LABORATORIO 8 - Stima puntuale e stima intervallare

STATISTICA E LABORATORIO (CDL in INTERNET OF THINGS, BIG DATA, MACHINE LEARNING)

Anno Accademico 2022-2023

Controllo di qualita'

Con riferimento al problema di controllo della qualità, si sono individuati 3 oggetti non conformi agli standard in un campione casuale semplice di $n = 40$ oggetti. L'obiettivo è stimare la proporzione p di oggetti difettosi prodotti dal macchinario.

```r
x <- c(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
       1,0,1,0,0,0,0,0,0,0,0,0)
p <- mean(x)
p
```

```
## [1] 0.075
```

```r
se <- sqrt(p*(1-p)/length(x))
se
```

```
## [1] 0.04164583
```

Web

Si considerano i dati relativi ad osservazioni ripetute del numero di visite ad un sito web in un'ora. I dati campionari sono analizzati come realizzazione di un campione casuale semplice $X_1, \dots, X_{12}$, costituito da variabili casuali con distribuzione $P(\lambda)$, con $\lambda > 0$ ignoto.

```
x <- c(0,0,3,0,1,0,0,2,1,0,0,2)

lambda <- mean(x)

lambda
```

```
## [1] 0.75
```

```
se <- sqrt(lambda/length(x))
se
```

```
## [1] 0.25
```

Campione gaussiano

Sia $X_1, \dots, X_5$ un campione casuale semplice da una popolazione $N(\mu; 16)$, con μ ignoto. Si sono simulati 100 campioni di dimensione $n = 5$ da un modello $N(175, 16)$.

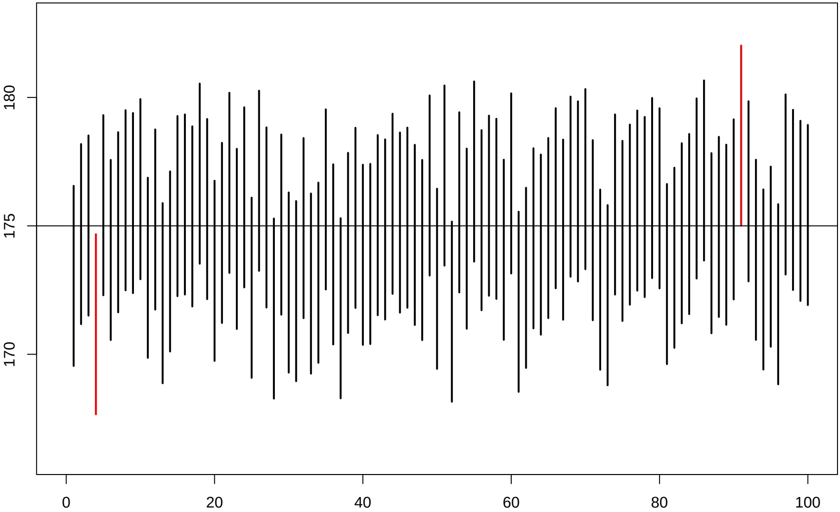
```
set.seed(1)
flag <- 0
# contatore intervalli che contengono il vero valore del parametro

y <- rnorm(5,175,sqrt(16))
# campione simulato da un modello N(175,16)

ci <- c(mean(y)-qnorm(0.975)*sqrt(16/5),
        mean(y)+qnorm(0.975)*sqrt(16/5))
# intervallo di confidenza per la media di livello 0.95
```

```r
if(c((ci[1]-175)*(ci[2]-175)<0))
  # condizione che 175 appartenga all'intervallo
  {plot(c(1,1),ci,ylim=c(166,183),xlim=c(0,100),type='l',
        xlab=' ',ylab=' ',lwd=2) # intervallo nero (contiene 175)
  flag <- flag+1} else
  plot(c(1,1),ci,ylim=c(166,183),xlim=c(0,100),type='l',
       xlab=' ',ylab=' ',col='red',lwd=2)
# intervallo rosso (non contiene 175)

# si ripete per altre 99 volte
for (i in 2:100){
  y <- rnorm(5,175,sqrt(16))
  ci <- c(mean(y)-qnorm(0.975)*sqrt(16/5),
          mean(y)+qnorm(0.975)*sqrt(16/5))
  if(c((ci[1]-175)*(ci[2]-175)<0))
  {lines(c(i,i),ci,lwd=2)
    flag <- flag+1}else
      lines(c(i,i),ci,col='red',lwd=2)
}
abline(h=175)
```


