ESERCIZI STATISTICA DESCRITTIVA

2022-10-24

STATISTICA DESCRITTIVA

```
library("dplyr")
```

```
##
## Caricamento pacchetto: 'dplyr'
```

```
## I seguenti oggetti sono mascherati da 'package:stats':
##
##     filter, lag
```

```
## I seguenti oggetti sono mascherati da 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("ggplot2")
```

A.1)

La tabella rappresenta 40 osservazioni della lunghezza (in mm) di una foglia di platano.

138	164	150	132	144	125	149	157
146	158	140	147	136	148	152	144
168	126	138	176	163	119	154	165
146	173	142	147	135	153	140	135
161	145	135	142	150	156	145	128

- VARIABILE Y = Lunghezza: QUANTITATIVA CONTINUA

```
#vettore unidimensionale in quanto vi è un'unica variabile di riferimento
Y = c(138,164,150,132,144,125,149,157,146,158,140,147,136,148,152,144,168,126,138,176,163,119,154,165,14
(n=length(Y)) #numero di osservazioni totali
```

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

```
# SUPPORTO DI Y
(Sy=unique(Y))
```

```
##  [1] 138 164 150 132 144 125 149 157 146 158 140 147 136 148 152 168 126 176 163
## [20] 119 154 165 173 142 135 153 161 145 156 128
```

- Suddivisione del supporto in classi:
  - $118 \dashv 128, 128 \dashv 138, ..., 168 \dashv 178$

```
# frequenze assolute per ogni valore del supporto
Fy = table(Y)
# frequenze relative per ogni valore del supporto
Py = table(Y)/length(Y)

FreqTable = data.frame(table(Y))


data.frame(cumsum(table(Y)))
```

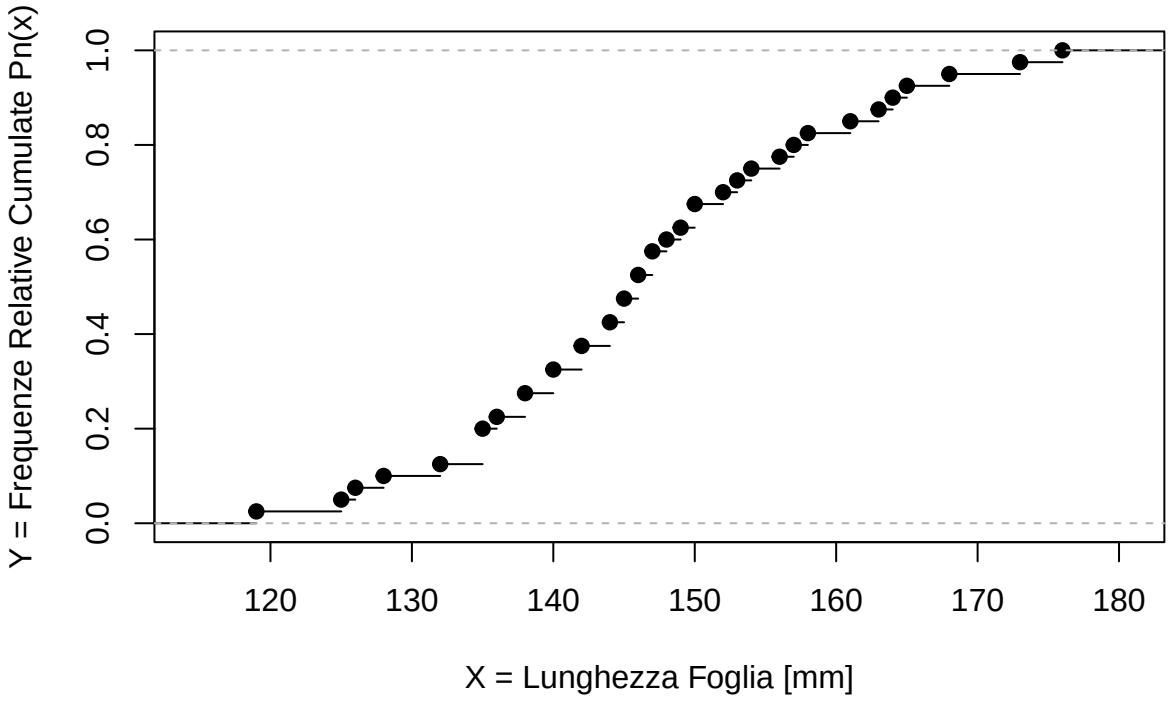
```
##     cumsum.table.Y..
## 119                1
## 125                2
## 126                3
## 128                4
## 132                5
## 135                8
## 136                9
## 138               11
## 140               13
## 142               15
## 144               17
## 145               19
## 146               21
## 147               23
## 148               24
## 149               25
## 150               27
## 152               28
## 153               29
## 154               30
## 156               31
## 157               32
## 158               33
## 161               34
## 163               35
## 164               36
## 165               37
## 168               38
## 173               39
## 176               40
```

```
#frequenze relative cumulate
(FreqRelCum=   data.frame(Y=sort(Sy),Pn=cumsum(table(Y)/length(Y))))
```

```
##       Y    Pn
## 119 119 0.025
## 125 125 0.050
## 126 126 0.075
## 128 128 0.100
## 132 132 0.125
## 135 135 0.200
```

```
## 136 136 0.225
## 138 138 0.275
## 140 140 0.325
## 142 142 0.375
## 144 144 0.425
## 145 145 0.475
## 146 146 0.525
## 147 147 0.575
## 148 148 0.600
## 149 149 0.625
## 150 150 0.675
## 152 152 0.700
## 153 153 0.725
## 154 154 0.750
## 156 156 0.775
## 157 157 0.800
## 158 158 0.825
## 161 161 0.850
## 163 163 0.875
## 164 164 0.900
## 165 165 0.925
## 168 168 0.950
## 173 173 0.975
## 176 176 1.000
```

```r
# ecdf() funzione di ripartizione empirica
plot(ecdf(Y),main=" ",xlab="X = Lunghezza Foglia [mm]", ylab="Y = Frequenze Relative Cumulate Pn(x)")
```




```
classIntervals=c(118,128,138,148,158,168,178)

# tabella delle frequenze assolute per le classi di valori
table(cut(Y,classIntervals))
```

```
##
## (118,128] (128,138] (138,148] (148,158] (158,168] (168,178]
##         4         7        13         9         5         2
```

```
# frequenza assolute cumulate
cumFreqAss = cumsum(
  table(cut(Y,classIntervals))
)


#tabella frequenze relative
table(
  cut(Y , classIntervals )
)/length(Y)
```

