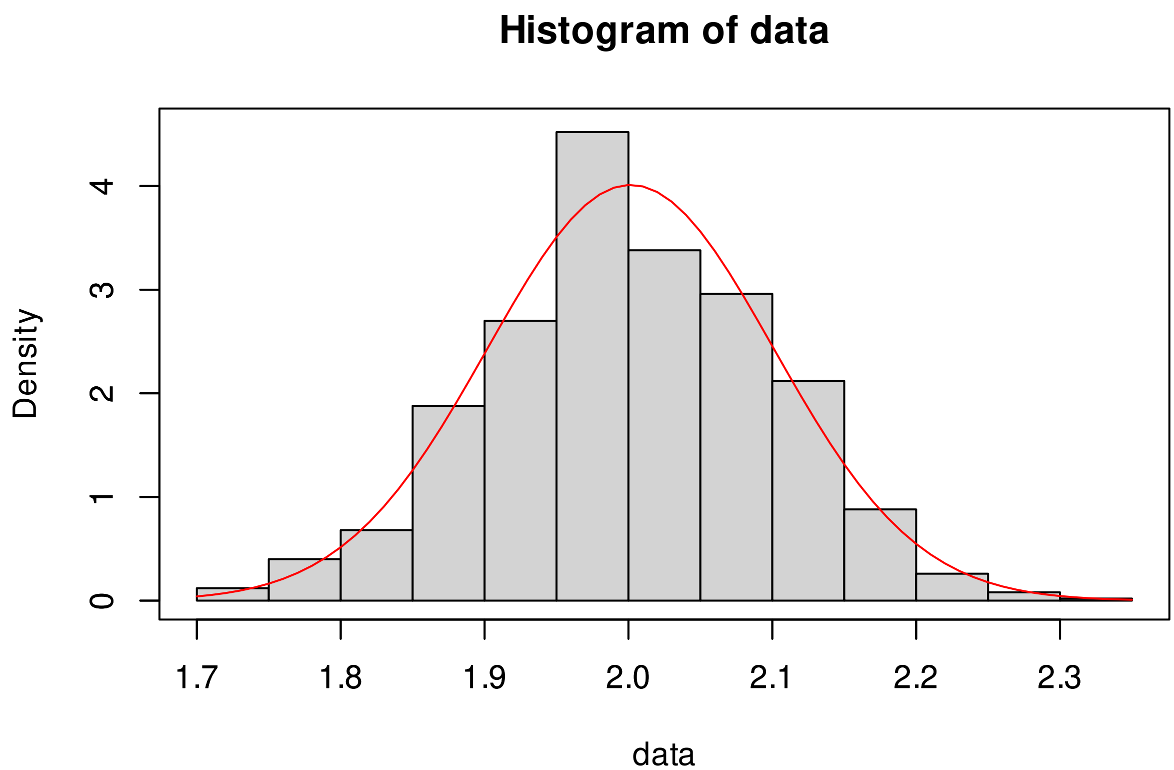
```
## This generates a list, with all the hist information.

H <- hist(data, freq = FALSE, plot = FALSE) ## ignore warning message
```

```
## Warning in hist.default(data, freq = FALSE, plot = FALSE): argument 'freq' is
## not made use of
```

```r
H
```

```
## $breaks
##  [1] 1.70 1.75 1.80 1.85 1.90 1.95 2.00 2.05 2.10 2.15 2.20 2.25 2.30 2.35
##
## $counts
##  [1]   6  20  34  94 135 226 169 148 106  44  13   4   1
##
## $density
##  [1] 0.12 0.40 0.68 1.88 2.70 4.52 3.38 2.96 2.12 0.88 0.26 0.08 0.02
##
## $mids
##  [1] 1.725 1.775 1.825 1.875 1.925 1.975 2.025 2.075 2.125 2.175 2.225 2.275
## [13] 2.325
##
## $xname
## [1] "data"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

```r
## Histogram info to use.
limits <- H$breaks
lmin <- limits[1]
lmax <- limits[length(limits)]