180
175
170
0
20
40
60
80
100

Campione gaussiano con varianza nota

Sia $X_1, \ldots, X_{50}$ un campione casuale semplice di dimensione n $= 50$ da una popolazione normale con μ ignoto e $\sigma^2 = 2$.

```
x<-c(5.00752201898176,1.85069810556901,-0.142274409463075,2.4590016
    2.06258684424398,2.37533005745348,3.09290733473573,2.891938275
    3.96777040974456,-0.223556500772768,1.24582432319052,-1.3859790
    1.72650871215828,2.19206395580825,4.91904449247961,3.521120170
    2.40192745906163,-1.10705146936821,1.68380070693403, 1.2993577
    -0.417950191674O2,2.15347847576679,1.96459316544775, 0.4205949
    1.32823769557452,-0.0314838172008567,3.92485950796051,2.005343
    3.42092165639571,0.678639779218378,2.14348357431764,0.13301217
    0.723328948100102,-0.253209399057751,1.79498218949182,
    3.01820992429422,2.06437797769392,1.44670540434426,
    3.79729782081437,2.27671844221721,1.34351162035059,
    0.726171895764826,3.35659896163966,2.98814208796288,
    2.34117306638397,2.31372635290214,2.24141654282489,
    1.61046063578177,1.44823507405404,3.34988000168233)
```

```r
mean(x) # media campionaria
```

```
## [1] 1.883
```

```r
sqrt(2/50) # standard error
```

```
## [1] 0.2
```

```r
# valore critico di livello 0.05 di una normale standard
qnorm(0.05,lower.tail=FALSE)
```

```
## [1] 1.644854
```

```r
# intervallo di confidenza per la media di livello 0.90
c(mean(x)-qnorm(0.05,lower.tail=FALSE)*sqrt(2/50),
  mean(x)+qnorm(0.05,lower.tail=FALSE)*sqrt(2/50))
```

```
## [1] 1.554029 2.211971
```

Campione gaussiano con varianza ignota

Sia $X_1, \dots, X_{30}$ un campione casuale semplice di dimensione $n = 30$ da una popolazione normale con μ e σ^2 ignoti.

```
x <- c(0.314060524647885,1.45971087975415,0.0182426832227007,
       3.45606774617669,1.66599435980862,0.0396824836819647,
       1.88932877663903,2.24414881142886,2.01427779646988,
       0.768115601092236,3.33798143176343,1.75132159213247,
       0.321433145501246,-1.93205861703258,2.7908925611708,
       1.13645428072429,1.17710351034708,2.53478596980998,
       2.36138215181596,2.0399033031771,2.49963026244232,
       2.3061077641182,1.30545081075318,-1.6133681486199,
       2.07656597898075,1.12062197531918,0.979671881460535,
       -0.879957968202915,0.523793707215946,1.79105862271379)
```

```r
mean(x)
```

```
## [1] 1.316613
```

```r
var(x)
```

```
## [1] 1.707999
```

```r
# valore critico di livello 0.025 di una t(29)
qt(0.025,29,lower.tail=FALSE)
```

```
## [1] 2.04523
```

```r
# intervallo di confidenza per la media di livello 0.95
c(mean(x)-qt(0.025,29,lower.tail=FALSE)*sqrt(var(x)/30),
  mean(x)+qt(0.025,29,lower.tail=FALSE)*sqrt(var(x)/30))
```

```
## [1] 0.8286074 1.8046195
```

```r
# in alternativa t.test(x,conf.level = 0.95)$conf.int
```

Farmaco

Si vuole studiare l'efficacia di un farmaco per curare una determinata patologia. Si effettua una sperimentazione su 550 pazienti e si riscontra che il farmaco è efficace in 393 casi.

```
somma <- 393
n <- 550
p <- somma/n
p
```

```
## [1] 0.7145455
```

```
se <- sqrt(p*(1-p)/n)
se
```

```
## [1] 0.0192576
```

```r
qnorm(0.025,lower.tail=FALSE)
```

```
## [1] 1.959964
```

```r
c(p-qnorm(0.025,lower.tail=FALSE)*se,
  p+qnorm(0.025,lower.tail=FALSE)*se)
```

```
## [1] 0.6768013 0.7522896
```