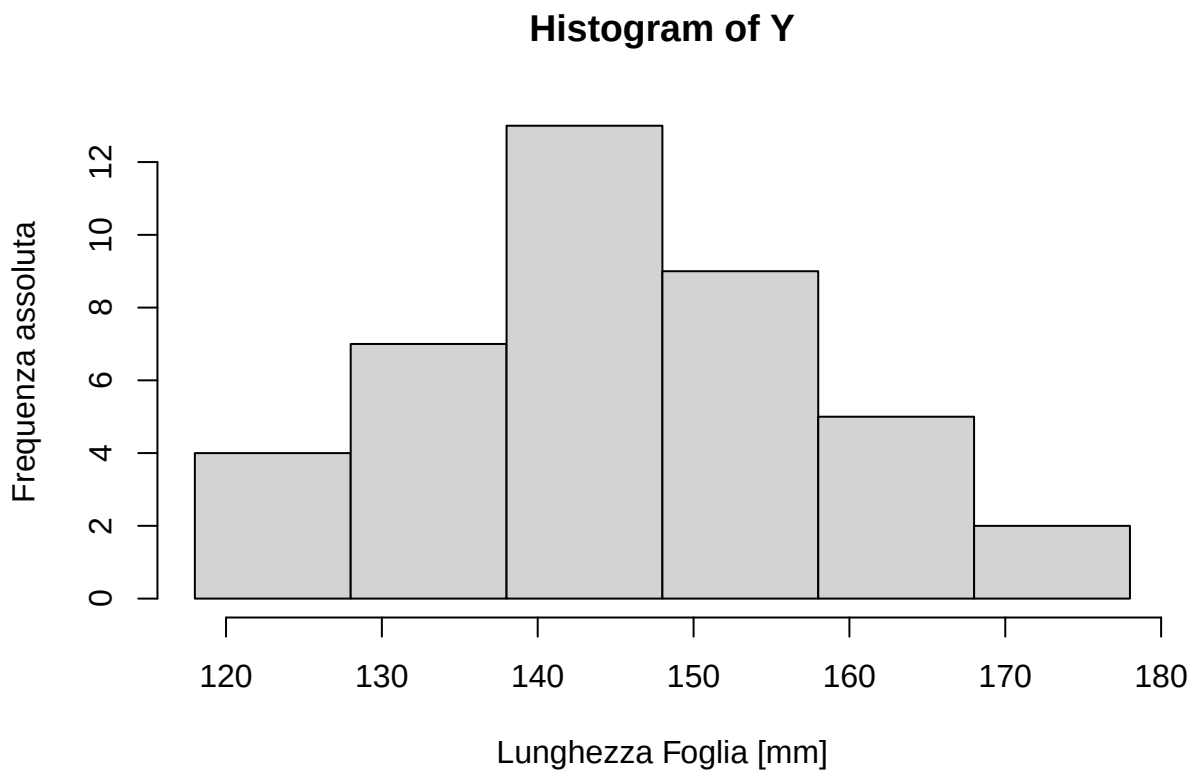
```
##
## (118,128] (128,138] (138,148] (148,158] (158,168] (168,178]
##     0.100     0.175     0.325     0.225     0.125     0.050
```

```
# frequenze relative cumulate
cumsum(
  table(cut(Y,classIntervals))/length(Y)
)
```

```
## (118,128] (128,138] (138,148] (148,158] (158,168] (168,178]
##     0.100     0.275     0.600     0.825     0.950     1.000
```

ISTOGRAMMA

```
# crea l'istogramma impostando le basi dell'asse X come le classi definite in precedenza
hist(Y,classIntervals,xlab = "Lunghezza Foglia [mm]",ylab = "Frequenza assoluta")
```



POLIGONO DI FREQUENZA

INDICI

```
moda = function(v){
  tmp = unique(v)
 Sy[which.max(tabulate(match(v,tmp)) )]
}
(modaY=moda(Y))
```

MODA

```
## [1] 135
```

```
(medianaY = median(Y))
```

MEDIANA

```
## [1] 146
```

```
# frequenze relative cumulate
freqRelCum = cumsum(table(Y)/length(Y))



quantile(Y,c(0.25,0.5,0.75))
```

```
##   25%   50%   75%
## 138.0 146.0 154.5
```

