3. GRAFICI PER DISTRIBUZIONI UNIVARIATE

TECNICHE DI ANALISI DI DATI I

"There is no statistical tool that is as powerful as a well chosen graph"

Chambers, Cleveland, Kleiner & Tuckey (1983)

"The greatest value of a picture is when it forces us to notice what we never expected to see"
Tukey(Exploratory data analysis, 1977)

"Excellence in statistical graphics consists of complex ideas communicated with clarity, precision and efficiency"
Tofte, 1983

In queste slide usiamo il dataframe ***attaccamento****: scaricatelo , leggetene la descrizione, aprite il dataframe in R, e,* ***prima di procedere oltre, fate il seguente esercizio:***

1. *Descrivete la struttura del dataframe*
2. *Descrivete il campione: quanti soggetti? Quanti hanno l'assistito in casa e quanti in RSA? Come sono distribuite le caratteristiche socio-anagrafiche?*
3. *Le sottoscale del CBI compongono un punteggio* ***totale****: createlo nel dataframe (chiamate la variabile $CBI_totale).*
4. *Anche le sottoscale del WHOQOL possono creare una dimensione complessiva: in questo caso, è data dalla* ***media*** *della qualità della vita nei diversi ambiti. Create la variabile (chiamatela $WHOQOL_media).*
6. *Considerate solo il sottogruppo con l'assistito in casa: quanti usufruiscono di un centro diurno? Cosa potete rilevare rispetto all'aiuto ricevuto? Quali considerazioni si potrebbero fare (e come) rispetto all'avere un aiuto e al burden totale?*

Grafici per distribuzioni univariate

Abbiamo usato **modelli numerici** per descrivere caratteristiche di **distribuzioni univariate**; ora useremo per lo stesso scopo dei **grafici:** un **esame grafico preliminare** della presentazione dei dati è **indispensabile**.

Obiettivo primario è **comunicare**, a se stessi e agli altri, cosa è successo .

I **vantaggi dei grafici** (Schmid, 1954):

1. grafici ben fatti sono più efficaci nel **creare interesse e** attrarre l'attenzione;
2. le **relazioni visive** rappresentate dai grafici sono comprese più **facilmente**;
3. l'uso dei grafici fa **risparmiare tempo**, dato che il significato essenziale di ampie raccolte di dati può essere compreso con uno sguardo;
4. i grafici e diagrammi offrono una raffigurazione più completa di un problema, rispetto a quella che potrebbe derivare da presentazioni tabulari o testuali dei dati;
5. i grafici aiutano a **far emergere realtà nascoste** e **relazioni**, stimolano e aiutano il pensiero analitico e l'investigazione.

Fare grafici con R

R adora i grafici. Tra i packages di base, `lattice` e `MASS` hanno funzioni per quasi tutti i grafici che faremo. Altri packages offrono soluzioni raffinate: il top è `ggplot2`, decisamente complesso. Man mano, comunque, vedremo altre possibilità grafiche per specifiche analisi.

R usa tre diversi **tipi di funzioni** per produrre grafici:

di alto livello

Creano un grafico: `plot`, `boxplot`, `histogram`, `pie`, `barplot`. **La funzione più generale è `plot`**, sensibile alla classe della distribuzione cui si applica.

