

Confidence intervals

Bachelor in Computer Science and Engineering

2020/21

1. Introduction

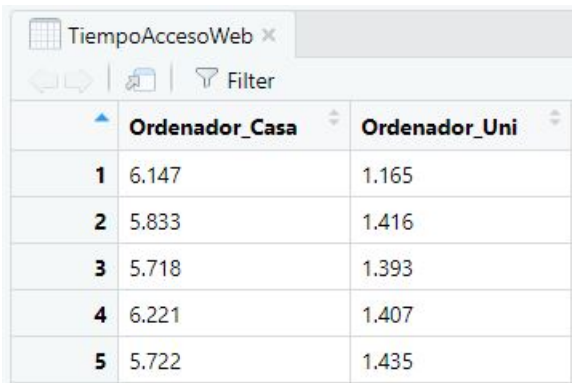
In R, most of the functions that return confidence intervals are functions related to the hypothesis tests that we have seen at the end of this topic. In this practice we will see how to calculate the confidence intervals using the formulas and also with the functions that implement hypothesis tests.

2 Example 1: Web-page accessing times

We want to construct the confidence intervals for the mean, μ , and for the variance, σ^2 , of the distribution of the accessing times to a web page of UC3M from a specific computer at home as well as from a computer in the UC3M campus. The confidence intervals will be constructed by using 55 observations (in seconds). Each observation consists of two accessing times, one measured on a home computer and one on a computer belonging to the university campus (file `TiempoAccesoWeb.xlsx`)

First we read and view the data file. The figure shows the first five observations of this datafile.

```
library(readxl)
TiempoAccesoWeb <- read_excel("TiempoAccesoWeb.xlsx")
```



	Ordenador_Casa	Ordenador_Uni
1	6.147	1.165
2	5.833	1.416
3	5.718	1.393
4	6.221	1.407
5	5.722	1.435

2.1. Univariate analysis of data

Before doing any kind of analysis it is useful to first describe the variables of interest. We start with the access times of the computer at home (variable `Ordenador_Casa`). The numerical and graphical analysis can be performed by

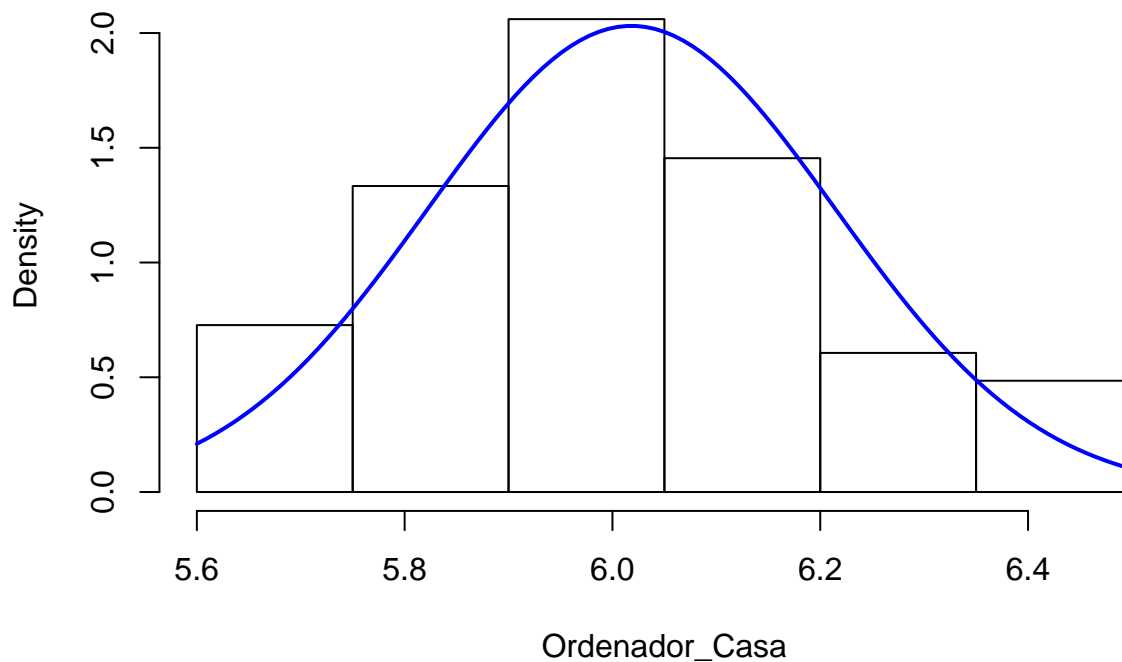
```
suppressWarnings(library(summarytools))
descr(TiempoAccesoWeb$Ordenador_Casa)
```

```
## Descriptive Statistics
## TiempoAccesoWeb$Ordenador_Casa
```

```
## N: 55
##
##
##      Ordenador_Casa
## -----
##      Mean          6.02
##      Std.Dev       0.20
##      Min           5.70
##      Q1            5.85
##      Median        6.02
##      Q3            6.14
##      Max           6.49
##      MAD           0.19
##      IQR           0.27
##      CV            0.03
##      Skewness       0.33
##      SE.Skewness    0.32
##      Kurtosis      -0.41
##      N.Valid       55.00
##      Pct.Valid     100.00
```

```
hist(TiempoAccesoWeb$Ordenador_Casa, breaks = seq(5.6, 6.5, .15),
     probability = TRUE, # histogram has a total area = 1
     xlab = "Ordenador_Casa")
curve(dnorm(x, mean(TiempoAccesoWeb$Ordenador_Casa), sd(TiempoAccesoWeb$Ordenador_Casa)),
     col="blue", lwd=2, add=TRUE, yaxt="n")
```