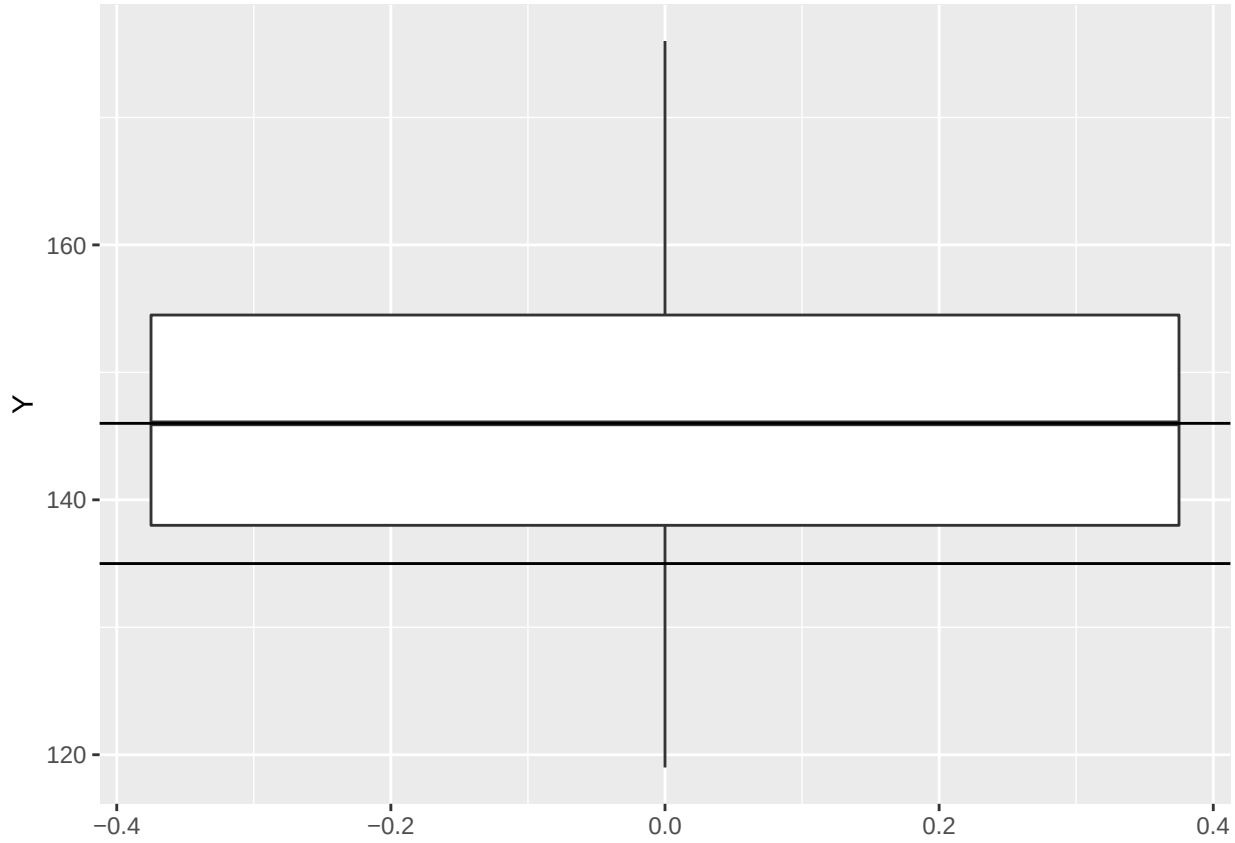
```
median(Y) == quantile(Y,0.5)
```

```
##  50%
## TRUE
```

BOXPLOT

```
ggplot(data.frame(Y)) + geom_boxplot(aes(y=Y)) + geom_hline(yintercept = modaY)  + geom_hline(yintercep
```

```
## Warning: geom_hline(): Ignoring `mapping` because `yintercept` was provided.
```



CAMPO DI VARIAZIONE

RANGE DI VALORI

```
(dimRange = max(Y)-min(Y))
```

```
## [1] 57
```

```
(range = c(min(Y),max(Y)))
```

```
## [1] 119 176
```

SCARTO INTERQUANTILICO

```
(Scy = quantile(Y,0.75)-quantile(Y,0.25))
```

```
##  75%
## 16.5
```

```
FreqRelCum[FreqRelCum$Pn>=0.75,"Y"][1] - FreqRelCum[FreqRelCum$Pn>=0.25,"Y"][1]
```

```
## [1] 16
```

A.2)

No. guasti	Autovetture FIAT	Autovetture OPEL
0	9	33
1	13	20
2	10	6
3	5	1
4	3	0
Totale	40	60

Y = numero di guasti delle autovetture di marce OPEL e FIAT

Quantitaviva discreta == qualitativa nominale

```
Sy_FIAT = c(0,1,2,3,4)   #Supporto di Y_FIAT ,
Sy_OPEL = c(0,1,2,3)


(FreqFIAT = data.frame(nGuasti = c(0,1,2,3,4), fi = c(9,13,10,5,3)))
```

```
##   nGuasti fi
## 1       0  9
## 2       1 13
## 3       2 10
## 4       3  5
## 5       4  3
```

```
# calcola le frequenze relative
(FreqFIAT = FreqFIAT %>% mutate(pi=fi/sum(fi)))
```