di basso livello

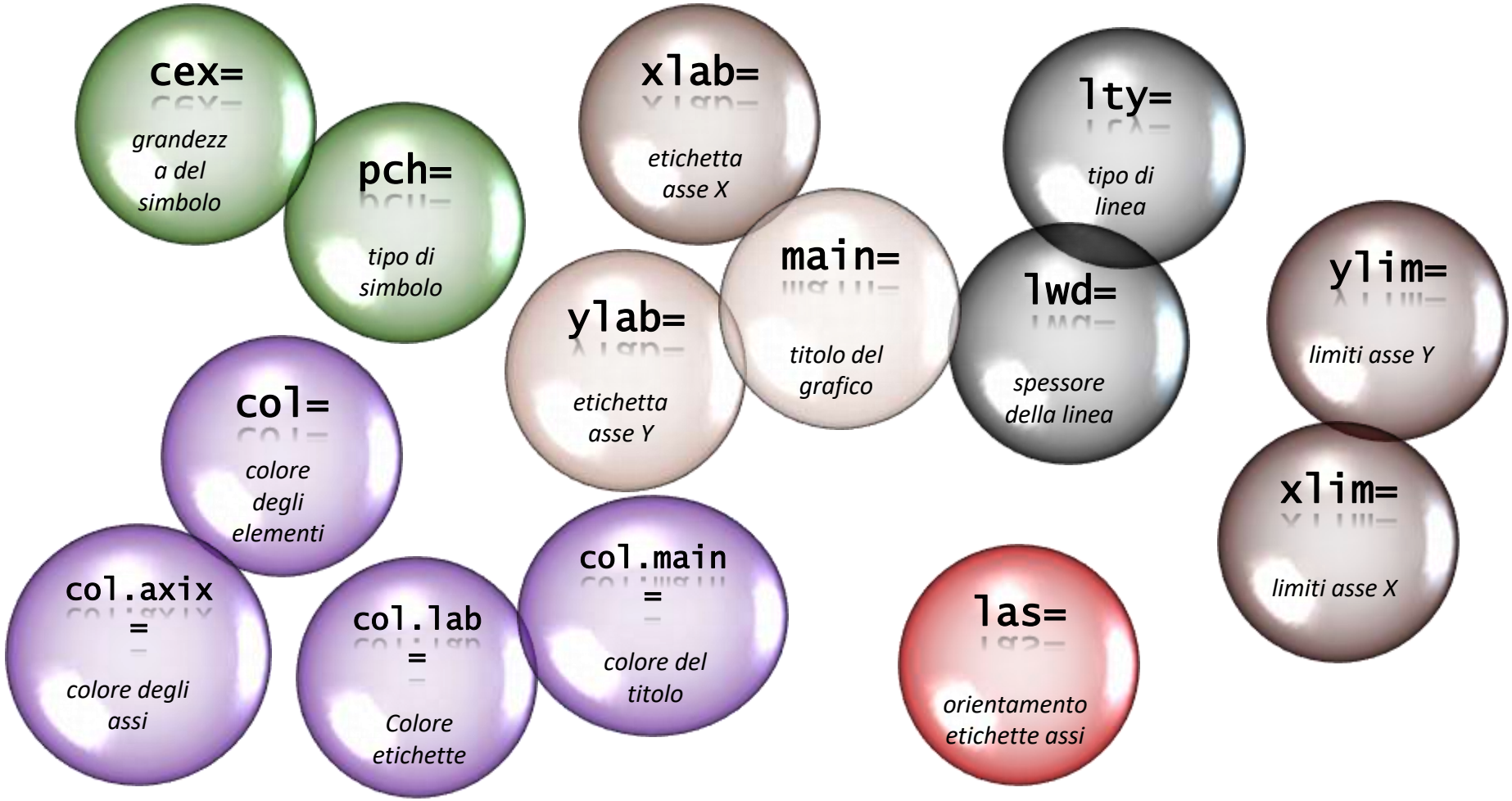
Aggiungono parti a un grafico esistente: per esempio, `abline` sovrappone una linea secondo le coordinate indicate nella funzione

interattivo

Aggiungono o estraggono interattivamente informazioni da un grafico; per esempio, `identify` identifica in un `plot` numeri corrispondenti alla riga del soggetto nel dataframe, `scatter3d` muove grafici di regressione multipla nel piano.

Parametri

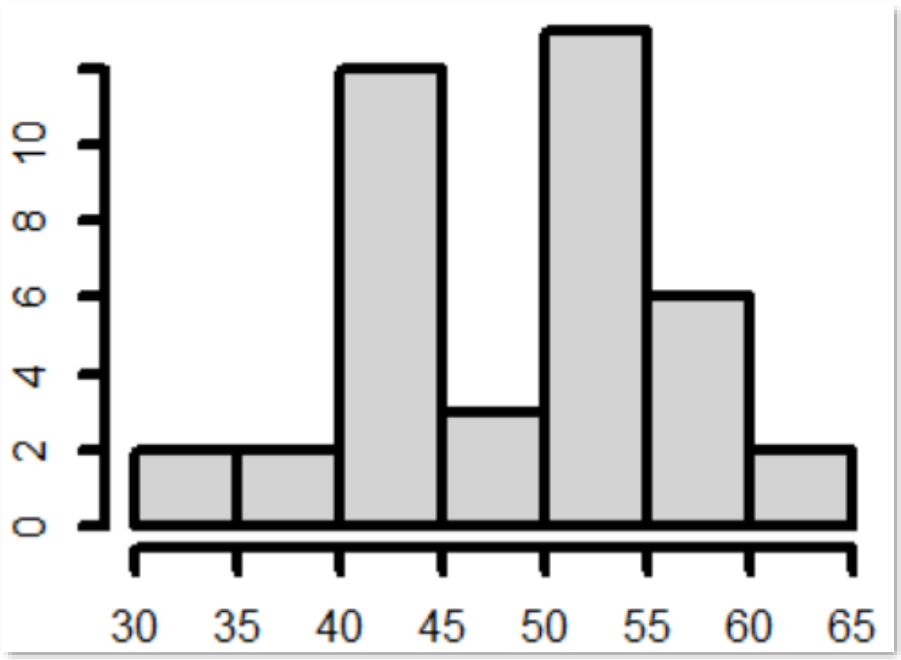
Può essere personalizzata una gran quantità di parametri grafici; per l'elenco completo, chiedete `help(par)`, per la descrizione di quelli più frequenti guardate nella dispensa.