Histogram of TiempoAccesoWeb\$Ordenador_Casa



We can notice in the figure that the variable `Ordenador_Casa` has a Normal-liked distribution: it is quite

symmetric and bell-shaped. The hypothesis of normality is important to compute the confidence intervals. For example to construct a confidence interval for the variance it is *mandatory* to assume the normality since only in that case we know that the estimator is distributed as a Chi-squared distribution.

The summary statistics include measures of central tendency, measures of variability and measures of shape, we can notice that the values of the Skewness and the Kurtosis are quite small confirming the fact that the histogram looks like a Normal distribution.

Among these values, the sample mean and variance are the “point” estimations of the population mean and variance. That is, we have that in this sample, the “point” estimation of the parameters of interest are $\hat{\mu} = 6.02$ and $\hat{\sigma}^2 = 0.20^2$.

Our objective is to make an “interval” estimation of these parameters.

2.2 Analysis of the Normality of the variable

We perform a goodness-of-fit test to check if the variable can be assumed normal. For simplicity, we will use the normality tests provided by package `nortest`

```
library(nortest)
ad.test(TiempoAccesoWeb$Ordenador_Casa)

##
## Anderson-Darling normality test
##
## data: TiempoAccesoWeb$Ordenador_Casa
## A = 0.4999, p-value = 0.2005

cvm.test(TiempoAccesoWeb$Ordenador_Casa)

##
## Cramer-von Mises normality test
##
## data: TiempoAccesoWeb$Ordenador_Casa
## W = 0.06418, p-value = 0.3275

lillie.test(TiempoAccesoWeb$Ordenador_Casa)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: TiempoAccesoWeb$Ordenador_Casa
## D = 0.069611, p-value = 0.7283

pearson.test(TiempoAccesoWeb$Ordenador_Casa)

##
## Pearson chi-square normality test
##
## data: TiempoAccesoWeb$Ordenador_Casa
## P = 8.4545, p-value = 0.2942

sf.test(TiempoAccesoWeb$Ordenador_Casa)

##
## Shapiro-Francia normality test
##
## data: TiempoAccesoWeb$Ordenador_Casa
## W = 0.96939, p-value = 0.1515
```

As we see, the histogram with superimposed curve shows a good fit and all p-values are bigger than 0.05 which is the usual level of significance (5%), therefore we cannot reject the hypothesis that the variable is normally distributed.

Assuming that the variable is normal we can proceed by computing the confidence intervals for mean and the variance of the variable `Ordenador_Casa`. If we weren't able to assume normality we couldn't proceed to compute both confidence intervals. If the sample size were large enough (>30) we could still use the confidence interval for the mean (why?). In case of small samples and absence of normality, it would be not valid.

2.3 Confidence intervals

To obtain the confidence intervals for the mean, μ , and the variance, σ^2 , we evaluate the following expressions:

- Confidence interval for μ with known variance:

$$\bar{x} \pm z_{\alpha/2} \frac{\sigma}{\sqrt{n}},$$

where n is the sample size, σ is the known standard deviation, $z_{\alpha/2}$ is the $(\alpha/2)$ -percentil of the standard normal distribution (`qnorm(alpha/2)` in R) and \bar{x} is the sample mean (`mean(x)` in R).