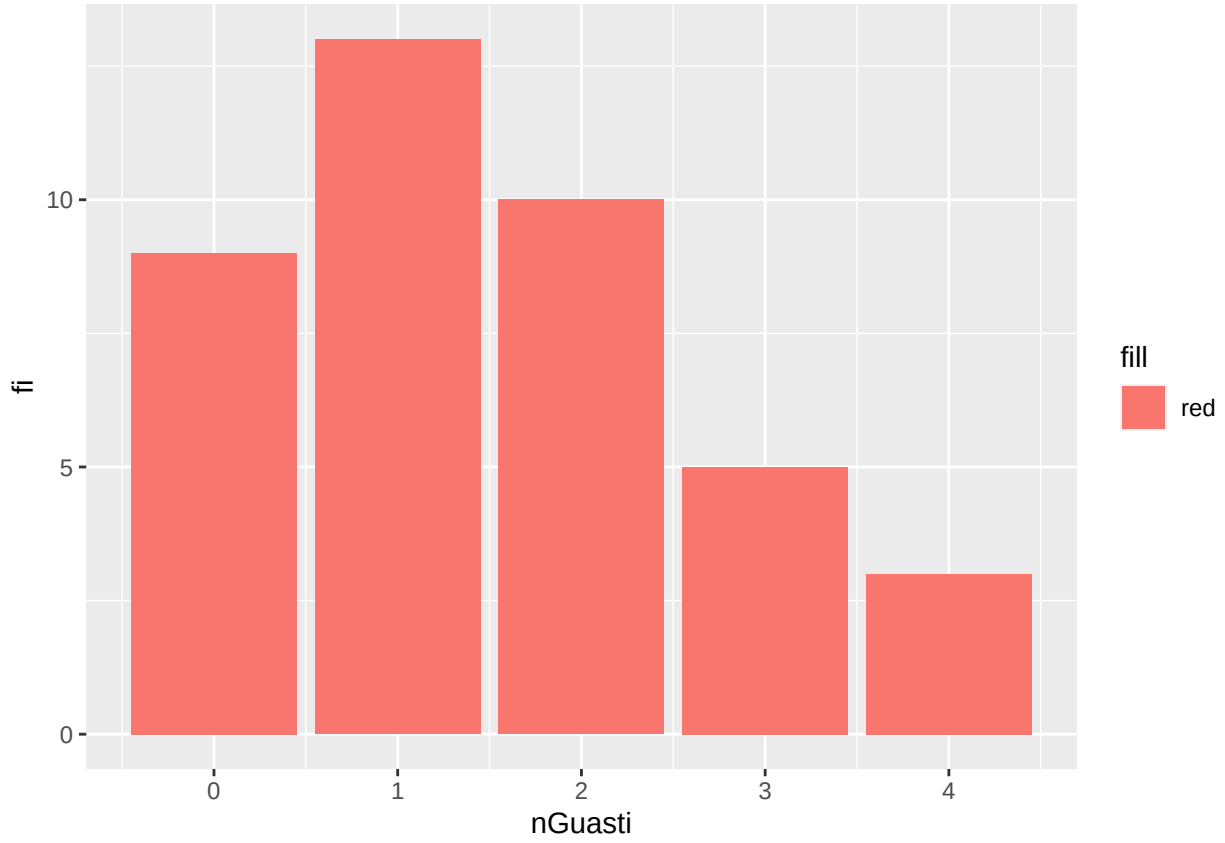
```
##   nGuasti fi    pi
```

```
## 1       0  9 0.225
## 2       1 13 0.325
## 3       2 10 0.250
## 4       3  5 0.125
## 5       4  3 0.075
```

