LABORATORIO 2 - Statistica Descrittiva Univariata

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

Anno Accademico 2023-2024

Section 1

Variabili statistiche

Esempio di matrice dei dati (data frame)

```
## Nota: e' necessario installare e caricare i
## pacchetti "DescTools" e "moments" ##
# vettore di caratteri
genere <- c("M","M","F","M","F","M","M","F")

# vettore numerico
eta <- c(28,17,20,32,16,34,18,25)

# fattore ordinato
livistr <- factor(c(3,2,3,4,2,4,3,3),
                  levels = c("1", "2", "3", "4"), ordered = TRUE)
dist <- c(5.0,7.5,12.0,3.2,NA,12.3,25.0,7.7) # vettore numerico

# creazione data frame, "genere" viene interpretato come factor
matrdati <- data.frame(genere,eta,livistr,dist)
```

```
str(matrdati)
```

```
## 'data.frame':    8 obs. of  4 variables:
##  $ genere : chr  "M" "M" "F" "M" ...
##  $ eta    : num  28 17 20 32 16 34 18 25
##  $ livistr: Ord.factor w/ 4 levels "1"<"2"<"3"<"4": 3 2 3 4 2 4 3
##  $ dist   : num  5 7.5 12 3.2 NA 12.3 25 7.7
```

```
matrdati
```

```
##   genere eta livistr dist
## 1      M  28       3  5.0
## 2      M  17       2  7.5
## 3      F  20       3 12.0
## 4      M  32       4  3.2
## 5      F  16       2   NA
## 6      M  34       4 12.3
## 7      M  18       3 25.0
## 8      F  25       3  7.7
```

Distribuzioni di frequenza

```r
#  Esempio di matrice dei dati
# frequenze assolute
table(matrdati$genere)
```

```
##
## F M
## 3 5
```

```r
# frequenze relative
table(matrdati$genere)/sum(table(matrdati$genere))
```

```
##
##     F     M
## 0.375 0.625
```

```r
table(matrdati$livistr)
```

```
##
## 1 2 3 4
## 0 2 4 2
```

```r
table(matrdati$livistr)/sum(table(matrdati$livistr))
```

```
##
##    1    2    3    4
## 0.00 0.25 0.50 0.25
```

```r
# frequenze assolute cumulate
cumsum(table(matrdati$livistr))
```

```
## 1 2 3 4
## 0 2 6 8
```

```r
# frequenze relative cumulate
cumsum(table(matrdati$livistr))/sum(table(matrdati$livistr))
```

```
##    1    2    3    4
## 0.00 0.25 0.75 1.00
```

```r
# si usa cut per definire le classi 0-5 5-15 15-
freq_ass = table(cut(matrdati$dist,c(0,5,15,Inf)))
freq_ass
```

```
##
##    (0,5]   (5,15] (15,Inf]
##        2        4        1
```

```r
freq_rel = freq_ass/sum(freq_ass)
freq_rel
```

```
##
##     (0,5]    (5,15]  (15,Inf]
## 0.2857143 0.5714286 0.1428571
```

```
freq_ass_cum = cumsum(freq_ass)
freq_ass_cum
```

```
##    (0,5]   (5,15] (15,Inf]
##        2        6        7
```

```
freq_rel_cum=freq_ass_cum/sum(freq_ass)
freq_rel_cum
```

```
##     (0,5]    (5,15]  (15,Inf]
## 0.2857143 0.8571429 1.0000000
```

Perni

In uno stabilimento industriale ci sono tre macchinari per la produzione di perni di acciaio, che devono rispettare le specifiche di diametro. Per valutarne l'efficacia del procedimento produttivo si analizzano $n = 400$ perni che vengono classificati, con riferimento agli standard richiesti per il diametro, in:

- fine: il diametro è troppo fine rispetto alle specifiche richieste;
- ok: il diametro soddisfa le specifiche richieste;
- spesso: il diametro è troppo spesso rispetto alle specifiche richieste.

```r
## matrice con i dati grezzi
perni <- rbind(cbind(rep("M1",10), rep("Fine",10)),
               cbind(rep("M1",102),rep("Ok",102)),
               cbind(rep("M1",8), rep("Spesso",8)),
               cbind(rep("M2",34), rep("Fine",34)),
               cbind(rep("M2",161),rep("Ok",161)),
               cbind(rep("M2",5), rep("Spesso",5)),
               cbind(rep("M3",10), rep("Fine",10)),
               cbind(rep("M3",60), rep("Ok",60)),
               cbind(rep("M3",10), rep("Spesso",10)))

# matrice trasformata in data frame
perni <- as.data.frame(perni)

# nomi delle colonne
colnames(perni) <- c("Macchinario","Diametro")
```

```r
str(perni)
```

```
## 'data.frame':    400 obs. of  2 variables:
##  $ Macchinario: chr  "M1" "M1" "M1" "M1" ...
##  $ Diametro   : chr  "Fine" "Fine" "Fine" "Fine" ...
```

```r
# tabella frequenze assolute
table(perni$Macchinario, perni$Diametro)
```

```
##
##      Fine  Ok Spesso
##   M1   10 102      8
##   M2   34 161      5
##   M3   10  60     10
```

```r
# totali di riga
apply(table(perni$Macchinario, perni$Diametro),1,sum)
```

```
##  M1  M2  M3
## 120 200  80
```

```r
# in alternativa
# margin.table(table(perni$Macchinario, perni$Diametro),1)
```

```r
# totali di colonna
apply(table(perni$Macchinario, perni$Diametro),2,sum)
```

```
##    Fine      Ok Spesso
##      54     323      23
```

```r
# in alternativa
# margin.table(table(perni$Macchinario, perni$Diametro),2)
```

```r
# tabella frequenze di Diametro per Macchinario
freq.ass = table(perni$Macchinario, perni$Diametro)
freq.ass
```

```
##
##       Fine  Ok Spesso
##    M1   10 102      8
##    M2   34 161      5
##    M3   10  60     10
```

```r
freq.rel = freq.ass/apply(freq.ass,1,sum)
freq.rel
```

```
##
##             Fine          Ok      Spesso
##    M1 0.08333333 0.85000000 0.06666667
##    M2 0.17000000 0.80500000 0.02500000
##    M3 0.12500000 0.75000000 0.12500000
```

```r
# frequenze assolute Diametro
freq.ass.Diam = margin.table(freq.ass,2)
freq.ass.Diam
```

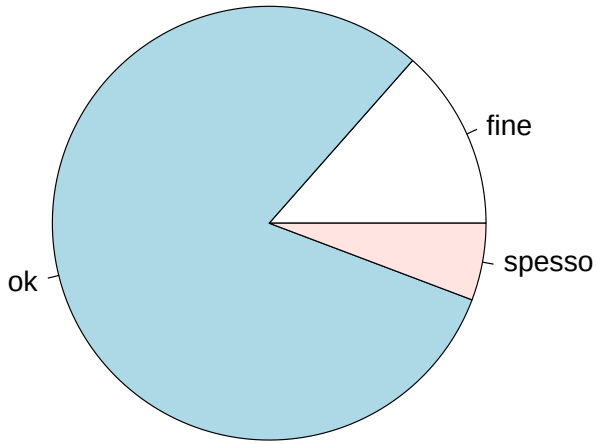
```
##
##   Fine     Ok Spesso
##     54    323     23
```

```r
# frequenze relative Diametro
freq.rel.Diam <-freq.ass.Diam/sum(freq.ass.Diam)
freq.rel.Diam
```

```
##
##   Fine     Ok Spesso
## 0.1350 0.8075 0.0575
```

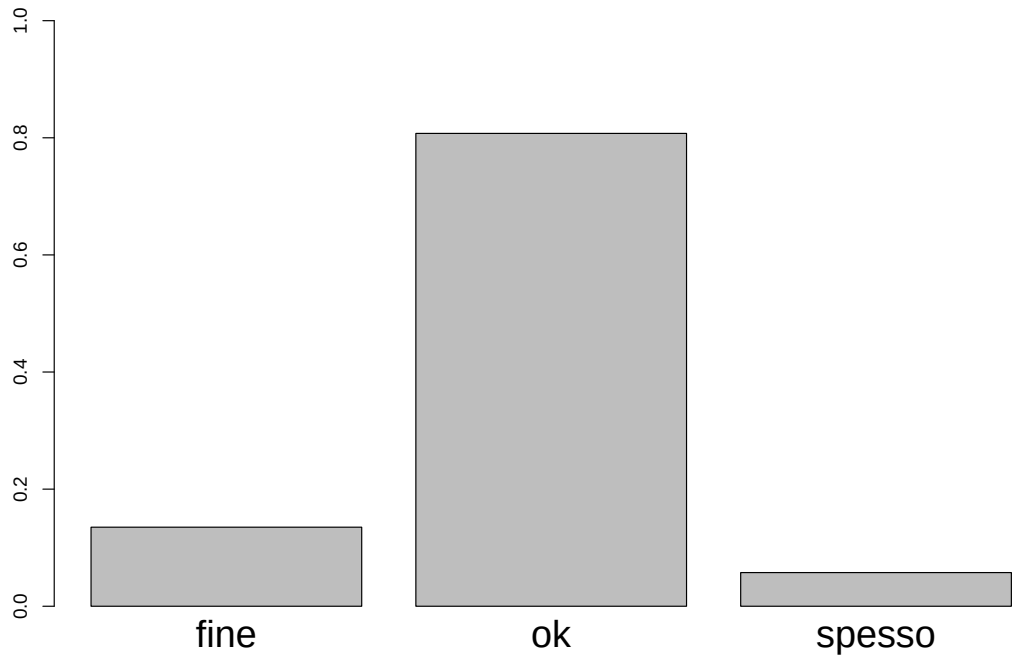
Rappresentazioni grafiche

```
# Diagramma circolare
pie(freq.ass.Diam,c("fine","ok","spesso"),cex=1.5) # freq. assolute
```



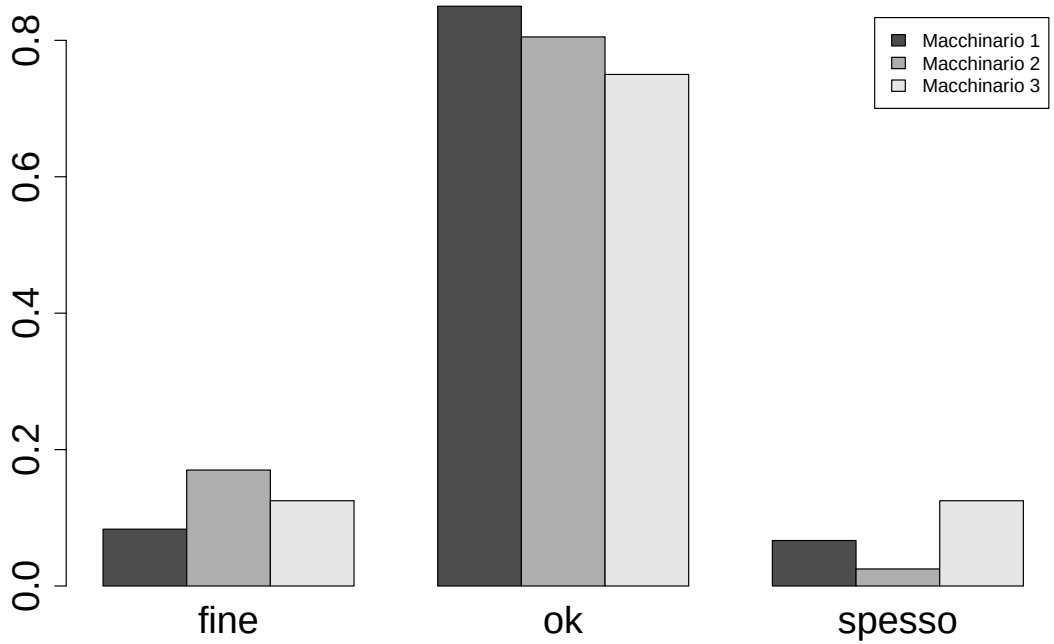

```
# stesso risultato con le freq. relative
#pie(freq.rel.Diam,c("fine","ok","spesso"),cex=1.5)
```

```
# Diagramma a barre
barplot(freq.rel.Diam,names.arg=c("fine","ok","spesso"),
        ylim=c(0,1),cex.names=2)
```