In the example

```
alpha = 0.05
n = length(TiempoAccesoWeb$Ordenador_Casa)
xmean = mean(TiempoAccesoWeb$Ordenador_Casa)
xsd = sd(TiempoAccesoWeb$Ordenador_Casa)
z = qnorm(alpha/2, lower.tail = FALSE)
LowerLimit = xmean - z * xsd / sqrt(n)
UpperLimit = xmean + z * xsd / sqrt(n)
LowerLimit
```

```
## [1] 5.966438
```

```
UpperLimit
```

```
## [1] 6.070253
```

where we assume that σ is known and equals to `sd(TiempoAccesoWeb$Ordenador_Casa)`.

- Confidence interval for μ with unknown variance:

$$\bar{x} \pm t_{n-1, \alpha/2} \frac{s}{\sqrt{n}},$$

where n is the sample size, s is the sample standard deviation (`sd(x)` in R), \bar{x} is the sample mean (`mean(x)` in R) and $t_{n-1, \alpha/2}$ is the $(\alpha/2)$ -percentil of the Student's t distribution with $gl = n - 1$ (`qt(alpha/2, gl = n-1)` in R).

In the example

```
alpha = 0.05
n = length(TiempoAccesoWeb$Ordenador_Casa)
xmean = mean(TiempoAccesoWeb$Ordenador_Casa)
xsd = sd(TiempoAccesoWeb$Ordenador_Casa)
t = qt(alpha/2, df = n-1, lower.tail = FALSE)
LowerLimit = xmean - t * xsd / sqrt(n)
UpperLimit = xmean + t * xsd / sqrt(n)
LowerLimit
```

```
## [1] 5.965249
```

```
UpperLimit
```

```
## [1] 6.071442
```

which does not differ much from the previous one because $z = 1.959964$ and $t = 2.004879$. Therefore the mean access time to the webpage from a computer at home is between 5.96 and 6.07 seconds with a confidence level of 95%.

- Confidence interval for σ^2 :

$$\frac{(n-1)s^2}{\chi_{n-1,1-\alpha/2}^2}, \frac{(n-1)s^2}{\chi_{n-1,\alpha/2}^2},$$

where s^2 is the sample variance (`var(x)` in R), and $\chi_{n-1,1-\alpha/2}^2$ and $\chi_{n-1,\alpha/2}^2$ are the $(1-\alpha/2)$ - and $(\alpha/2)$ -percentil of the χ^2 distribution with $gl = n-1$ (`qchisq(1-alpha/2, gl = n-1)` and `qchisq(alpha/2, gl = n-1)` in R), respectively.

In the example

```
alpha = 0.05
n = length(TiempoAccesoWeb$Ordenador_Casa)
s2 = var(TiempoAccesoWeb$Ordenador_Casa)
chi.lower = qchisq(1-alpha/2, df = n-1)
chi.upper = qchisq(alpha/2, df = n-1)
LowerLimit = (n-1) * s2 / chi.lower
UpperLimit = (n-1) * s2 / chi.upper
LowerLimit
```

```
## [1] 0.02734039
```

```
UpperLimit
```

```
## [1] 0.05853708
```

Therefore the variance of the access time to the webpage from a computer at home is between 0.027 and 0.059 seconds² with a confidence level of 95%.

The above confidence intervals can be obtained as output of some functions implementing hypothesis testing:

- Confidence interval for μ with known variance using `z.test`:

```
suppressWarnings(library(BSDA))
z.test(TiempoAccesoWeb$Ordenador_Casa, sigma.x = xsd)$conf.int
```

```
## [1] 5.966438 6.070253
```

```
## attr(,"conf.level")
```

```
## [1] 0.95
```

- Confidence interval for μ with unknown variance using `t.test`:

```
t.test(TiempoAccesoWeb$Ordenador_Casa)$conf.int
```

```
## [1] 5.965249 6.071442
```

```
## attr(,"conf.level")
```

```
## [1] 0.95
```

- Confidence interval for σ^2 using `varTest`:

```
suppressWarnings(library(EnvStats))
varTest(TiempoAccesoWeb$Ordenador_Casa)$conf.int
```

```
##           LCL           UCL
## 0.02734039 0.05853708
## attr("conf.level")
## [1] 0.95
```

3. Example 2: Loop execution time

We consider the file `TiempoBucle.xlsx` that contains the durations in seconds of the executions of a Matlab program under different conditions. For each set of conditions we repeated the measurements of the executions 200 times. We want to construct the confidence interval for the population mean and variance of the execution times under the conditions defined for the variable `Estado1`.