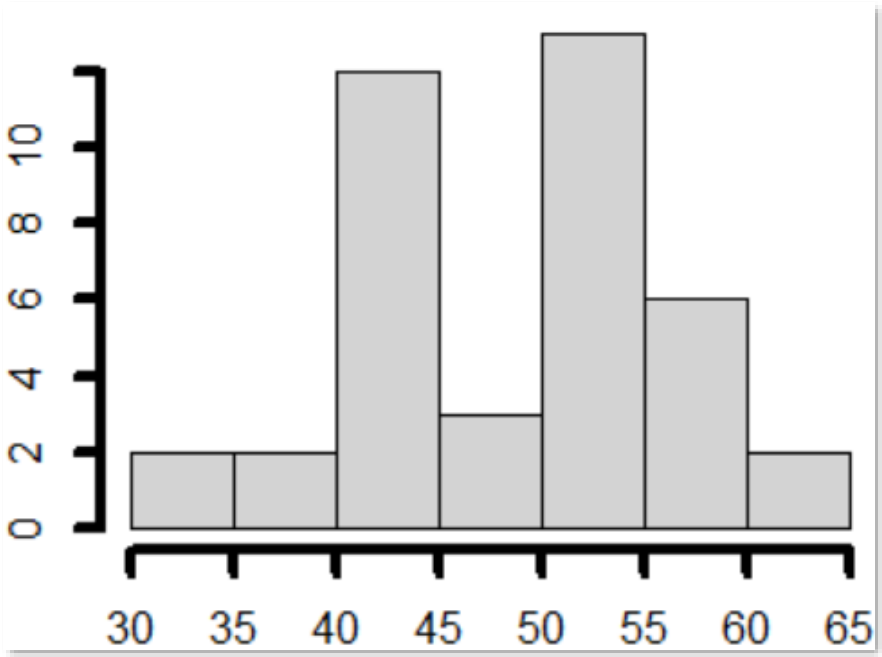
Alcuni parametri possono essere impostati come argomenti di `par` **prima** di lanciare il comando del grafico, ma molti possono essere anche inseriti **come argomenti** della funzione che crea il grafico (ad esempio, `plot`) o essere indicati **dopo** aver creato il grafico (per esempio, `abline`).

In molti casi il loro effetto sul grafico è lo stesso, in altri no:

```
par(lwd=4)
hist(attaccamento$eta, lwd=4)
```




```
hist(attaccamento$eta, lwd=4)
```



Grafici

per variabili numeriche o

scatterplot

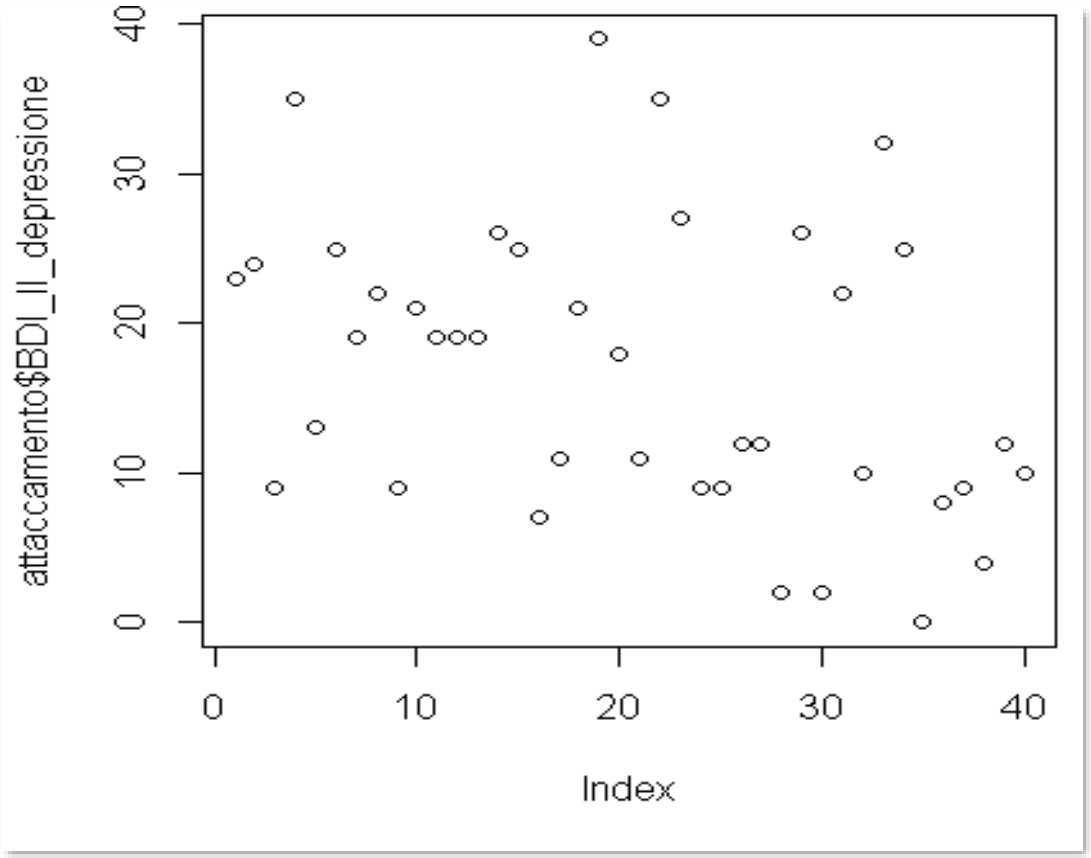
Quando la variabile da rappresentare è `numeric`, `plot(distribuzione)` crea un **grafico sequenziale**, in cui sono rappresentate le coordinate in X e in Y di ogni soggetto. **Con una sola distribuzione**, in X (Index) sono elencati i casi, in Y i valori della variabile di ogni caso.