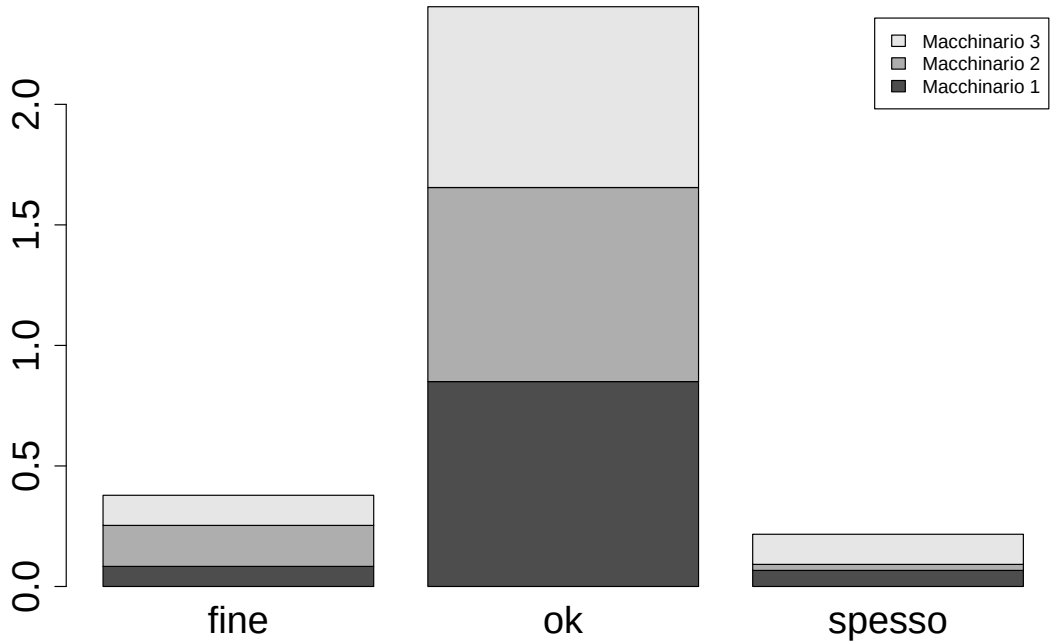

```
# stesso risultato con le freq. assolute,
# togliendo l'opzione ylim=c(0,1)
# barplot(freq.rel.Diam,names.arg=c("fine","ok","spesso"),
#cex.names=2)
```

```
# tabella frequenze relative di Diametro per Macchinario
freq.rel <- freq.ass/apply(freq.ass,1,sum)
barplot(freq.rel,beside=T,names.arg=c("fine","ok","spesso"),
legend.text=c("Macchinario 1","Macchinario 2","Macchinario 3"),
cex.axis=2,cex.names=2)
```




```
# in questo caso bisogna considerare le frequenze relative
```

```
# con l'opzione beside=F
barplot(freq.rel,beside=F,names.arg=c("fine","ok","spesso"),
legend.text=c("Macchinario 1","Macchinario 2","Macchinario 3"),
cex.axis=2,cex.names=2)
```



Figli

Si considera il numero di figli con riferimento alle famiglie residenti in un determinato territorio.

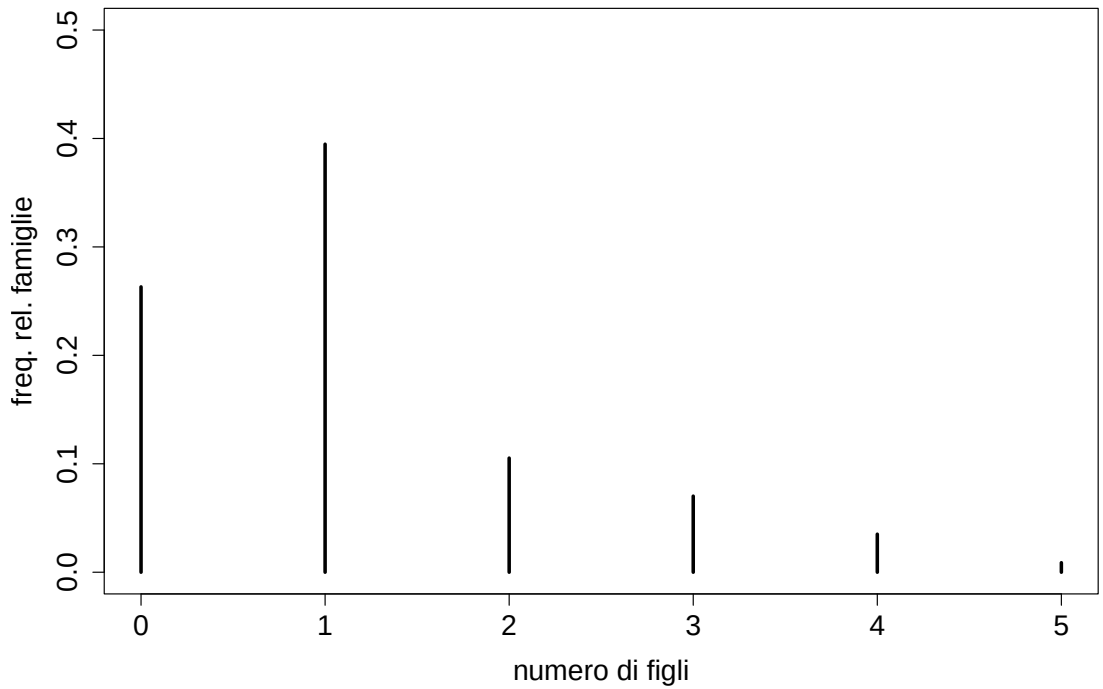
```
figli <-c(rep(0,30),rep(1,45),rep(2,12),rep(3,8),rep(4,4),rep(5,1))
table(figli)
```

```
## figli
##  0  1  2  3  4  5
## 30 45 12  8  4  1
```

```
table(figli)/sum(figli)
```

```
## figli
##           0           1           2           3           4           5
## 0.26315789 0.39473684 0.10526316 0.07017544 0.03508772 0.00877193
```

```
plot(table(figli)/sum(figli),lwd=3,ylim=c(0,0.5),
     xlab="numero di figli",ylab="freq. rel. famiglie",
     cex.lab=1.5,cex.axis=1.5)
```

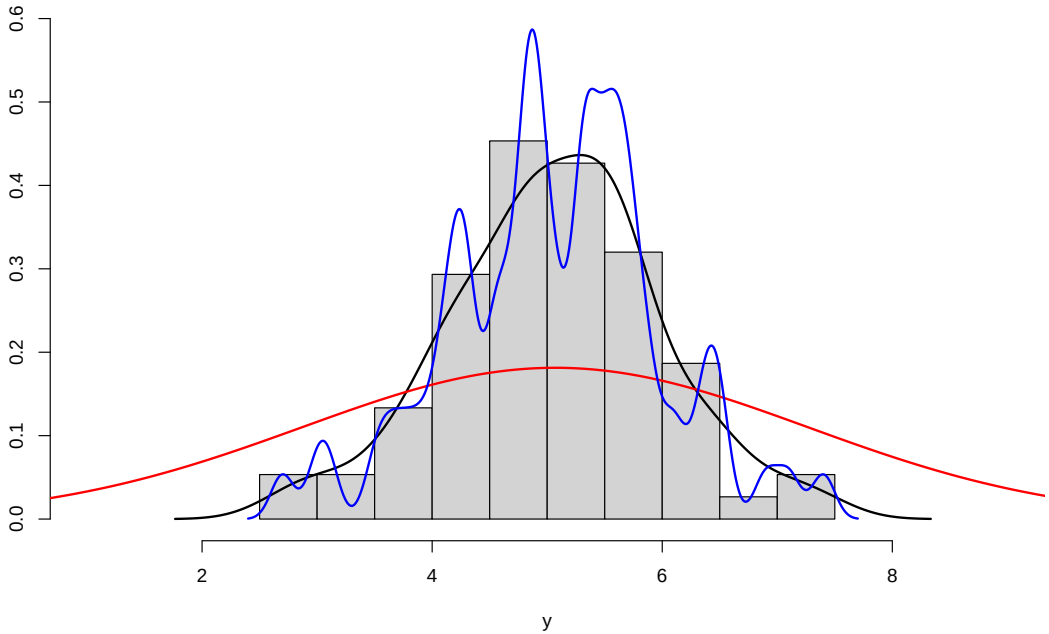



```
# risultato simile (su scala diversa) considerando le freq. assolute
# plot(table(figli),lwd=3,ylim=c(0,55),xlab="numero di figli",
#ylab="freq. rel. famiglie",cex.lab=1.5,cex.axis=1.5)
# il comando plot(figli) produce un risultato non desiderato
```

Istogramma e stima della densità

```
y <- c(4.3, 5.1, 4.1, 6.5, 5.3, 4.1, 5.4, 5.7, 5.5, 4.6, 6.5, 5.3,
       4.3, 2.7, 6.1, 4.9, 4.9, 5.9, 5.8, 5.5, 5.9, 5.7, 5.0, 3.0,
       5.6, 4.9, 4.8, 3.5, 4.5, 5.4, 6.3, 4.8, 5.3, 4.9, 3.6, 4.5,
       4.6, 4.9, 6.1, 5.7, 4.8, 4.7, 5.6, 5.5, 4.3, 4.2, 5.3, 5.7,
       4.8, 5.8, 5.3, 4.3, 5.3, 3.8, 6.4, 6.9, 4.6, 3.9, 5.5, 4.8,
       7.4, 4.9, 5.6, 5.0, 4.2, 5.1, 3.1, 6.4, 5.1, 7.1, 5.4, 4.2,
       5.6, 4.0, 3.7)
# istogramma delle frequenze relative (freq = FALSE)
hist(y,xlab="y",ylab=" ", xlim=c(1,9),ylim=c(0,0.65),main=" ",
     freq = FALSE)
# density() fornisce le coordinate della stima della densita'
# (scelta ottimale della banda)
lines(density(y),lwd=2)
# stima della densita' con banda troppo grande (bw=2)
lines(density(y,bw=2),lwd=2,col='red')
# stima della densita' con banda troppo piccola (bw=0.1)
lines(density(y,bw=0.1),lwd=2,col='blue')
```