## Create our plot

x <- seq(lmin, lmax, by = 0.01)
y <- dnorm(x, mhu, sqrt(sigma2))


y_max = max(y, H$density)

## Make the comparison

hist(data, freq = FALSE, xlim = c(lmin, lmax), ylim = c(0, y_max*1.01))
par(new = TRUE)
plot(x, y, type = 'l', col = 'red'
     , xlim = c(lmin, lmax)
     , ylim = c(0, y_max*1.01)
     , xlab = ""
     , ylab = "")
```



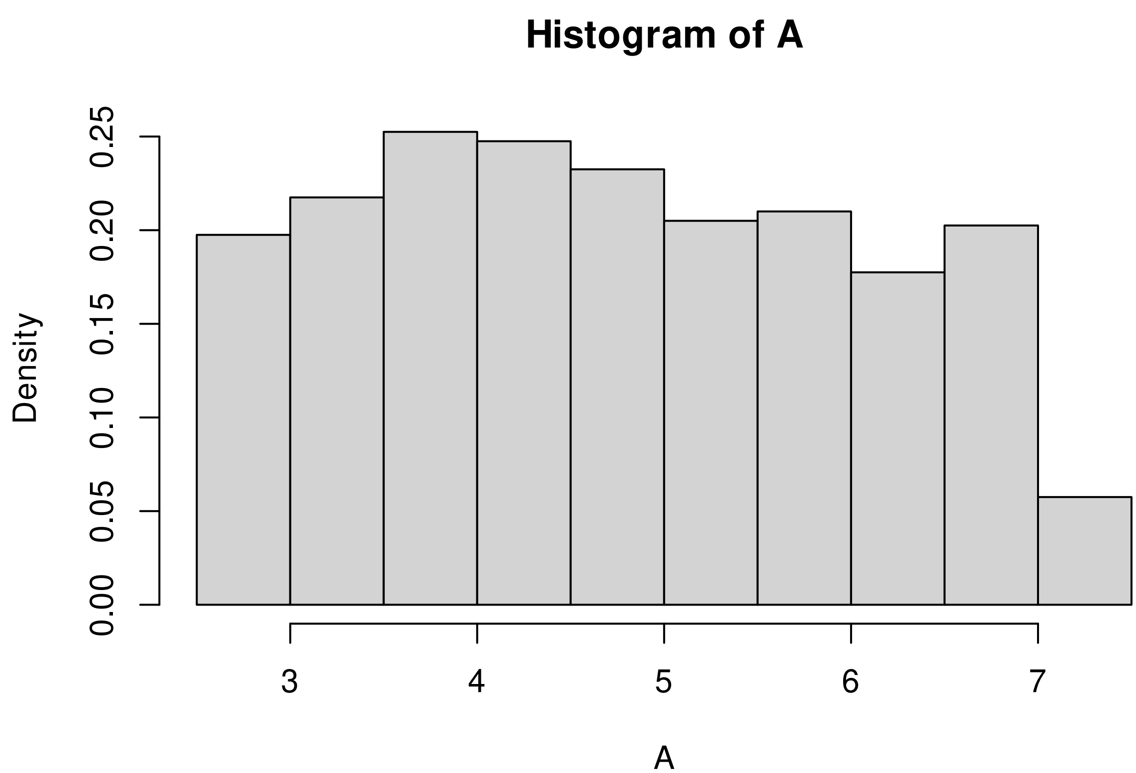
Test the goodness of fitness.