Sito web

Si vuole studiare il numero di accessi all'ora ad un sito web per il commercio elettronico. Si hanno i dati sugli accessi nelle ultime $n = 96$ ore, che risultano essere complessivamente 383.

```
x <- c(6,3,4,3,5,3,4,5,1,6,6,3,3,4,7,3,6,3,5,7,4,5,3,3,5,2,5,2,3,2,
     3,1,5,6,5,6,3,3,6,4,5,3,3,10,5,2,2,4,7,4,8,4,3,3,2,0,5,2,4,
     5,9,4,4,2,5,4,4,1,2,4,4,1,1,5,7,4,4,4,9,4,5,4,3,2,5,4,2,4,2,
     6,4,4,3,4,3,2)
sum(x)
```

```
## [1] 383
```

```
lambda <- mean(x)
lambda
```

```
## [1] 3.989583
```

```r
se <- sqrt(lambda/length(x))
se
```

```
## [1] 0.2038582
```

```r
qnorm(0.05,lower.tail=FALSE)
```

```
## [1] 1.644854
```

```r
c(lambda-qnorm(0.05,lower.tail=FALSE)*se,
  lambda+qnorm(0.05,lower.tail=FALSE)*se)
```

```
## [1] 3.654266 4.324900
```

Campione gaussiano con varianza ignota

Sia $X_1, \ldots, X_{30}$ un campione casuale semplice di dimensione $n = 30$ da una popolazione normale con μ e σ^2 ignoti. Si vuole determinare un intervallo di confidenza per σ^2 con livello $1 - \alpha = 0.95$.

```r
x <- c(0.314060524647885,1.45971087975415,0.0182426832227007,
     3.45606774617669,1.66599435980862,0.0396824836819647,
     1.88932877663903,2.24414881142886,2.01427779646988,
     0.768115601092236,3.33798143176343,1.75132159213247,
     0.321433145501246,-1.93205861703258,2.7908925611708,
     1.13645428072429,1.17710351034708,2.53478596980998,
     2.36138215181596,2.0399033031771,2.49963026244232,
     2.3061077641182,1.30545081075318,-1.6133681486199,
     2.07656597898075,1.12062197531918,0.979671881460535,
     -0.879957968202915,0.523793707215946,1.79105862271379)
mean(x)
```

```
## [1] 1.316613
```

```
var(x)
```

```
## [1] 1.707999
```

```
# valore critico di livello 0.975 di una chi^2(29)
qchisq(0.975,29,lower.tail=FALSE)
```

```
## [1] 16.04707
```

```
# valore critico di livello 0.025 di una chi^2(29)
qchisq(0.025,29,lower.tail=FALSE)
```

```
## [1] 45.72229
```

```
# intervallo di confidenza per la varianza di livello 0.95
c(29*var(x)/qchisq(0.025,29,lower.tail=FALSE),
  29*var(x)/qchisq(0.975,29,lower.tail=FALSE))
```

```
## [1] 1.083322 3.086667
```

```
# in alternativa
# S2 <- 29*var(x)/30
# c(30*S2/qchisq(0.025,29,lower.tail=FALSE),
# 30*S2/qchisq(0.975,29,lower.tail=FALSE))
```

Esercizio 1 Cap.7 (Ieva et al.,2016)

Si consideri il file “glucosio.txt” (dati da: Bland M., An Introduction to Medical Statistics, Oxford University Press, 1995). I dati consistono nella misura della concentrazione di glucosio nel sangue [mmol/l] in un gruppo di 40 studenti di medicina. Si costruisca un intervallo di confidenza di livello 99% per la vera media della concentrazione di glucosio nel sangue.

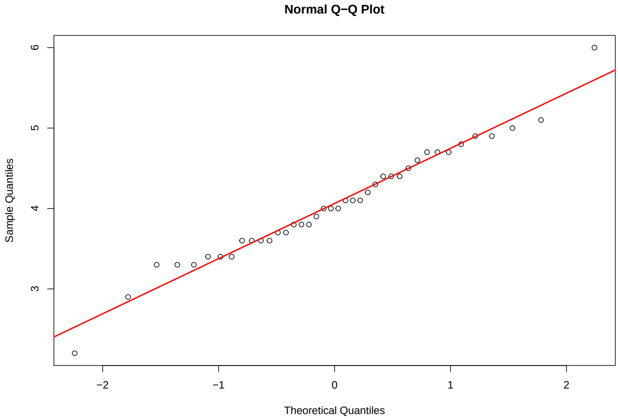
```
setwd("C:/Users/mamel/Desktop/Didattica 2022-2023/LABORATORIO DI STA
gluc = read.table(file="glucosio.txt", header=T)
head(gluc)
```

```
##   Glucosio
## 1      4.7
## 2      3.6
## 3      3.8
## 4      2.2
## 5      4.7
## 6      4.1
```

```
attach(gluc)
```