3.1 Univariate analysis of data

We analyze the execution times contained in the variable `Estado1`. First we read and view the data file. The figure shows the first five observations of this datafile.

TiempoBucle							
	Estado1	Estado2	Estado3	Estado4	Tiempos	Estados	Resumen
1	0.210	0.220	1.142	1.061	0.210	1	1-Matlab6.1 sin inicializar ventanas abiertas
2	0.180	0.181	1.051	1.052	0.180	1	2- como 1- pero inicializando variables
3	0.180	0.190	1.042	1.041	0.180	1	3- como 1 con Matlab5.3
4	0.181	0.190	1.041	1.052	0.181	1	4- como 2 con Matlab5.3
5	0.190	0.191	1.032	1.051	0.190	1	NA

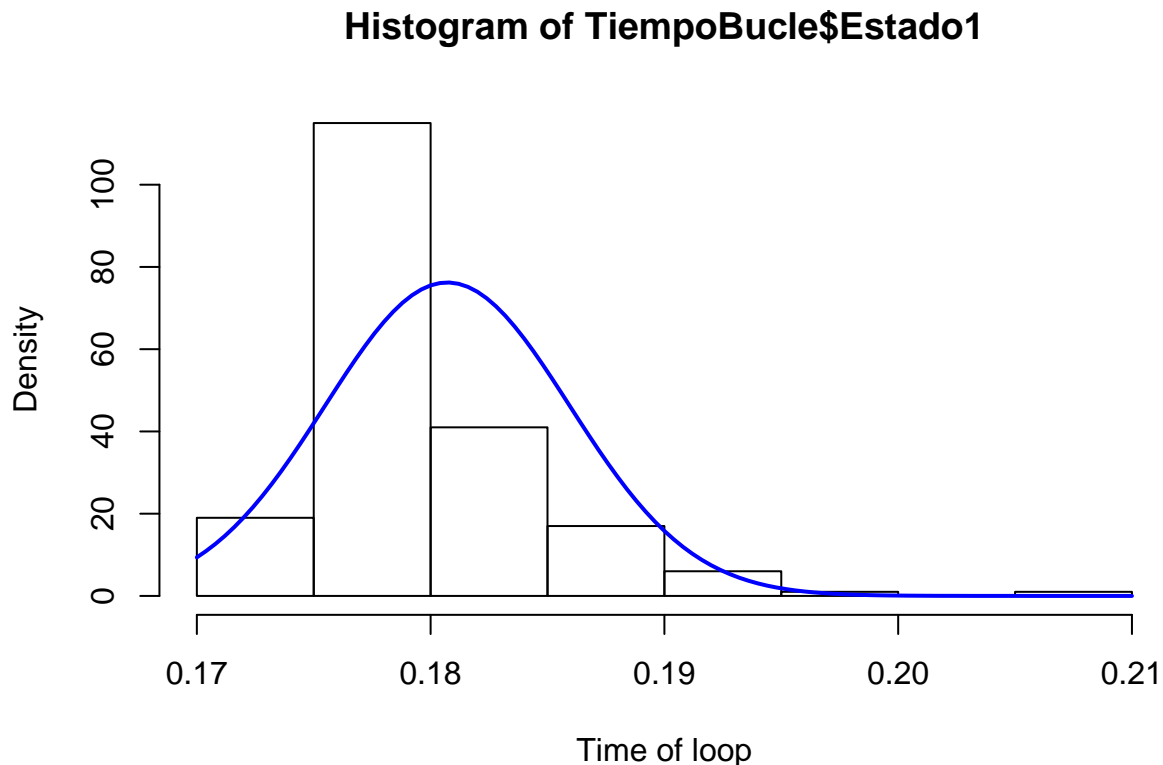
The graphical and numerical analysis is obtained by

```
suppressWarnings(library(summarytools))
descr(TiempoBucle$Estado1, na.rm = TRUE)
```

```
## Descriptive Statistics
## TiempoBucle$Estado1
## N: 800
##
##           Estado1
## -----
##           Mean    0.18
##          Std.Dev   0.01
##           Min     0.17
##           Q1      0.18
##          Median   0.18
##           Q3      0.18
##           Max     0.21
##           MAD     0.00
##           IQR     0.00
##           CV      0.03
##          Skewness  1.08
##         SE.Skewness 0.17
```

```
##          Kurtosis      5.61
##          N.Valid      200.00
##          Pct.Valid     25.00
```

```
hist(TiempoBucle$Estado1,
     probability = TRUE, # histogram has a total area = 1
     xlab = "Time of loop")
curve(dnorm(x, mean(TiempoBucle$Estado1, na.rm = T), sd(TiempoBucle$Estado1, na.rm = T)),
     col="blue", lwd=2, add=TRUE, yaxt="n")
```