```
ks.test(x, 'pnorm', mhu, sqrt(sigma2))
```

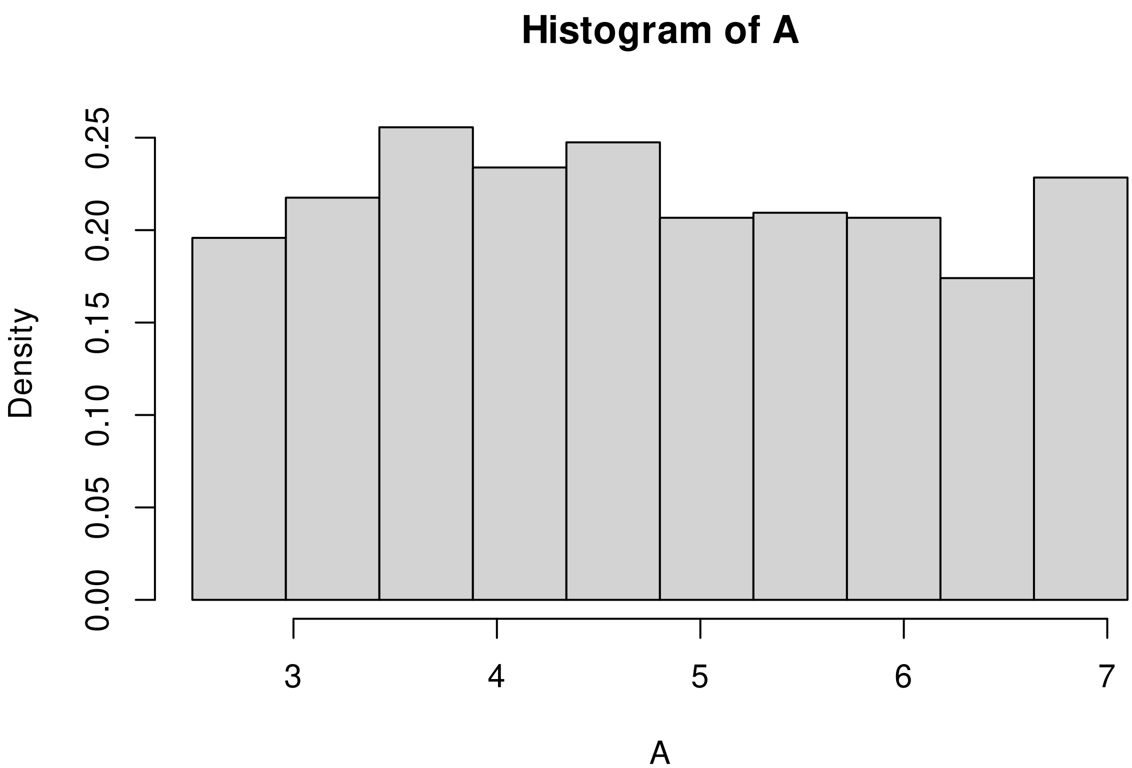
```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  x
## D = 0.25147, p-value = 0.0003644
## alternative hypothesis: two-sided
```

Data set 2

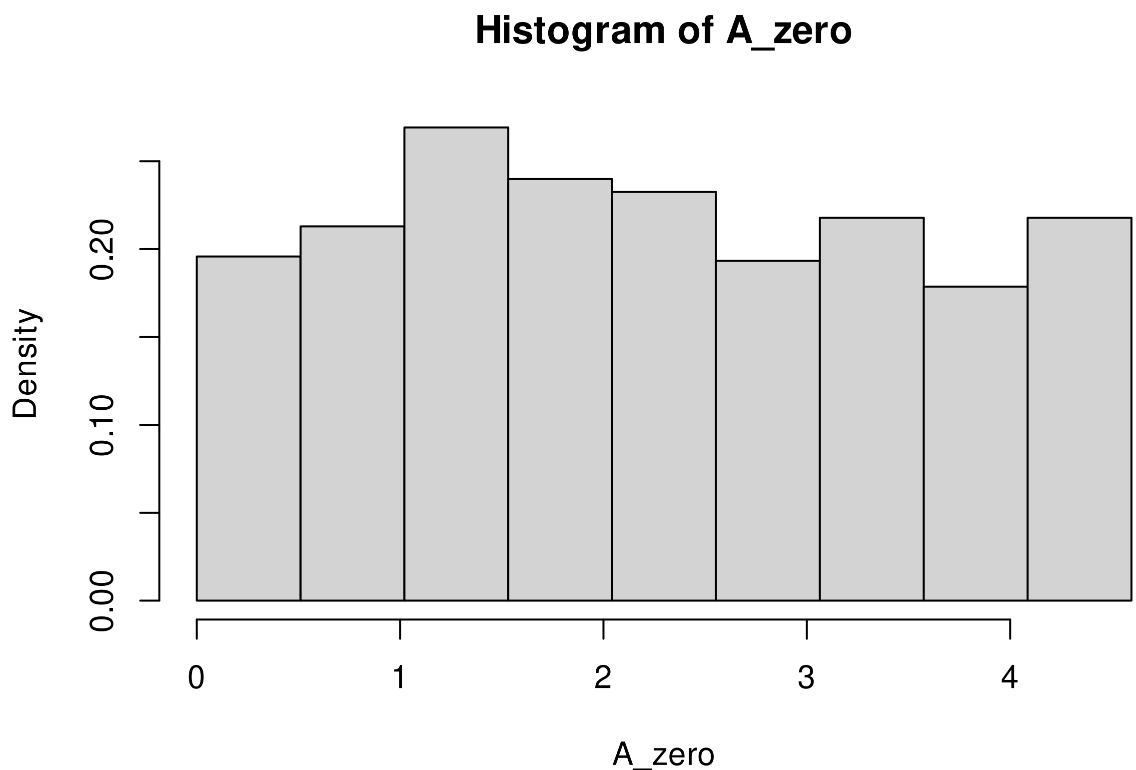
```
## Loading the data
A <- as.matrix(read.table("/cloud/project/docs/data/data2.txt"))