Istogramma e stima della densità



Funzione di ripartizione empirica

```
y <- c(2, 2, 3, 5, 2, 5, 5, 4, 3, 1, 2, 2, 4, 3, 4, 3, 4, 7, 2, 4,
       5, 2, 4, 1, 2, 3, 0, 2, 5, 2, 3, 3, 3, 2, 4, 4, 4, 1, 4, 3,
       4, 3, 4, 3, 3, 4, 0, 3, 4, 4)
table(y) # frequenze assolute
```

```
## y
##  0  1  2  3  4  5  7
##  2  3 11 13 15  5  1
```

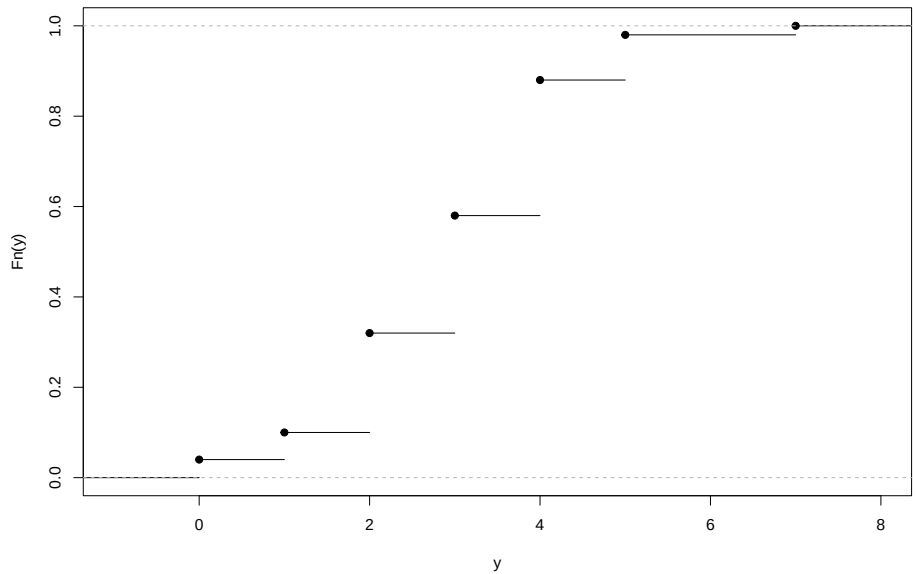
```
table(y)/length(y) # frequenze relative
```

```
## y
##    0    1    2    3    4    5    7
## 0.04 0.06 0.22 0.26 0.30 0.10 0.02
```

```
cumsum(table(y)/length(y)) # frequenze relative cumulate
```

```
##    0    1    2    3    4    5    7
## 0.04 0.10 0.32 0.58 0.88 0.98 1.00
```

```
plot(ecdf(y),main=" ",xlab="y", ylab="Fn(y)")
```




```
# ecdf() fornisce le coordinate della funzione di ripartizione
#empirica
```

Geyser Old Faithful

Si dispone di dati riferiti alle durate delle pause (in minuti) e alla tipologia delle eruzioni che precedono le pause (lunga o corta), con riferimento al geyser Old Faithful che si trova nel parco nazionale di Yellowstone, Wyoming, USA. Si hanno le seguenti n = 272 osservazioni

```
?faithful # data set disponibile in R
```

```
## avvio in corso del server httpd per la guida ... fatto
```

```
str(faithful)
```

```
## 'data.frame':    272 obs. of  2 variables:
##  $ eruptions: num  3.6 1.8 3.33 2.28 4.53 ...
##  $ waiting  : num  79 54 74 62 85 55 88 85 51 85 ...
```

```
# vettore di caratteri: "Corta",
# se l'eruzione e' minore di 3 minuti e "Lunga" altrimenti
```

```r
duration <- ifelse(faithful$eruptions < 3,"Corta", "Lunga")
duration <- factor(duration) # il vettore viene specificato
# come fattore con due livelli: "Corta" e "Lunga"

faithful1 <- data.frame(Pausa=faithful$waiting,Eruzione=duration)
# nuovo data frame con le variabili Pausa ed Eruzione
str(faithful1)
```

```
## 'data.frame':    272 obs. of  2 variables:
##  $ Pausa   : num  79 54 74 62 85 55 88 85 51 85 ...
##  $ Eruzione: Factor w/ 2 levels "Corta","Lunga": 2 1 2 1 2 1 2 2
```

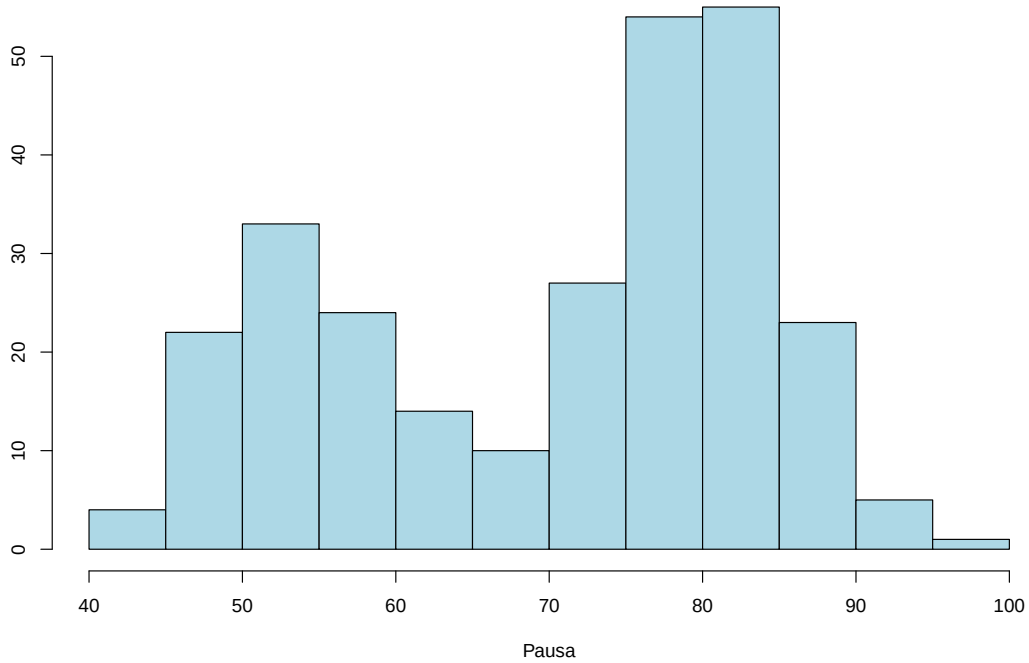
```r
head(faithful1) # le prime 6 righe del data frame
```

```
##   Pausa Eruzione
## 1    79    Lunga
## 2    54    Corta
## 3    74    Lunga
## 4    62    Corta
## 5    85    Lunga
## 6    55    Corta
```

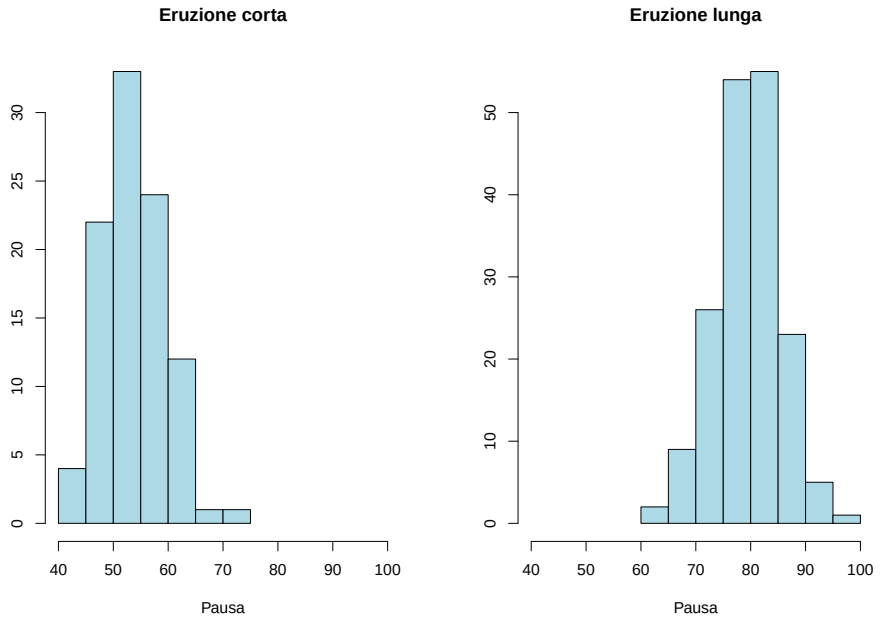
```r
faithful1[270:272,] # le ultime 3 righe del data frame
```

```
##     Pausa Eruzione
## 270    90    Lunga
## 271    46    Corta
## 272    74    Lunga
```

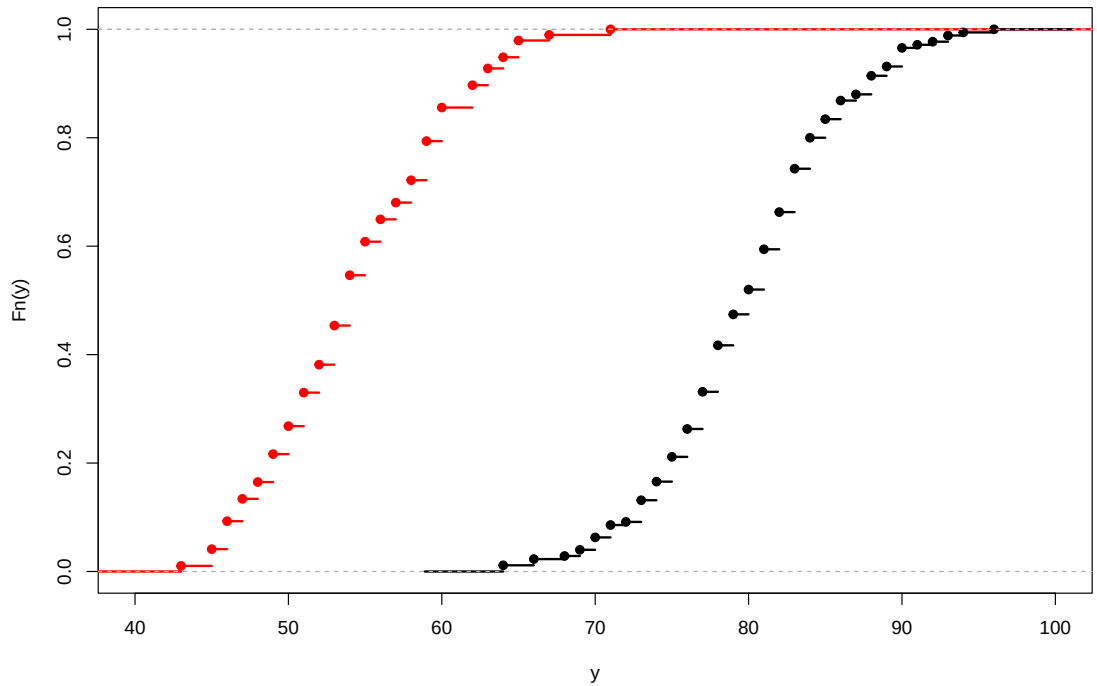
```
hist(faithful1$Pausa,xlab="Pausa",ylab=" ",col="lightblue",main=" ")
```



```
par(mfrow=c(1,2)) # finestra grafica con due pannelli su una riga
# istogramma di Pausa per Eruzione="Corta" e
# di Pausa per Eruzione="Lunga"
hist(faithful1$Pausa[faithful1$Eruzione=="Corta"],
     xlim=c(40,100),xlab="Pausa",ylab=" ",col="lightblue",
     main="Eruzione corta")
hist(faithful1$Pausa[faithful1$Eruzione=="Lunga"],
     xlim=c(40,100),xlab="Pausa",ylab=" ",col="lightblue",
     main="Eruzione lunga")
```



```
# Funzione di ripartizione empirica di Pausa per
# Eruzione="Corta" (rosso) e di Pausa per Eruzione="Lunga" (nero)
plot(ecdf(faithful1$Pausa[faithful1$Eruzione=="Corta"]),
     xlim=c(40,100),main=" ",xlab="y", ylab="Fn(y)",col="red",lwd=2)
plot(ecdf(faithful1$Pausa[faithful1$Eruzione=="Lunga"]),
     main=" ",xlab=" ", ylab=" ",add=TRUE,lwd=2)
```