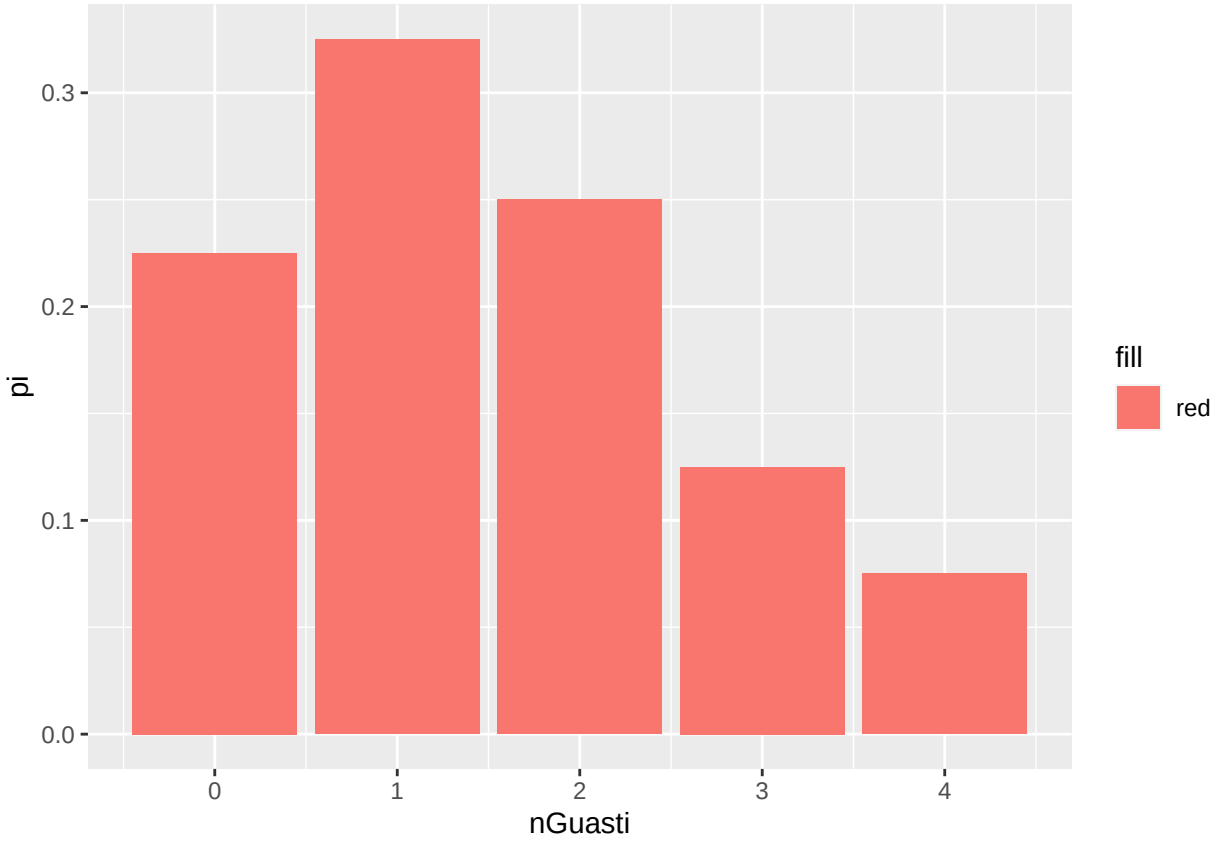
```
if (sum(FreqFIAT$pi) == 1){
  print("LA SOMMA DELLE FREQUENZE RELATIVE CORRISPONDE AL 100%")
}
```

```
## [1] "LA SOMMA DELLE FREQUENZE RELATIVE CORRISPONDE AL 100%"
```

```
ggplot(FreqFIAT) + geom_col(aes(nGuasti,fi,fill="red"))
```




```
# I DUE GRAFICI SONO ANALOGHI
ggplot(FreqFIAT) + geom_col(aes(nGuasti,pi,fill="red"))
```




```
# funzione per il calcolo delle frequenze cumulate assolute o relative
calcolaFreqCum = function(arr){
  a = c(arr[1])
  for (i in 2:length(arr)){
    a = append(a,arr[i]+a[i-1])
  }
  return (a)
}
```

```
(FreqFIAT = FreqFIAT %>% mutate(Fi=calcolaFreqCum(fi))    )
```

```
##   nGuasti fi    pi Fi
## 1       0  9 0.225  9
## 2       1 13 0.325 22
## 3       2 10 0.250 32
## 4       3  5 0.125 37
## 5       4  3 0.075 40
```

```
(FreqFIAT = FreqFIAT %>% mutate(Pi=calcolaFreqCum(pi))    )
```

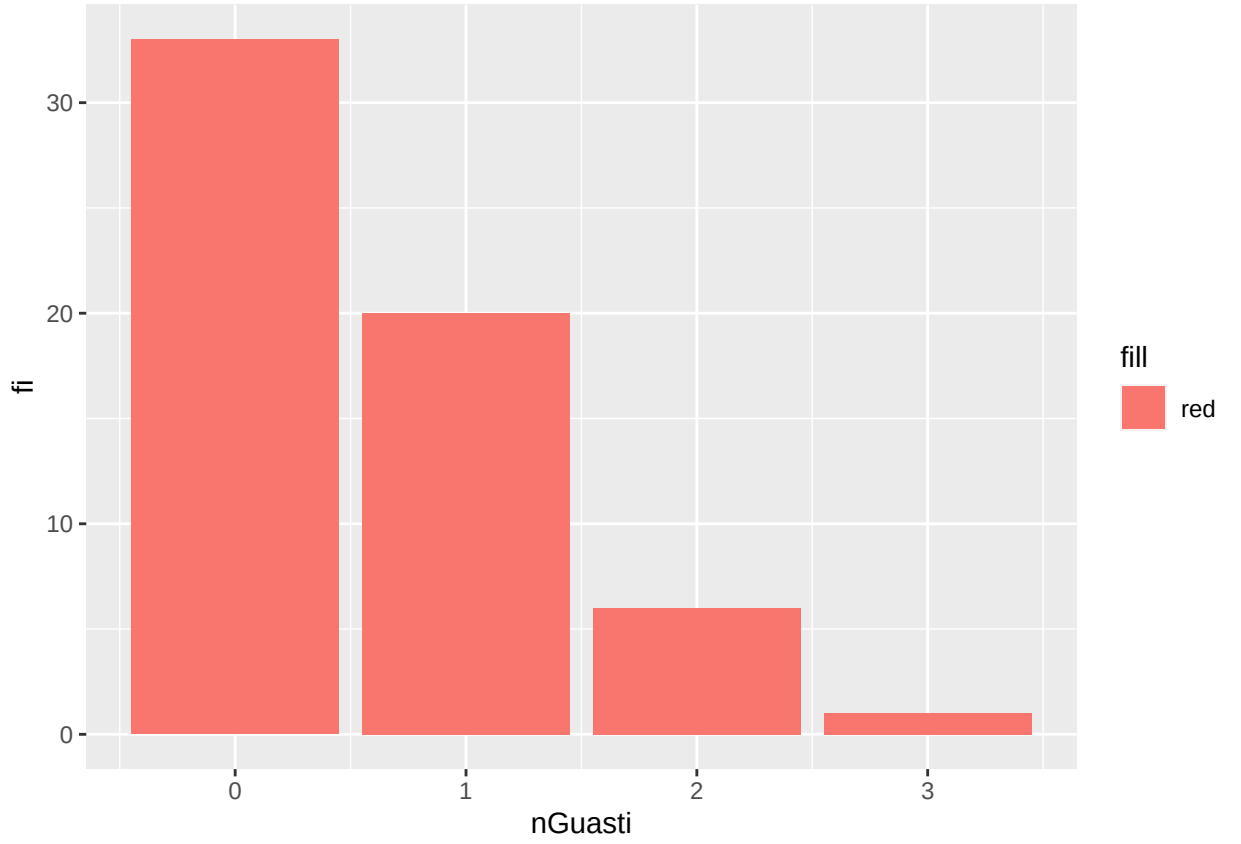
```
##   nGuasti fi    pi Fi    Pi
## 1       0  9 0.225  9 0.225
## 2       1 13 0.325 22 0.550
## 3       2 10 0.250 32 0.800
## 4       3  5 0.125 37 0.925
## 5       4  3 0.075 40 1.000
```

```r
FreqOPEL = data.frame(nGuasti = c(0,1,2,3), fi = c(33,20,6,1))
# calcola le frequenze relative
FreqOPEL = FreqOPEL %>% mutate(pi=fi/sum(fi))

if (sum(FreqOPEL$pi) == 1){
  print("LA SOMMA DELLE FREQUENZE RELATIVE CORRISPONDE AL 100%")
}
```

```
## [1] "LA SOMMA DELLE FREQUENZE RELATIVE CORRISPONDE AL 100%"
```

```r
ggplot(FreqOPEL) + geom_col(aes(nGuasti,fi,fill="red"))
```




```r
(FreqOPEL = FreqOPEL  %>% mutate(Fi=calcolaFreqCum(fi))  )
```

```
##   nGuasti fi         pi Fi
## 1       0 33 0.55000000 33
## 2       1 20 0.33333333 53
## 3       2  6 0.10000000 59
## 4       3  1 0.01666667 60
```

```r
(FreqOPEL = FreqOPEL  %>% mutate(Pi=calcolaFreqCum(pi))  )
```

```
##   nGuasti fi         pi Fi        Pi
## 1       0 33 0.55000000 33 0.5500000
## 2       1 20 0.33333333 53 0.8833333
## 3       2  6 0.10000000 59 0.9833333
## 4       3  1 0.01666667 60 1.0000000
```