## Make visualizations

hist(A, freq = FALSE)
```




```
## The right box exist because r creates one class with just one element.

## There is a formula that generates the right amount of classes, given by:

## K aproxx 1 + 3.22*log(n, 10) ## (in practice take the floor)



## Use right parameters:

n = length(A)
K = floor(1 + 3.22*log(n, 10))
A_min = min(A)
A_max = max(A)

A_mp = (A_max - A_min)/K

epsilon = (A_max - A_min)/10^9

limits = seq(A_min, A_max, by = A_mp)


hist(A, freq = FALSE, breaks = limits)
```




```
## We guess is an uniform distribution

## Make start in 0

A_zero = A - min(A)

A_min = min(A_zero)
A_max = max(A_zero)
hist(A_zero, freq = FALSE, breaks = seq(A_min, A_max, length = 10))
```

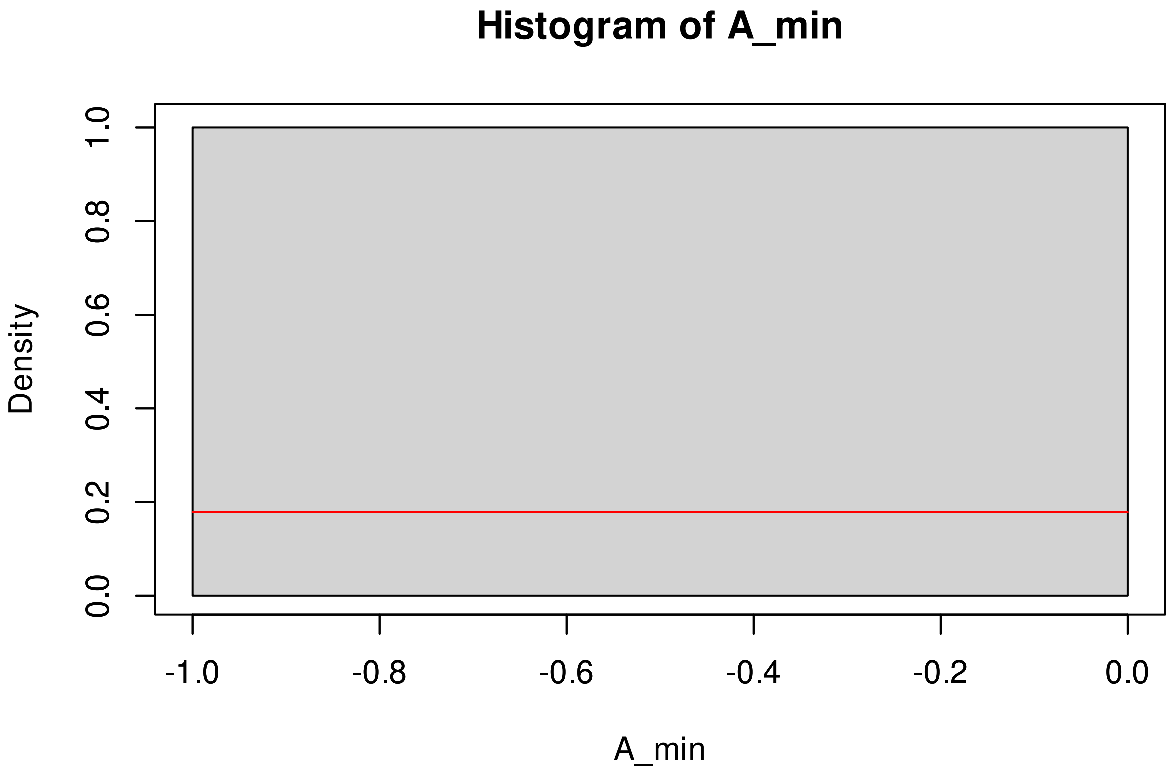