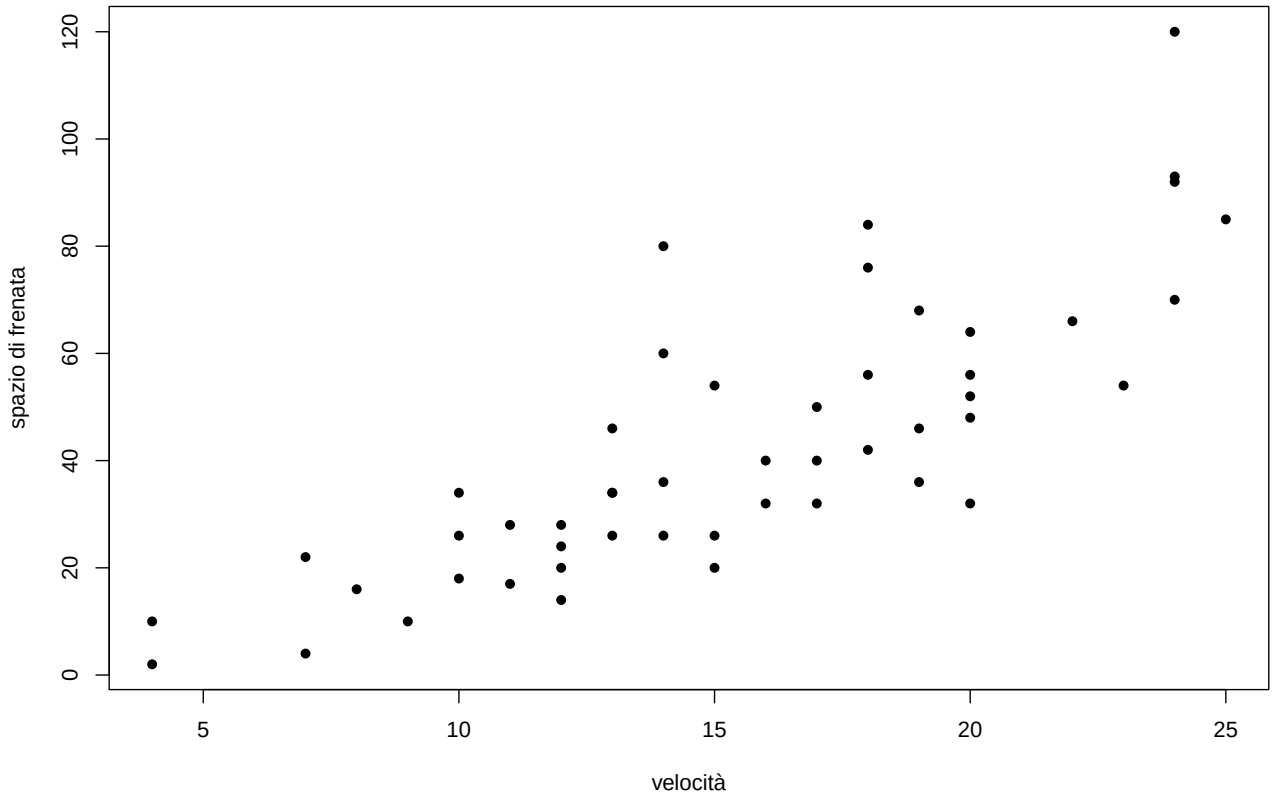
Velocità

Si dispone di dati riferiti alla velocità, in miglia orarie, e allo spazio di frenata, in piedi, per $n = 50$ automobili degli anni '20.

```
?cars # data set disponibile in R
str(cars)
```

```
## 'data.frame':    50 obs. of  2 variables:
##  $ speed: num  4 4 7 7 8 9 10 10 10 11 ...
##  $ dist : num  2 10 4 22 16 10 18 26 34 17 ...
```

```
plot(cars$speed,cars$dist,main=" ",
     xlab="velocità",ylab="spazio di frenata",pch=16)
```

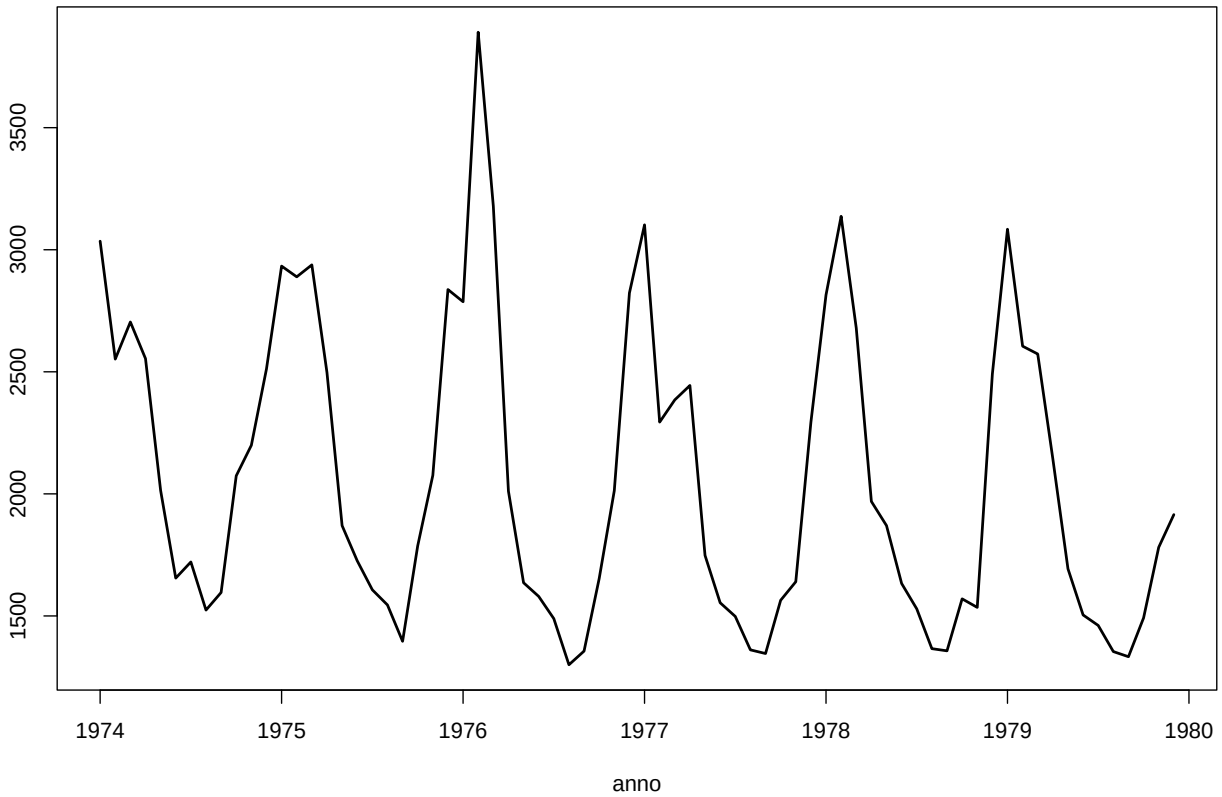


Patologie polmonari

Si considerano i dati riferiti al numero di decessi mensili per patologie polmonari (bronchiti, asma, enfisema) rilevati nel Regno Unito dal 1974 al 1979.

```
?UKLungDeaths  # data set disponibile in R
data(UKLungDeaths) # per caricare dataset
```

```r
# per rappresentare graficamente una serie temporale
ts.plot(ldeaths, xlab="anno", ylab=" ",lwd=2)
```



Section 2

Indici sintetici

Esempio 1 media

```
y <- c(27,30,30)
mean(y)
```

```
## [1] 29
```

```
y <- c(28,30,30)
mean(y)
```

```
## [1] 29.33333
```

Esempio 2 media

```
y <- c(rep(0,109),rep(1,65),rep(2,22),rep(3,3),rep(4,1))
mean(y) # calcolo della media con i dati grezzi
```

```
## [1] 0.61
```

```
table(y) # frequenze assolute
```

```
## y
##   0   1   2   3   4
## 109  65  22   3   1
```

```
# calcolo della media con la tabella di frequenza
sum(seq(0,4,1)*table(y))/sum(table(y))
```

```
## [1] 0.61
```

Esempio 3 media

```
yc <- c((0+10)/2,(10+15)/2,(15+20)/2)
p <- c(0.3,0.52,0.18)
sum(yc*p)
```

```
## [1] 11.15
```

Luogo di lavoro

Un lavoratore può raggiungere il luogo di lavoro in bicicletta o in automobile. Vorrebbe scegliere il mezzo di trasporto che gli consente il maggiore risparmio di tempo.

```
x <- c(23,32,44,21,36,30,28,33,45,34,29,31)
y <- c(22,24,22,33,26,31,24,28,32,31,37,24)

min(x)
```

```
## [1] 21
```

```
min(y)
```

```
## [1] 22
```

```
max(x)
```

```
## [1] 45
```

```
max(y)
```

```
## [1] 37
```

```
mean(x)
```

```
## [1] 32.16667
```

```
mean(y)
```

```
## [1] 27.83333
```

Polveri sottili

Si vuole studiare l'emissione di polveri sottili (PM), in grammi per 5 litri, per $n = 13$ veicoli a gasolio. I dati grezzi vengono riportati nella seguente tabella, dove si individuano anche i veicoli con un alto chilometraggio (A) e un basso chilometraggio (B)

```
km <- factor(c("B", "A", "A", "B", "B", "B", "A", "B", "B", "A",
               "A", "B", "B"))
pm <- c(2.30, 2.15, 3.50, 2.60, 2.75, 2.82, 4.05, 2.25, 2.68, 3.00,
        4.02, 2.85, 3.38)
mean(pm) # media tutti i veicoli
```

```
## [1] 2.95
```

```
mean(pm[km=="A"]) # media veicoli A
```

```
## [1] 3.344
```

```r
mean(pm[km=="B"]) # media veicoli B
```

```
## [1] 2.70375
```

```r
table(km)
```

```
## km
## A B
## 5 8
```

```r
table(km)/length(km)
```

```
## km
##         A         B
## 0.3846154 0.6153846
```

```r
pm[11] <- 40.2 # dato anomalo
mean(pm)
```

```
## [1] 5.733077
```

Voti

Si consideri la variabile statistica qualitativa ordinale Y che descrive il voto di n = 5 studenti.

```
y <- ordered(c("S","S","B","B","O"),levels=c("S","B","O"))
#median(y) # R aspetta un vettore numerico
```

```
library("DescTools") # installare e caricare la libreria "DescTools"
```

```
## Warning: il pacchetto 'DescTools' è stato creato con R versione 4
```

```
Median(y) # utilizzare la funzione Median (M maiuscola)
```

```
## [1] B
## Levels: S < B < O
```

```
y <- ordered(c("S","S","S","B","B","O"),levels=c("S","B","O"))
Median(y) # la mediana e' sia S che B,
```

```
## Warning in Median.factor(y): Median is between two values; using
```

```
## [1] S
## Levels: S < B < O
```

```
# la funzione ritorna il primo dei due
```

```r
y <- ordered(c("S","S","B","B","O","O"),levels=c("S","B","O"))
Median(y) # i due valori sono entrambi B
```

```
## [1] B
## Levels: S < B < O
```

Serie TV

Si consideri la variabile statistica quantitativa discreta Y che descrive il numero di puntate, di una serie televisiva, viste da $n = 8$ famiglie.

```
y <- c(0,1,3,3,4,6,6,6)
median(y) # mediana convenzionale
```

```
## [1] 3.5
```

```
y <- c(0,1,3,3,4,6,6)
median(y)
```

```
## [1] 3
```

Componenti per famiglia

Sia Y la variabile quantitativa discreta che descrive il numero di componenti delle famiglie residenti in Liguria alla data del Censimento 1981.