MODA

Elemento con più occorrenze

```
(ModaFIAT = FreqFIAT[FreqFIAT$fi == max(FreqFIAT$fi),"nGuasti" ])
```

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

```
(ModaOPEL = FreqOPEL[FreqOPEL$fi == max(FreqOPEL$fi),"nGuasti" ])
```

```
## [1] 0
```

MEDIANA

Corrisponde al quantile di livello 0.5, quindi il valore che è preceduto da 50%n valori <= mediana

```
(MedianaFIAT = FreqFIAT[FreqFIAT$Pi >=0.5 ,"nGuasti" ][1])
```

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

```
(MedianaOPEL = FreqOPEL[FreqOPEL$Pi >=0.5 ,"nGuasti" ][1])
```

```
## [1] 0
```

MEDIA

```
calcolaMedia = function(valori,frequenze){
  somma =0
    for (i in 1:length(valori)){
      somma = somma + (valori[i]*frequenze[i])
    }
  return (somma)
}

(MediaFIAT = calcolaMedia(FreqFIAT$nGuasti,FreqFIAT$pi))
```

```
## [1] 1.5
```

```
(MediaOPEL = calcolaMedia(FreqOPEL$nGuasti,FreqOPEL$pi))
```

```
## [1] 0.5833333
```

VARIANZA

$$V(Y) = \frac{1}{n} * \sum_{j=1}^{J} (y_j - E(Y))^2 * f_j$$

$$J = |Sy|$$

```
calcolaVarianza = function(valori,frequenze,media){
  somma = 0
  for (i in 1:length(valori)){
    somma = somma + (((valori[i]-media)**2) *frequenze[i])
  }
  return (1/sum(frequenze)*somma)
}

(VarianzaFIAT = calcolaVarianza(FreqFIAT$nGuasti,FreqFIAT$fi,MediaFIAT))
```

```
## [1] 1.4
```

```
(VarianzaOPEL = calcolaVarianza(FreqOPEL$nGuasti,FreqOPEL$fi,MediaOPEL))
```

```
## [1] 0.5430556
```

COEFFICIENTE DI VARIAZIONE

$$CVy = \frac{\sigma_y}{|E(Y)|}$$

$$\sigma_y = \sqrt{V(Y)}$$

```
(sigmaFIAT = VarianzaFIAT**0.5)
```

```
## [1] 1.183216
```

```
(sigmaOPEL = VarianzaOPEL**0.5)
```

```
## [1] 0.736923
```

```
(CV_FIAT = sigmaFIAT / abs(MediaFIAT))
```

```
## [1] 0.7888106
```

```
(CV_OPEL = sigmaOPEL / abs(MediaOPEL))
```

```
## [1] 1.263297
```

A.4)

Stipendio	No. di dipendenti
1000 ⊣ 1100	8
1100 ⊣ 1200	10
1200 ⊣ 1300	16
1300 ⊣ 1400	14
1400 ⊣ 1500	10
1500 ⊣ 1800	5
1800 ⊣ 2500	2
Totale	65