```r
## Estimate the parameter

n <- length(A_zero)

theta <- ((n+1)/n)*max(A_zero)

## Create our theoretical density plot


H <- hist(A_min, freq = FALSE, plot = FALSE) ## ignore warning message
```

```
## Warning in hist.default(A_min, freq = FALSE, plot = FALSE): argument 'freq' is
## not made use of
```

```r
## Histogram info to use.
limits <- H$breaks
lmin <- limits[1]
lmax <- limits[length(limits)]

## Create our plot

x <- seq(lmin, lmax, by = 0.01)
y <- dunif(x, min = lmin, max = theta )

y_max = max(y, H$density)

## Make the comparison

hist(A_min, freq = FALSE, xlim = c(lmin, lmax), ylim = c(0, y_max*1.01))
par(new = TRUE)
```

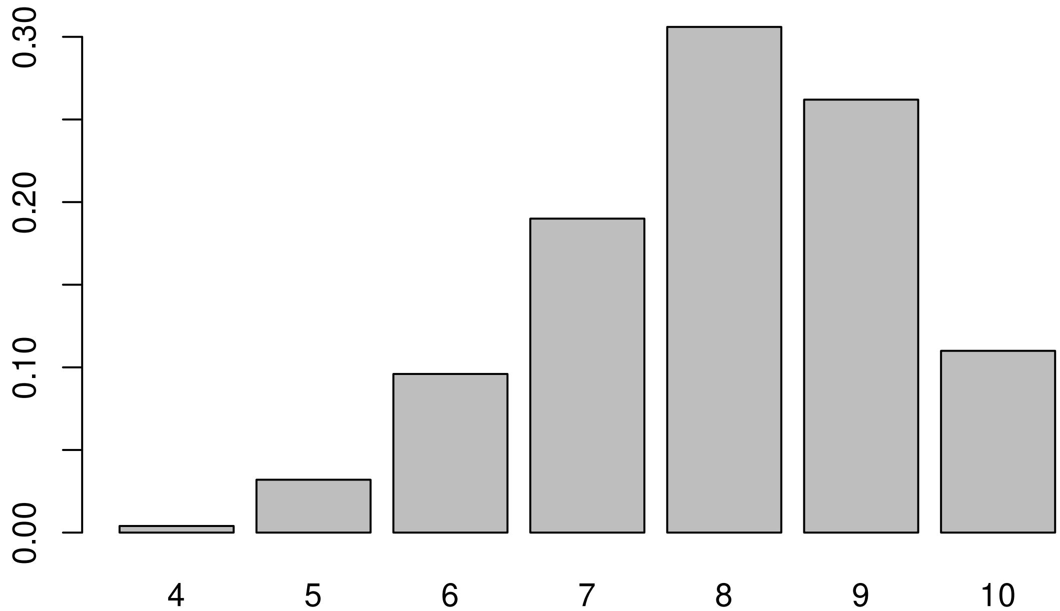
```r
plot(x, y, type = 'l', col = 'red'
      , xlim = c(lmin, lmax)
      , ylim = c(0, y_max*1.01)
      , xlab = ""
      , ylab = "")
```



Discrete case

data4.txt