Per la Liguria:

```
liguria <- c(rep(1,197906),rep(2,203709),rep(3,168536),rep(4,117509)
             rep(5,29727),rep(6,6577),rep(7,1707),rep(8,906))
length(liguria)
```

```
## [1] 726577
```

```
table(liguria)
```

```
## liguria
##      1      2      3      4      5      6      7      8
## 197906 203709 168536 117509  29727   6577   1707    906
```

```
cumsum(table(liguria))
```

```
##      1      2      3      4      5      6      7      8
## 197906 401615 570151 687660 717387 723964 725671 726577
```

```
table(liguria)/length(liguria)
```

```
## liguria
##           1           2           3           4           5
## 0.272381317 0.280368082 0.231958898 0.161729590 0.040913764 0.009
##           7           8
## 0.002349372 0.001246943
```

```
cumsum(table(liguria)/length(liguria))
```

```
##         1         2         3         4         5         6
## 0.2723813 0.5527494 0.7847083 0.9464379 0.9873517 0.9964037 0.998
```

```
median(liguria)
```

```
## [1] 2
```

```
table(campania)/length(campania)
```

```
## campania
##          1          2          3          4          5          6
## 0.14375298 0.19388154 0.17767022 0.22647683 0.14557059 0.06302321
##          8
## 0.02229741
```

```
cumsum(table(campania)/length(campania))
```

```
##         1         2         3         4         5         6
## 0.1437530 0.3376345 0.5153047 0.7417816 0.8873522 0.9503754 0.977
```

```
median(campania)
```

```
## [1] 3
```

Polveri sottili

```
pm <- c(2.30, 2.15, 3.50, 2.60, 2.75, 2.82, 4.05, 2.25, 2.68,
        3.00, 4.02, 2.85, 3.38)
median(pm)
```

```
## [1] 2.82
```

```
pm[11] <- 40.2 # dato anomalo
median(pm)
```

```
## [1] 2.82
```

Asfalto

Si considerano i dati relativi ai valori di resistenza alla rottura di $n = 24$ misture di asfalto (in megapascal).

```
y <- c(30,75,79,80,80,105,126,138,149,179,179,191,223,232,232,236,
       240,242,245,247,254,274,384,470)
mean(y)
```

```
## [1] 195.4167
```

```
median(y)
```

```
## [1] 207
```

```r
## R usa una procedura diversa per il calcolo del quantile convenzio
# basata sulla media pesata degli eventuali due valori
quantile(y,probs=c(0.25,0.5,0.75))
```

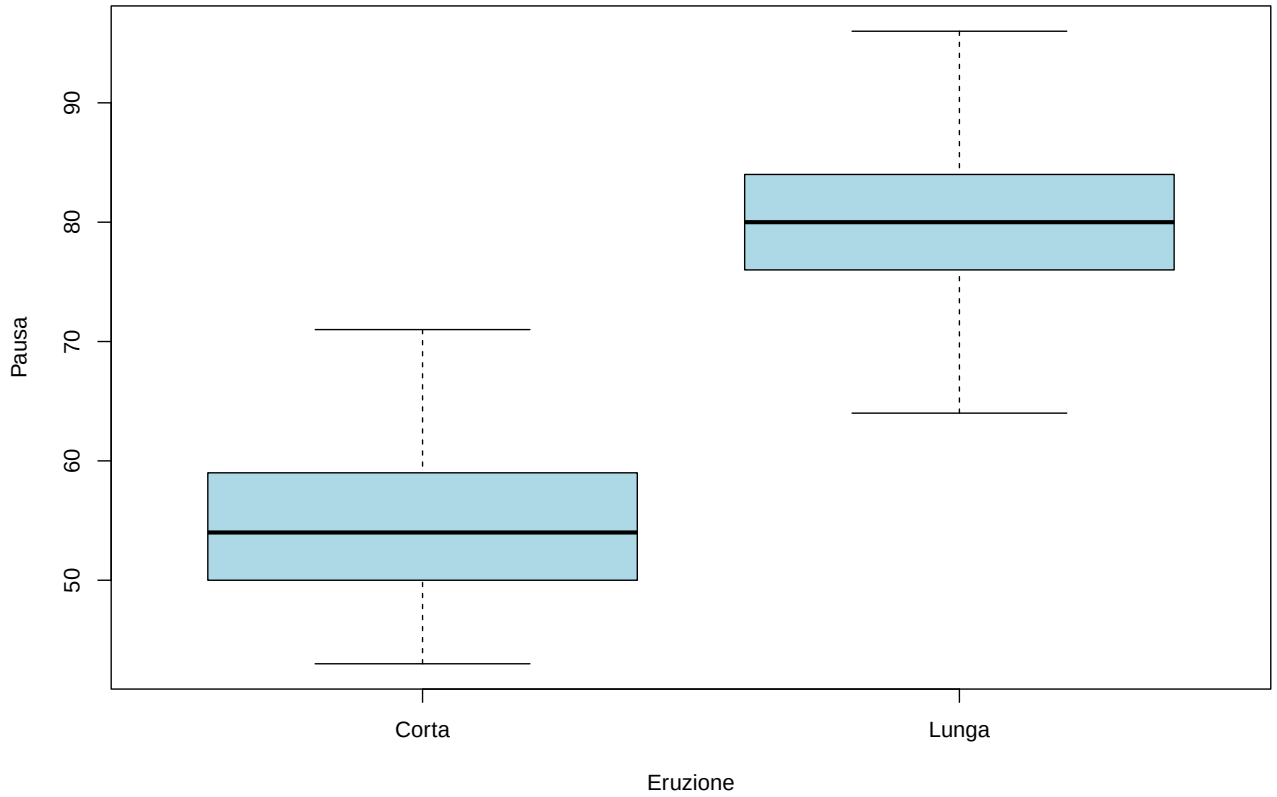
```
##     25%     50%     75%
## 120.750 207.000 242.750
```

Old Faithful Geyser

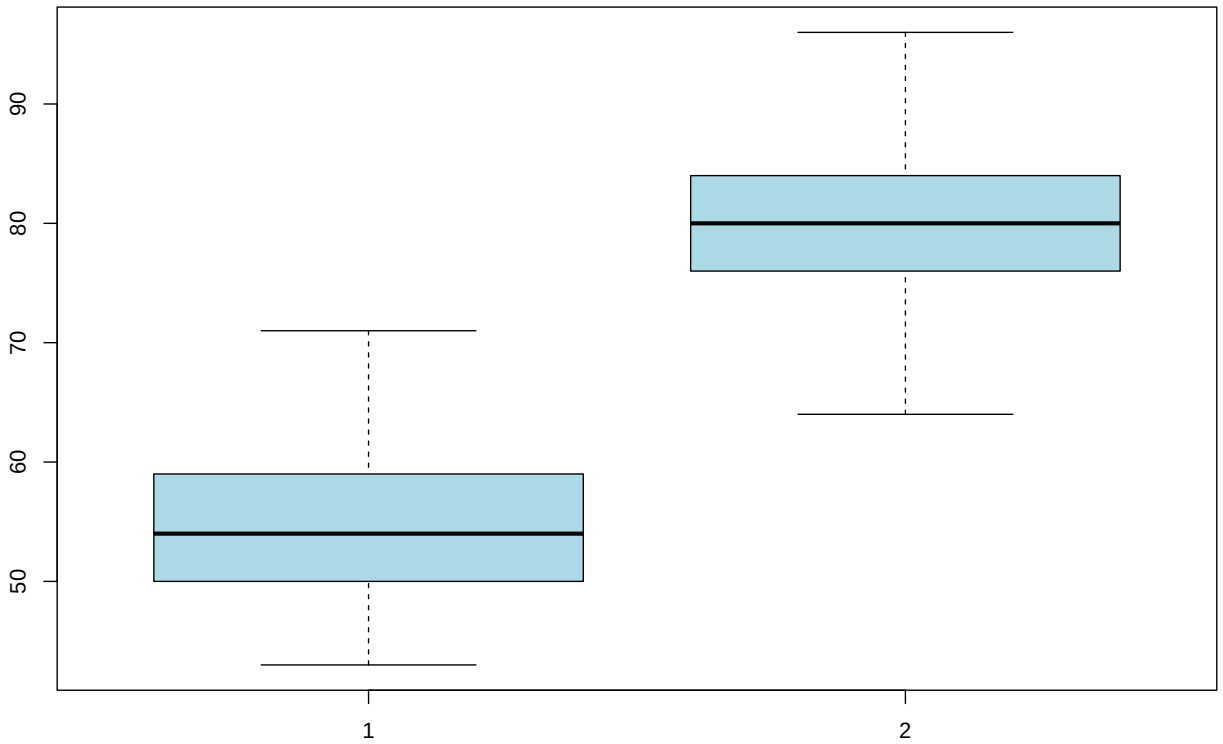
Si considerano i dati riferiti alle durate delle eruzioni del geyser Old Faithful.

```
duration<-ifelse(faithful$eruptions < 3,"Corta", "Lunga")
duration<-factor(duration)
faithful1<-data.frame(Pausa=faithful$waiting,Eruzione=duration)
```

```
boxplot(Pausa~Eruzione,data=faithful1,col="lightblue")
```



```
# in alternativa
boxplot(faithful1$Pausa[faithful1$Eruzione=="Corta"],
        faithful1$Pausa[faithful1$Eruzione=="Lunga"],
        col="lightblue")
```



Creta

Si vuole valutare la qualità della creta proveniente da due diverse cave. A tale scopo si rileva il numero medio di impurità per cm2 su 410 vasi, 180 costruiti con la creta della prima cava e 210 con la creta della seconda.

```
cava1<-c(rep(1,20),rep(2,40),rep(3,70),rep(4,35),
         rep(5,10),rep(6,5))
cava2<-c(rep(4,15),rep(5,20),rep(6,55),rep(7,80),
         rep(8,40),rep(9,20))
table(cava1)
```

```
## cava1
##  1  2  3  4  5  6
## 20 40 70 35 10  5
```

```
sum(table(cava1))
```

```
## [1] 180
```

```
table(cava2)
```

```
## cava2
##  4  5  6  7  8  9
## 15 20 55 80 40 20
```

```
sum(table(cava2))
```

```
## [1] 230
```