Y = stipendio

QUANTITATIVA CONTINUA

```
(FreqTable = data.frame(Low = c(1000,1100,1200,1300,1400,1500,1800),High= c(1100,1200,1300,1400,1500,180
```

```
##    Low High fi
## 1 1000 1100  8
## 2 1100 1200 10
## 3 1200 1300 16
## 4 1300 1400 14
## 5 1400 1500 10
## 6 1500 1800  5
## 7 1800 2500  2
```

```r
# calcolo frequenza relativa

(FreqTable = FreqTable %>% mutate(pi = fi/sum(fi)))
```

```
##    Low High fi         pi
## 1 1000 1100  8 0.12307692
## 2 1100 1200 10 0.15384615
## 3 1200 1300 16 0.24615385
## 4 1300 1400 14 0.21538462
## 5 1400 1500 10 0.15384615
## 6 1500 1800  5 0.07692308
## 7 1800 2500  2 0.03076923
```

```r
if (sum(FreqTable$pi) == 1){
  print("LA SOMMA DELLE FREQUENZE RELATIVE CORRISPONDE AL 100%")
}
```

```
## [1] "LA SOMMA DELLE FREQUENZE RELATIVE CORRISPONDE AL 100%"
```

```r
(FreqTable = FreqTable %>% mutate(Fi=calcolaFreqCum(fi))    )
```

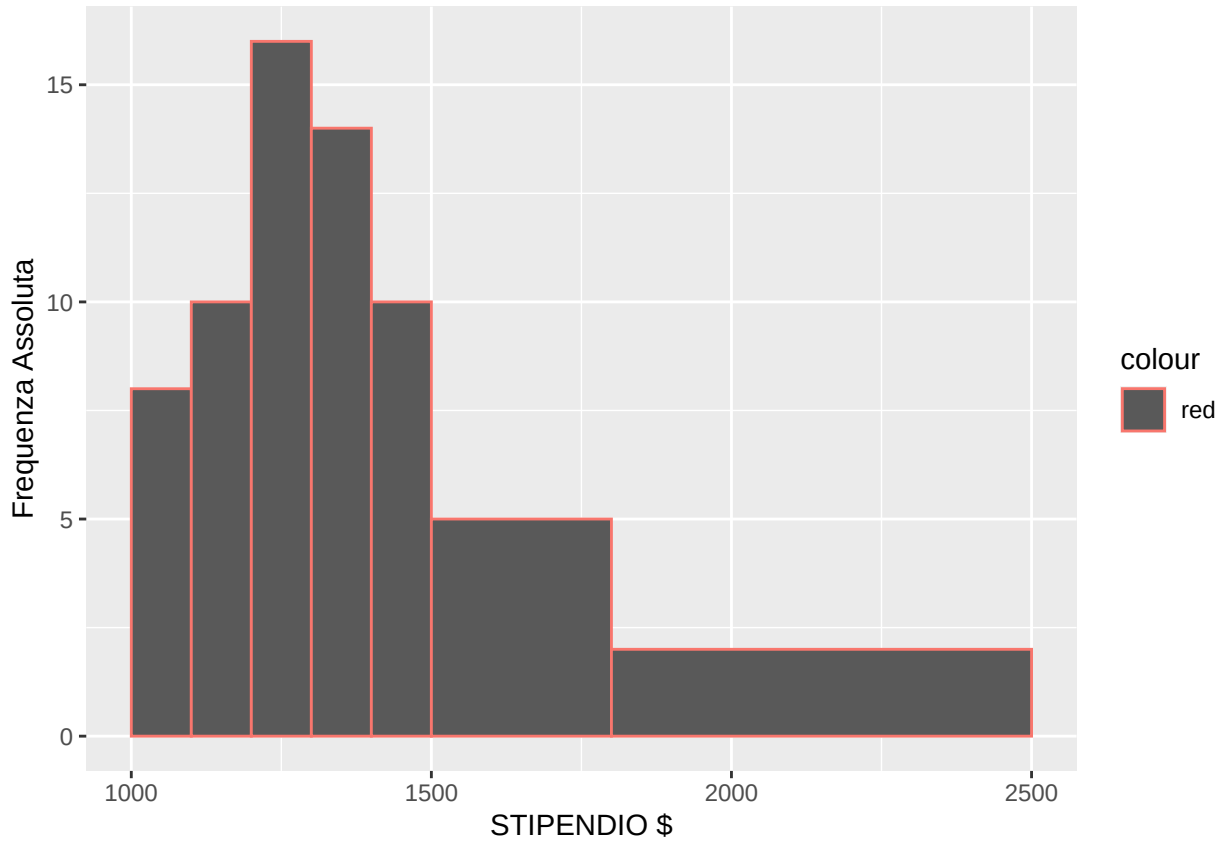
```
##    Low High fi         pi Fi
## 1 1000 1100  8 0.12307692  8
## 2 1100 1200 10 0.15384615 18
## 3 1200 1300 16 0.24615385 34
## 4 1300 1400 14 0.21538462 48
## 5 1400 1500 10 0.15384615 58
## 6 1500 1800  5 0.07692308 63
## 7 1800 2500  2 0.03076923 65
```

```r
(FreqTable = FreqTable %>% mutate(Pi=calcolaFreqCum(pi))    )
```