```r
## Loading the data
A <- as.matrix(read.table("/cloud/project/docs/data/data4.txt"))

barplot(table(A)/length(A))
```




```
## A binomial distribution can be approach by a Gaussian distribution.

## Methods of moments
## Don't forget to correct the estimation of np to be an integer.
## and then correct the ph value




m = mean(A)
s2 = var(A)*(499/500)

ph = 1-s2/m
nh = m/ph

nh
```

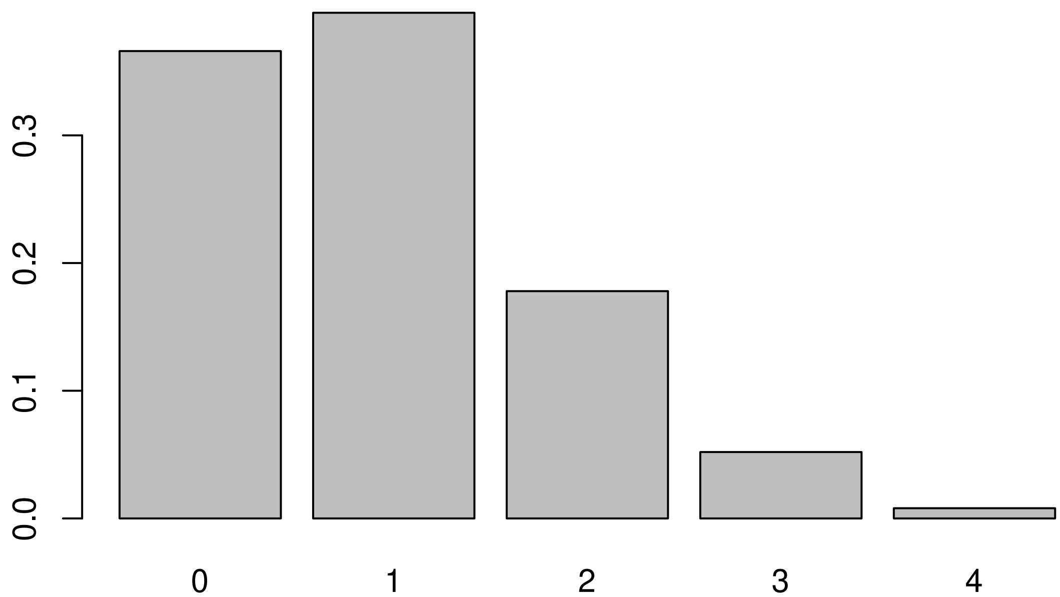
```
##          V1
## V1 10.0325
```

```
nh = 10

ph = m/nh

ph
```

```
## [1] 0.7988
```

data5.txt

```
## Loading the data
A <- as.matrix(read.table("/cloud/project/docs/data/data5.txt"))

barplot(table(A)/length(A))
```




```
## A binomial distribution can be approach by a Gaussian distribution.

## Methods of moments
## Don't forget to correct the estimation of np to be an integer.
## and then correct the ph value


m = mean(A)
s2 = var(A)*(499/500)

ph = 1-s2/m
nh = m/ph

nh
```

```
##          V1
## V1 7.38796
```

```
nh = 7

ph = m/nh

ph
```

```
## [1] 0.1342857
```

```
## When p is small is very hard to get correct estimations of n when you
## don't have enough number of observations
```