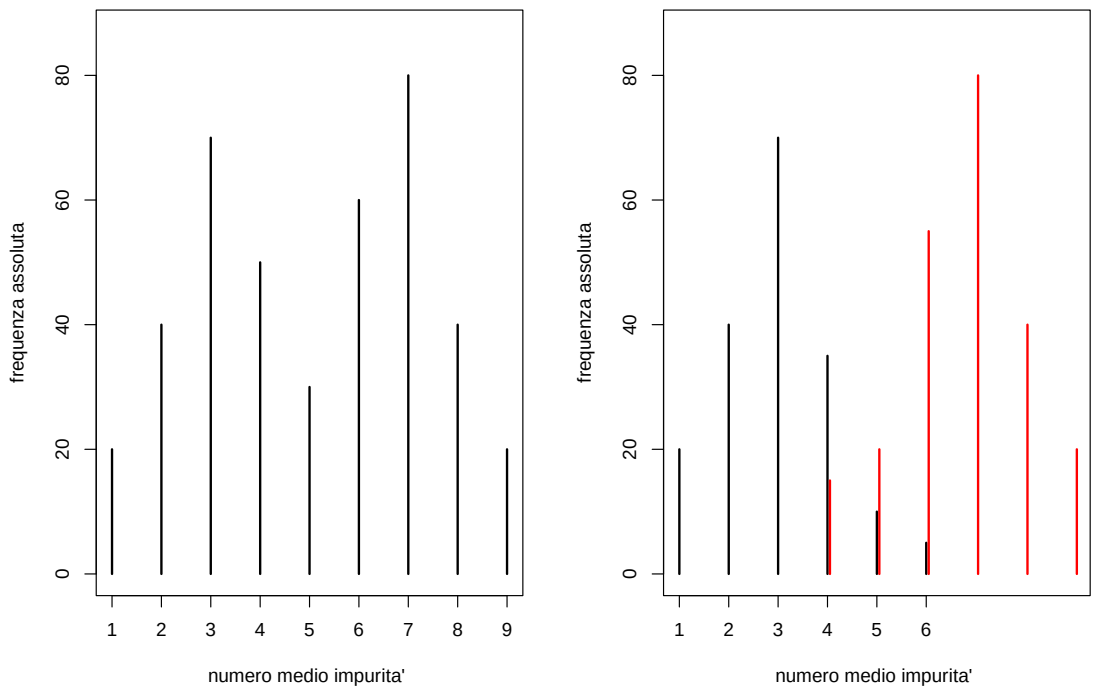
```
table(c(cava1,cava2))
```

```
##
##  1  2  3  4  5  6  7  8  9
## 20 40 70 50 30 60 80 40 20
```

```
sum(table(c(cava1,cava2)))
```

```
## [1] 410
```

```
par(mfrow=c(1,2))
plot(table(c(cava1,cava2)),xlab="numero medio impurita' ",
     ylab="frequenza assoluta",ylim=c(0,87),lwd=2)
plot(table(cava1),xlab="numero medio impurita' ",
     ylab="frequenza assoluta",xlim=c(1,9),ylim=c(0,87),lwd=2)
lines(seq(4,9,1)+0.05,table(cava2),type="h",col='red',lwd=2)
```




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

Sonnifero

Per valutare e confrontare l'effetto come sonnifero di due distinte molecole si sono considerati $n = 10$ volontari, senza storia pregressa di insonnia, ai quali è stato somministrato in una notte un placebo e in un'altra il sonnifero.

```
xx <- matrix(c(0.7,-1.6,-0.2,-1.2,-0.1,3.4,3.7,0.8,0,2,1.9,0.8,1.1,
               0.1,-0.1,4.4,5.5,1.6,4.6,4.6),ncol=2)
xx
```

```
##       [,1] [,2]
##  [1,]  0.7  1.9
##  [2,] -1.6  0.8
##  [3,] -0.2  1.1
##  [4,] -1.2  0.1
##  [5,] -0.1 -0.1
##  [6,]  3.4  4.4
##  [7,]  3.7  5.5
##  [8,]  0.8  1.6
##  [9,]  0.0  4.6
## [10,]  2.0  4.6
```

```r
summary(xx[,1]) # il calcolo dei quantili convenzionali
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   -1.600  -0.175   0.350   0.750   1.700   3.700
```

```r
# si basa su una opportuna media pesata
quantile(xx[,1],type=2)
```

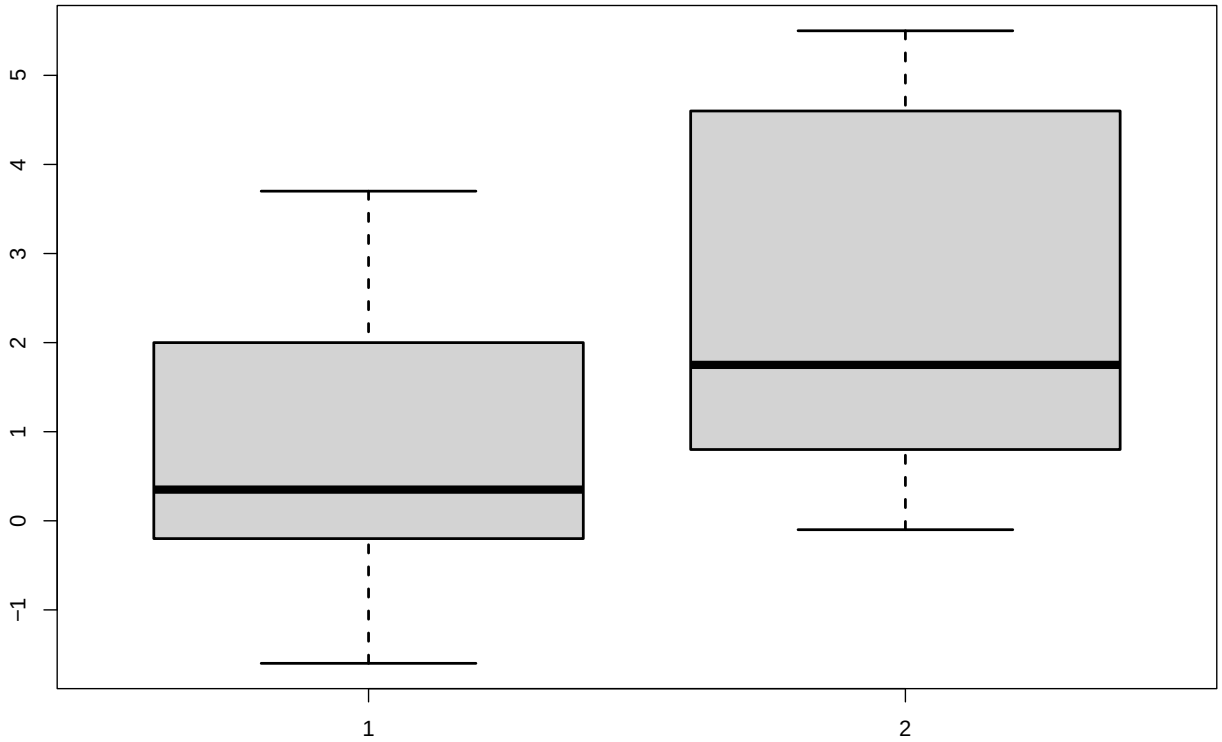
```
##    0%   25%   50%   75%  100%
## -1.60 -0.20  0.35  2.00  3.70
```

```r
summary(xx[,2]) # il calcolo dei quantili convenzionali
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   -0.100   0.875   1.750   2.450   4.550   5.500
```

```r
# si basa su una opportuna media pesata
```

```
boxplot(xx, at= 1:2,lwd=2)
```



Inquinamento

Per confrontare l'efficacia di due diversi dispositivi per contenere l'inquinamento atmosferico si sono analizzati i fumi prodotti da una certa industria. Si sono considerati 360 campioni di fumo e si è misurata la quantità di pulviscolo inquinante in g/min. In 180 si è utilizzato il dispositivo anti-inquinante A, mentre sui campioni rimanenti si è utilizzato il dispositivo anti-inquinante B.

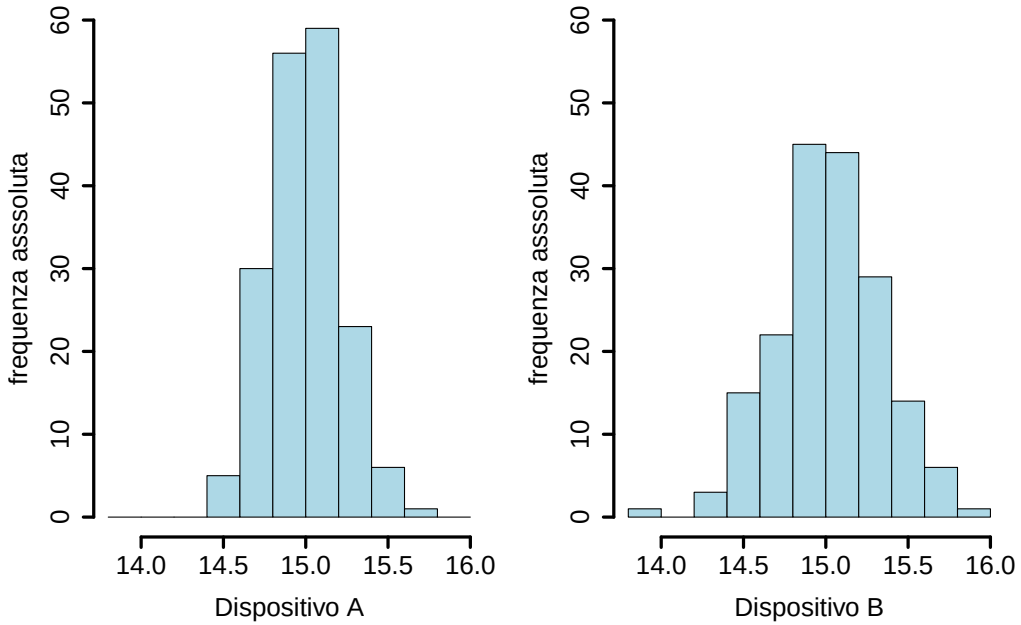
```
summary(dispA)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.44   14.87   15.00   15.00   15.14   15.64
```