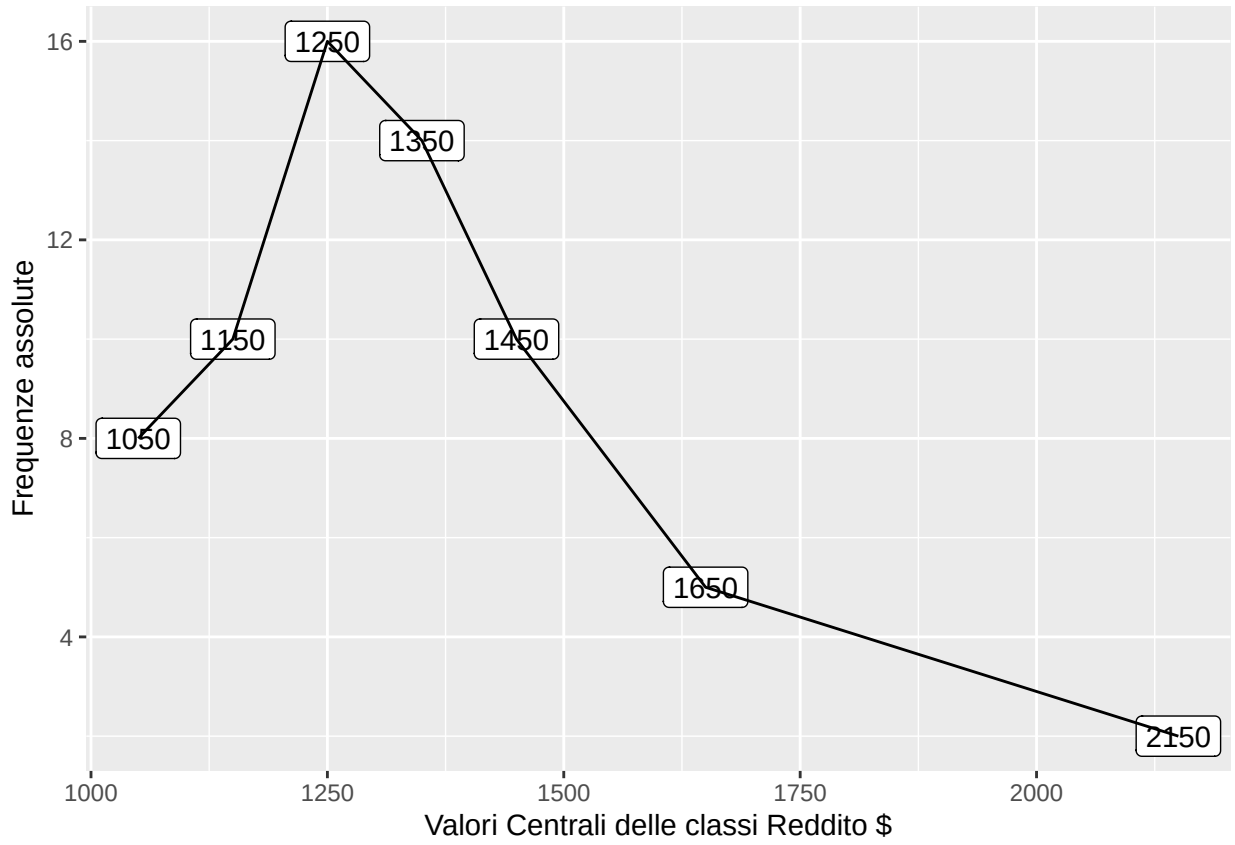
```
##    Low High fi         pi Fi        Pi
## 1 1000 1100  8 0.12307692  8 0.1230769
## 2 1100 1200 10 0.15384615 18 0.2769231
## 3 1200 1300 16 0.24615385 34 0.5230769
## 4 1300 1400 14 0.21538462 48 0.7384615
## 5 1400 1500 10 0.15384615 58 0.8923077
## 6 1500 1800  5 0.07692308 63 0.9692308
## 7 1800 2500  2 0.03076923 65 1.0000000
```

```r
ggplot(FreqTable) + geom_col(aes((High+Low)/2,fi,width =(High-Low),color="red")) + labs(x="STIPENDIO $"
```

```
## Warning: Ignoring unknown aesthetics: width
```




```
ggplot(FreqTable,aes((High+Low)/2,fi)) + geom_label(aes(label=(High+Low)/2  ))+geom_line()  + labs(x="Va
```



MEDIANA

Avendo come riferimento delle classi si va a cercare quale classe è preceduta dal 50% delle osservazioni <= classe stessa

```
(ClasseMediana = FreqTable[FreqTable$Pi >= 0.5,c("Low","High")][1,])
```

```
##    Low High
## 3 1200 1300
```

MEDIA

Avendo a disposizione le classi di valori è opportuno trovare dei valori intermedi che permettano il calcolo della MEDIA

$$E(Y) = \frac{1}{n}\sum_{j=1}^{J} y_j^c f_j$$

$$y_i^c = (y_{i-1} + y_i)/2$$

```
(Yc = (FreqTable$Low+FreqTable$High)/2)
```

```
## [1] 1050 1150 1250 1350 1450 1650 2150
```

```
(MediaClassi = calcolaMedia(Yc,FreqTable$pi))
```

```
## [1] 1320.769
```

VARIANZA

$$[y_{j-1} \dashv y_j], j \in [1, J]$$

$$y_j^c = \frac{(y_{j-1} + y_j)}{2}, j \in [1, J]$$

punto centrale per le singole classi di valori

$$V(Y) = \frac{1}{n} \sum_{j=1}^{J} (y_j^c - E(Y))^2 * f_j$$

```
(VarianzaClassi= calcolaVarianza(Yc,FreqTable$fi,MediaClassi))
```

```
## [1] 46991.72
```