We can observe that the distribution is concentrated around 0,18 seconds and it shows a strong positive symmetry. The variable looks more peaked than a bell, and therefore we conclude that it does not look Normal.

Among the statistics' values, the sample mean and variance are the “point” estimations of the population mean and variance. That is, we have that in this sample, the “point” estimation of the parameters of interest are $\hat{\mu} = 0.18$ and $\hat{\sigma}^2 = 0.01^2$.

Our objective is to make an “interval” estimation of these parameters.

3.2 Analysis of the Normality of the variable

To construct the confidence intervals we need to check if the variable of interest is normally distributed or not (why?). The histogram above did not look like a normal density function. In addition the values of Skewness and Kurtosis are high that denotes a positive asymmetry and a shape more peaked than a Normal one. The following lines performs some normality goodness-of-fit tests:

```

library(nortest)
ad.test(TiempoBucle$Estado1)

##
## Anderson-Darling normality test
##
## data: TiempoBucle$Estado1
## A = 29.356, p-value < 2.2e-16

cvm.test(TiempoBucle$Estado1)

## Warning in cvm.test(TiempoBucle$Estado1): p-value is smaller than 7.37e-10,
## cannot be computed more accurately

##
## Cramer-von Mises normality test
##
## data: TiempoBucle$Estado1
## W = 6.4149, p-value = 7.37e-10

lillie.test(TiempoBucle$Estado1)

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: TiempoBucle$Estado1
## D = 0.35329, p-value < 2.2e-16

pearson.test(TiempoBucle$Estado1)

##
## Pearson chi-square normality test
##
## data: TiempoBucle$Estado1
## P = 1150.8, p-value < 2.2e-16

sf.test(TiempoBucle$Estado1)

##
## Shapiro-Francia normality test
##
## data: TiempoBucle$Estado1
## W = 0.68502, p-value < 2.2e-16

```

The picture above shows that the fitting is indeed low precise. In addition from previous tests we see that the p-values are practically equal to zero, therefore we have to reject our null hypothesis of normality for the variable `Estado1`.

Since the sample size is large by the Central Limit Theorem we can assume that still the estimator for the population mean (the sample mean) has a normal distribution. Therefore even if the variable `Estado1` is not normal we can still construct a confidence interval for the mean, **but we CANNOT construct a confidence interval for its standard deviation or its variance.**

3.3 Confidence Intervals

To get the confidence interval for the mean, μ , we will use the output of functions `z.test` and `t.test`

```
library(BSDA)
z.test(TiempoBucle$Estado1, sigma.x = sd(TiempoBucle$Estado1, na.rm = T))$conf.int
```

```
## [1] NA NA
## attr(,"conf.level")
## [1] 0.95
```

```
t.test(TiempoBucle$Estado1)$conf.int
```

```
## [1] 0.1799852 0.1814448
## attr(,"conf.level")
## [1] 0.95
```

It should be noted that `z.test` does not work when there are NA. So, we should remove the NA using the function `na.exclude`

```
library(BSDA)
z.test(na.exclude(TiempoBucle$Estado1), sigma.x = sd(TiempoBucle$Estado1, na.rm = T))$conf.int
```

```
## [1] 0.1799897 0.1814403
## attr(,"conf.level")
## [1] 0.95
```

```
t.test(TiempoBucle$Estado1)$conf.int
```

```
## [1] 0.1799852 0.1814448
## attr(,"conf.level")
## [1] 0.95
```

Both intervals are very similar since the sample size is 200, then $z_\alpha \approx t_{n-1, \alpha}$.

Therefore the mean execution time of the Matlab program under the conditions used for the variable `Estado1` is a value contained in the interval [0.1799897, 0.1814403] seconds with a confidence level of 95%. Since the distribution of the random variable is not Normal, the confidence interval for the variance is not reliable and therefore we did not calculate it.