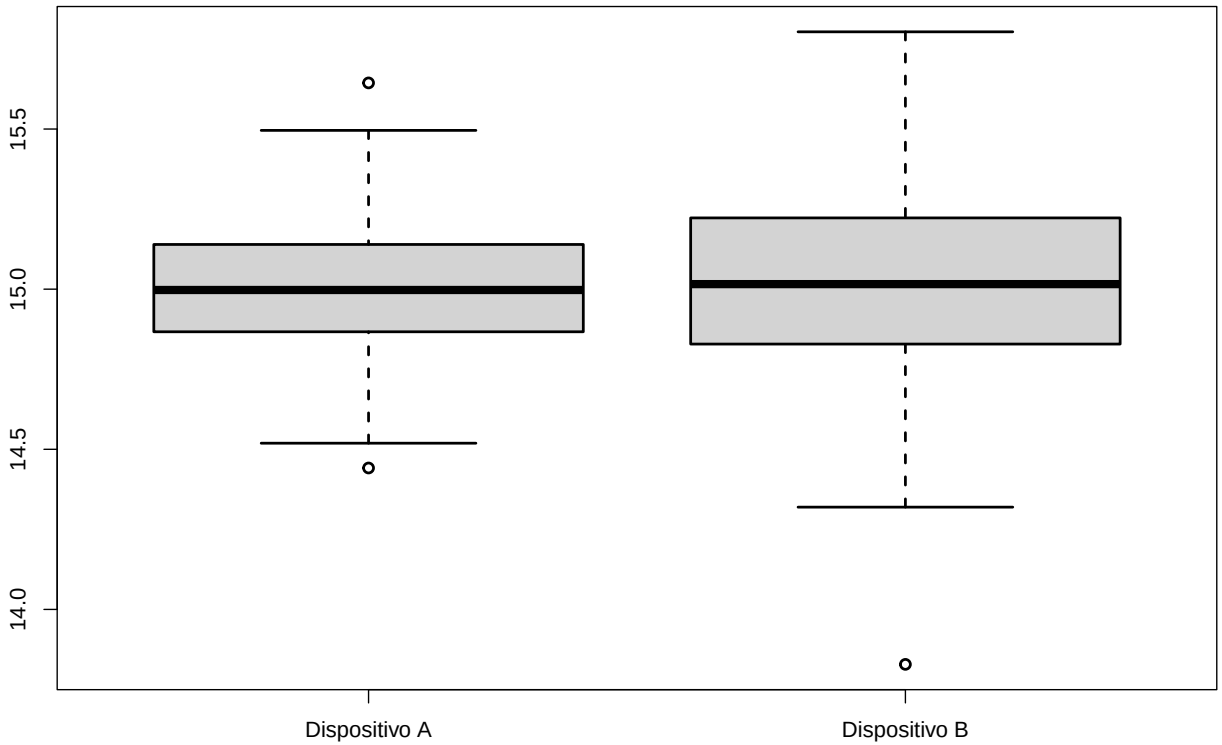
```
summary(dispB)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   13.83   14.83   15.02   15.02   15.22   15.80
```

```
par(mfrow=c(1,2))
hist(dispA,breaks=c(13.8,14,14.2,14.4,14.6,14.8,15,15.2,15.4,15.6,
                    15.8,16),lwd=3,xlab="Dispositivo A",
     ylim=c(0,60),ylab="frequenza asssoluta",
     cex.axis=1.5, cex.lab=1.5,col="lightblue",main=" ")
hist(dispB,breaks=c(13.8,14,14.2,14.4,14.6,14.8,15,15.2,15.4,15.6,
                    15.8,16),lwd=3,xlab="Dispositivo B",
     ylim=c(0,60),ylab="frequenza asssoluta",
     cex.axis=1.5, cex.lab=1.5,col="lightblue",main=" ")
```



```
boxplot(dispA, dispB, names=c("Dispositivo A", "Dispositivo B"),
        at=1:2,lwd=2)
```



```
range(dispA) # fornisce i valori di min e max,
```

```
## [1] 14.44163 15.64405
```

```
#in alternativa max(dispA)-min(dispA)

range(dispB) # fornisce i valori di min e max,
```

```
## [1] 13.82846 15.80354
```

```
#in alternativa max(dispB)-min(dispB)

IQR(dispA) #in alternativa quantile(dispA,0.75)-quantile(dispA,0.25)
```

```
## [1] 0.2699275
```

```
IQR(dispB) #in alternativa quantile(dispB,0.75)-quantile(dispB,0.25)
```

```
## [1] 0.3929925
```

```
var(dispA)*(length(dispA)-1)/length(dispA)
```

```
## [1] 0.0456162
```

```
#var() divide per n-1 invece che per n
var(dispB)*(length(dispB)-1)/length(dispB)
```

```
## [1] 0.09901038
```

```
sqrt(var(dispA)*(length(dispA)-1)/length(dispA))
```

```
## [1] 0.2135795
```

```
sqrt(var(dispB)*(length(dispB)-1)/length(dispB))
```

```
## [1] 0.3146591
```

Esempio 1 varianza

```
y <- c(rep(0,109),rep(1,65),rep(2,22),rep(3,3),rep(4,1))
mean(y)
```

```
## [1] 0.61
```

```
var(y)*199/200
```

```
## [1] 0.6079
```

```
sqrt(var(y)*199/200)
```

```
## [1] 0.7796794
```

```
table(y)
```

```
## y
##   0   1   2   3   4
## 109  65  22   3   1
```

```
media<-sum(seq(0,4,1)*table(y))/sum(table(y))
sum((seq(0,4,1)-media)^2*table(y))/sum(table(y))
```

```
## [1] 0.6079
```

```
# calcolo della varianza con la tabella di frequenza
sum(seq(0,4,1)^2*table(y))/sum(table(y))-media^2
```

```
## [1] 0.6079
```

```
# calcolo della varianza con la formula di calcolo
```

Esempio 2 varianza

```
yc <- c((0+10)/2,(10+15)/2,(15+20)/2)
p <- c(0.3,0.52,0.18)
media<-sum(yc*p) # media aritmetica
sum((yc-media)^2*p)
```

```
## [1] 19.5525
```

```
sum(yc^2*p)-media^2
```

```
## [1] 19.5525
```

Sbarchi

Si consideri la seguente tabella di frequenza che riporta le merci e i passeggeri sbarcati, con riferimento agli scali portuali di alcune regioni italiane nel 1988.

```
merci<-c(22806,21849,12627,4937)
passeggeri<-c(42,248,3,266)
tab=as.data.frame(cbind(merci,passeggeri))
row.names(tab)=c("Friuli V. - G.","Veneto","Emilia-R.","Marche")
tab
```

```
##                merci passeggeri
## Friuli V. - G. 22806         42
## Veneto         21849        248
## Emilia-R.      12627          3
## Marche          4937        266
```

```
mean(merci)
```

```
## [1] 15554.75
```

```
mean(passeggeri)
```

```
## [1] 139.75
```

```
var(merci)*3/4
```

```
## [1] 53376636
```

```
var(passeggeri)*3/4
```

```
## [1] 13978.19
```

```
sqrt(var(merci)*3/4)/mean(merci)
```

```
## [1] 0.4696914
```

```
sqrt(var(passeggeri)*3/4)/mean(passeggeri)
```

```
## [1] 0.8460063
```

Esempio simmetria

```
x1<-c(2.17, 2.8, -0.85, 2.38, 1.05, 1.2, -0.46, 1.98, 2.13, 1.67,
      -0.34, 1.77, 2.26, 2.05, 0.88, 0.43, 2.34, 1.17, 0.79, 1.95,
      1.87, 1.41, 1.2, 2.22, 2.47, 2.42, 0.8, 1.39, 2.26, 1.62,
      0.48, 1.38, 2.21, 1.67, 0.71, 1.59, 0.76, 2.25, 1.44, 1.33,
      2.17, 1.46, 1.99, 1.62, -1.82, 2.39, 0.08, -0.61, -1.15, 2.29)
mean(x1)
```

```
## [1] 1.3454
```

```
median(x1)
```

```
## [1] 1.605
```

```
x2<-c(0.24, -1.49, 0.29, 0.14, 0.25, -0.84, 0.81, -0.75, 0.82,
      -1.19, -1.56, 1.14, 1.22, -1.5, -0.12, 0.06, 0.41, 1.32,
      -0.18, -0.58, 0.55, 0.16, 0.39, -0.17, 0.14, -0.79, -0.22,
      -0.4, 1.19, -0.45, -1.6, 1.99, -0.94, 0.14, 1.86, -0.1,
      0.66, -0.34, -0.62, -0.56, -1.17, -0.93, -2.38, 2.01,
      0.68, 0.36, 0.64, -0.17, -0.05, 0.67)
mean(x2)
```

```
## [1] -0.0192
```

```
median(x2)
```

```
## [1] 0.005
```

```
x3<-c(2.57, 1.8, 1.6, 2.14, 4.76, 6.52, 4.32, 2.13, 2.06, 8.99, 2,
      2.29, 2.72, 2.17, 3.49, 3.31, 2.22, 2.32, 2.89, 2.83, 2.67,
      2.07, 2.46, 7.86, 5.1, 3.29, 6.21, 4.47, 2.22, 3.1, 2.86,
      2.23, 2.33, 2.14, 2.64, 3.24, 3.15, 5.77, 2.33, 2.25, 2.71,
      2.87, 2.69, 2.44, 2.46, 2.34, 2.33, 2.75, 2.69, 2.06)
mean(x3)
```

```
## [1] 3.1372
```

```
median(x3)
```

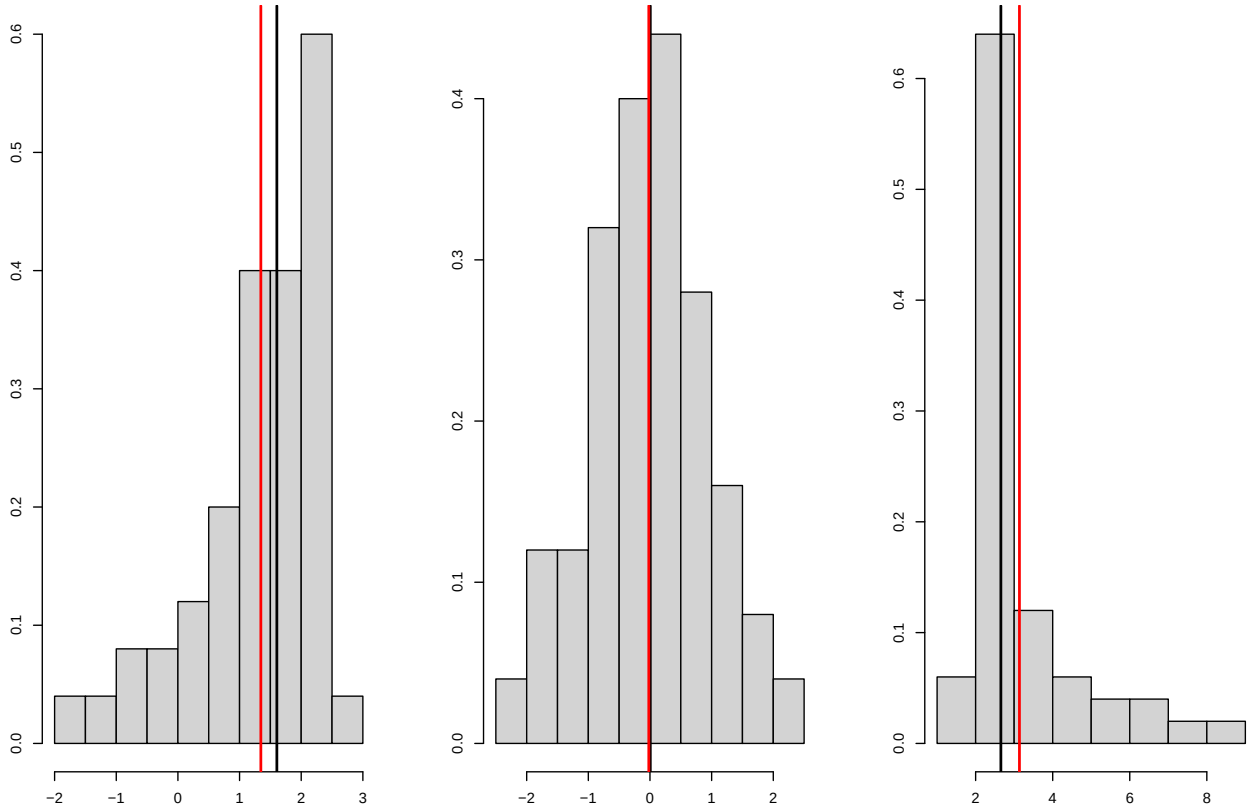
```
## [1] 2.655
```

Esempio simmetria

```
par(mfrow=c(1,3))
hist(x1,xlab="",ylab="",xlim=c(-2,3),main="",freq=FALSE)
abline(v=median(x1),lwd=2)
abline(v=mean(x1),lwd=2,col="red")

hist(x2,xlab="",ylab="",main="",freq=FALSE)
abline(v=median(x2),lwd=2)
abline(v=mean(x2),lwd=2,col="red")

hist(x3,xlab="",ylab="",main="",freq=FALSE)
abline(v=median(x3),lwd=2)
abline(v=mean(x3),lwd=2,col="red")
```