Verifica della normalità dei dati

```
qqnorm(Glucosio)
qqline(Glucosio,col='red',lwd=2)
```



```r
alpha= 0.01
n = length(Glucosio)
n
```

```
## [1] 40
```

```r
t.alpha = qt(1-alpha/2, n-1)
med = mean(Glucosio)
stdev = sd(Glucosio)

IC.alpha = c(med - t.alpha*stdev/(sqrt(n)),
             med + t.alpha*stdev/(sqrt(n)))
IC.alpha
```

```
## [1] 3.756003 4.353997
```

Esercizio 1 Cap.6 (Ieva et al.,2016)

Si consideri il file “magnesio.txt” (dati da: Bland, M., An introduction to Medical Statistics, Oxford University Press, 1995). I dati rappresentano la concentrazione di magnesio nel plasma (mmol/l) in 140 soggetti apparentemente sani.

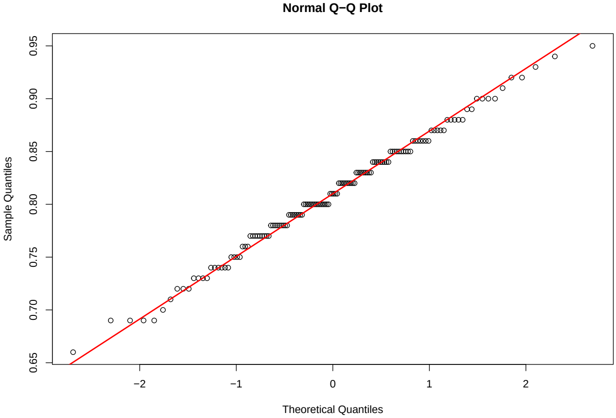
a) Verificare se l’assunzione di normalità dei dati è soddisfatta.
b) Costruire un intervallo di confidenza di livello 95% per la media della concentrazione di magnesio nel plasma.

```
magnesio <- read.csv("magnesio.txt", sep="")
head(magnesio)
```

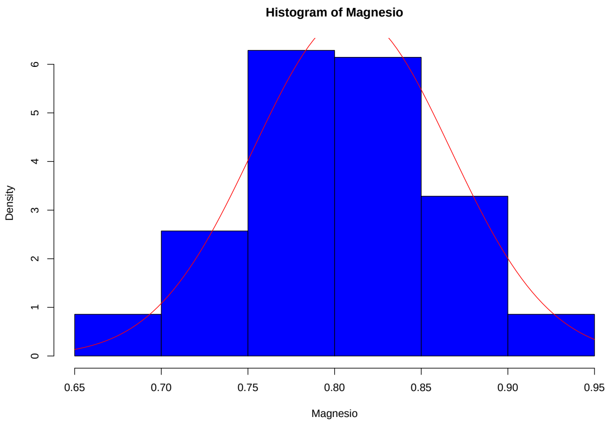
```
##   Magnesio
## 1     0.66
## 2     0.69
## 3     0.69
## 4     0.69
## 5     0.69
## 6     0.70
```

Verifica della normalità dei dati

```
attach(magnesio)
qqnorm(Magnesio)
qqline(Magnesio,col='red',lwd=2)
```



```
hist(Magnesio,prob=TRUE,col='blue')
curve(dnorm(x, mean=mean(Magnesio),sd=sd(Magnesio)),col='red',add=T)
```



```r
alpha <- 0.05
n <- length(Magnesio)
n
```

```
## [1] 140
```

```r
media.camp <- mean(Magnesio)
devstd.camp <- sd(Magnesio)
t.alpha <- qt(1-alpha/2,n-1)
IC.alpha_t <- c(media.camp - t.alpha*devstd.camp/sqrt(n),
              media.camp + t.alpha*devstd.camp/sqrt(n))
IC.alpha_t
```

```
## [1] 0.8004964 0.8195036
```

```r
z.alpha <- qnorm(1-alpha/2)
IC.alpha_z <- c(media.camp - z.alpha*devstd.camp/sqrt(n),
              media.camp + z.alpha*devstd.camp/sqrt(n))
IC.alpha_z
```

```
## [1] 0.8005792 0.8194208
```