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

Esempio simmetria



```
library("moments") # installare e caricare la libreria "moments"
skewness(x1)
```

```
## [1] -1.169473
```

```
skewness(x2)
```

```
## [1] -0.008665172
```

```
skewness(x3)
```

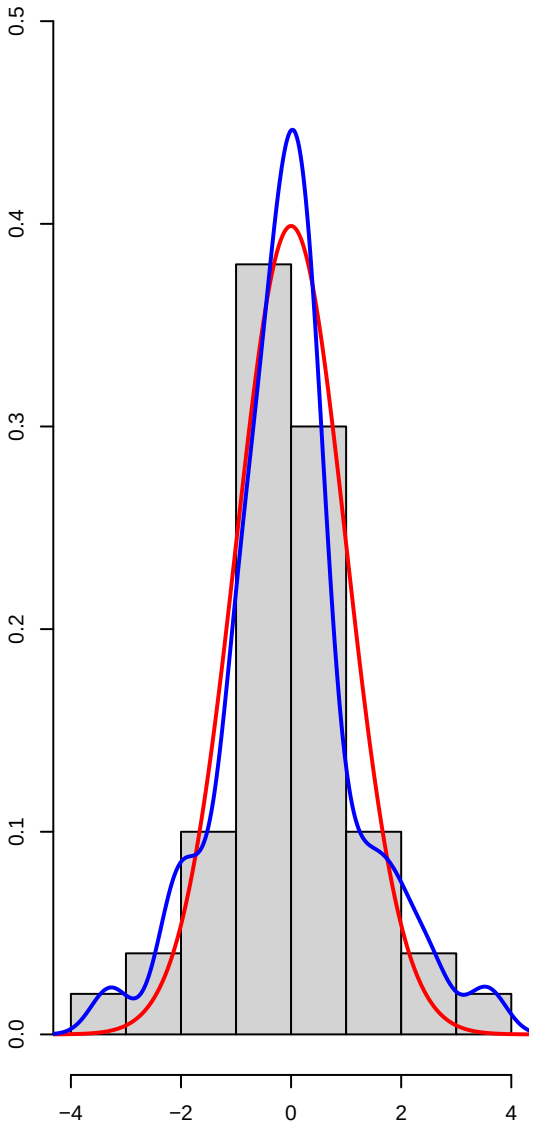
```
## [1] 2.12828
```

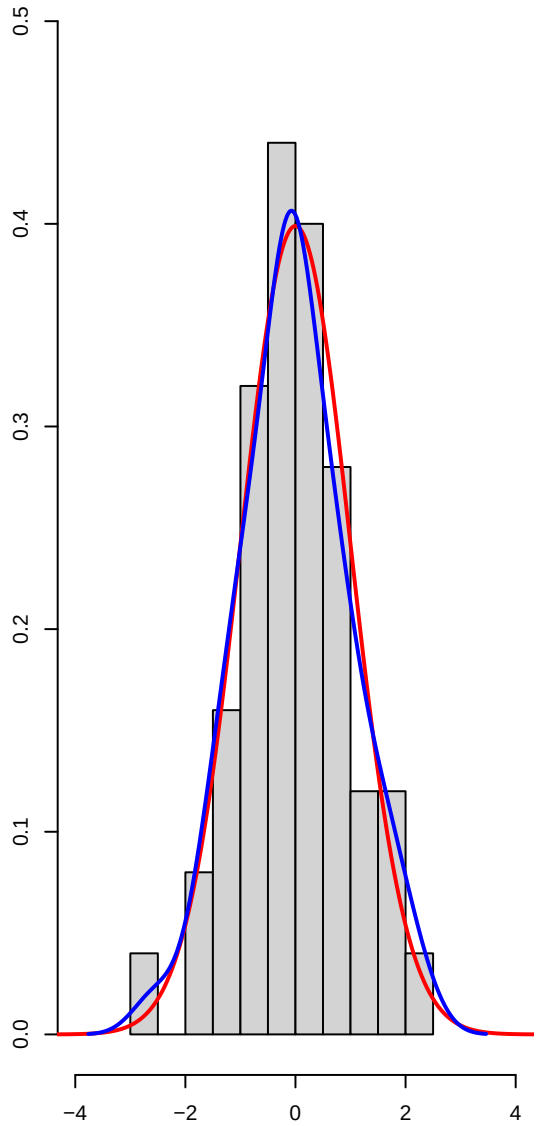
Esempio curtosi

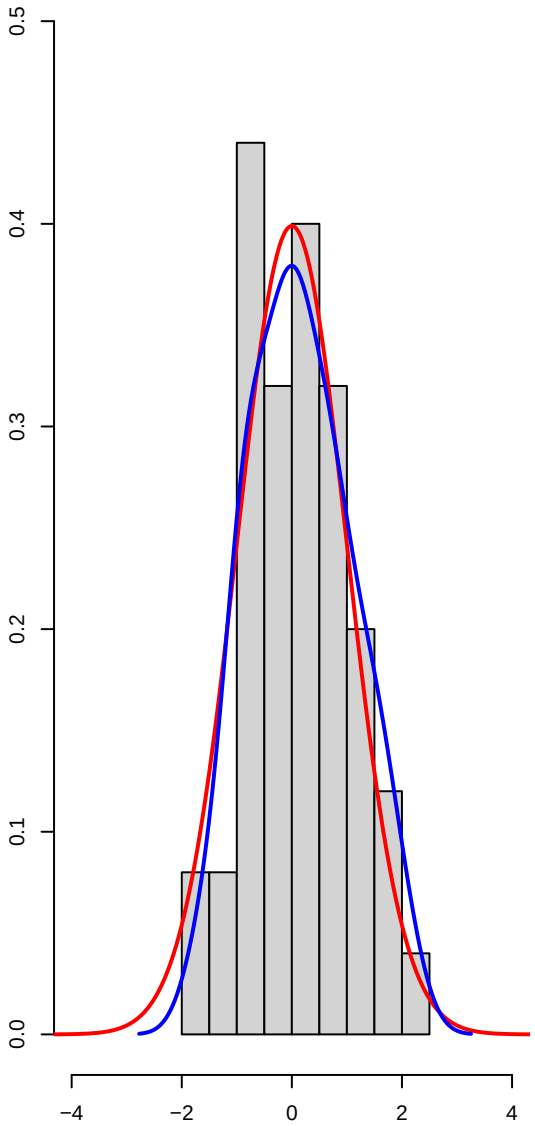
```
y1 <-c(-0.12, 0.02, 0.43, -0.12, -0.39, -1.11, 0.73, 1.9, 0.81,
       -0.8, 0.3, 0.18, -0.99, 1.33, 2.61, 1.04, 1.5, -0.83, 0.42,
       0.24, -0.87, -0.15, -1.7, -0.04, -0.3, 1.74, -0.55, -0.76,
       0.34, -1.99, -0.45, -0.42, 0.41, -1.28, -0.1, 0.17, 3.55,
       2.23, 0.31, -1.05, -0.32, 0, 0.48, -0.59, -2.22, -3.28,
       0.02, -0.11, -2.02, 0.36)
y2 <- c(-0.53, 1.21, 0.75, -0.22, 0.89, 0.79, 0.14, -0.22, 0.87,
        0.39, -0.2, 0.39, -0.14, 0.61, 1.32, -1.38, -0.9, -0.21,
        -0.03, 0.01, -2.54, 0.12, -0.12, 0.09, -0.22, 2.24,
        -0.61, -1.45, -0.89, -0.2, -0.75, 0.8, 1.66, -1.1,
        -0.85, -0.91, -0.82, 0.74, -1.2, 1.64, -1.71, 0.45,
        0.33, -0.44, 0.06, 0.09, -0.21, 1.37, -1.57, 1.9)
y3 <- c(-0.53, 1.21, 0.75, 0.22, 0.89, 0.79, 0.14, -0.22, 0.87,
        1.69, -0.2, 0.39, -0.14, 0.61, 0.72, -1.01, -0.9, -1.51,
        -0.03, 1.41, -0.54, 2.12, -0.12, 0.09, -0.22, 1.24,
        -0.61, 0.45, -0.89, -0.2, -0.75, 0.8, 1.66, 0.1, -0.85,
        -0.91, -0.82, 0.74, -1.2, 1.44, -0.71, 0.45, 0.33,
        -1.64, 0.06, 0.09, -0.21, 1.37, -0.87, 1.9)
```

```
par(mfrow=c(1,3))
hist(y1,freq = F,xlim=c(-4,4),ylim=c(0,0.5),xlab='',ylab='',main='')
lines(seq(-5,5,0.01),dnorm(seq(-5,5,0.01)),col='red',lwd=2)
lines(density(y1),col='blue',lwd=2)

hist(y2,freq = F,xlim=c(-4,4),ylim=c(0,0.5),xlab='',ylab='',main='')
lines(seq(-5,5,0.01),dnorm(seq(-5,5,0.01)),col='red',lwd=2)
lines(density(y2),col='blue',lwd=2)

hist(y3,freq = F,xlim=c(-4,4),ylim=c(0,0.5),xlab='',ylab='',main='')
lines(seq(-5,5,0.01),dnorm(seq(-5,5,0.01)),col='red',lwd=2)
lines(density(y3),col='blue',lwd=2)
```

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


0.0
0.1
0.2
0.3
0.4
0.5
−4
−2
0
2
4


0.0
0.1
0.2
0.3
0.4
0.5
−4
−2
0
2
4


0.0
0.1
0.2
0.3
0.4
0.5
−4
−2
0
2
4

Indice di curtosi del pacchetto “moments”

```
library("moments")
kurtosis(y1)
```

```
## [1] 4.116753
```

```
kurtosis(y2)
```

```
## [1] 2.926545
```

```
kurtosis(y3)
```

```
## [1] 2.289446
```

Inquinamento

```
library("moments")
skewness(dispA)
```

```
## [1] 0.09535216
```

```
skewness(dispB)
```

```
## [1] -0.2245719
```

```
kurtosis(dispA)
```

```
## [1] 2.937236
```

```
kurtosis(dispB)
```

```
## [1] 3.397726
```