Per conoscere la **distribuzione dei punteggi di depressione soggetto per soggetto**:

```
plot(a$BDI_II_depressione)
```

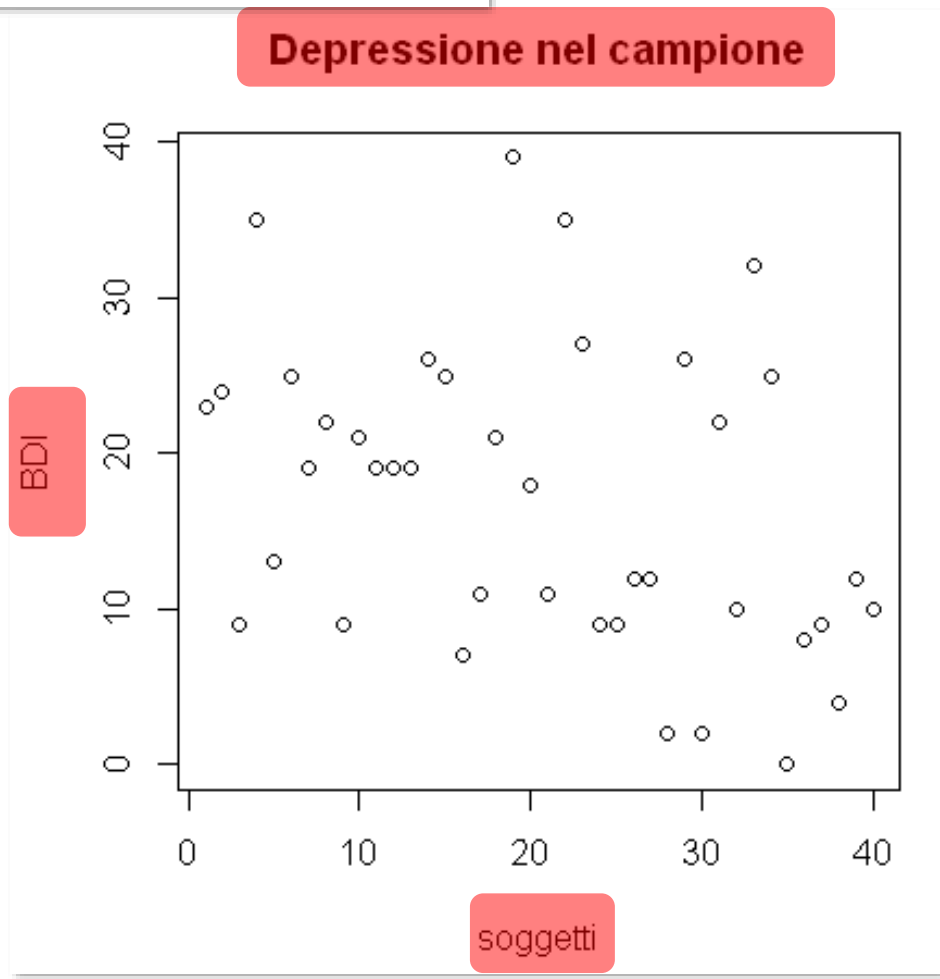
In X i 40 soggetti, in Y i punteggi al BDI: ogni pallino **rappresenta il punteggio al test** di un soggetto.

Di default, X è stato etichettato Index, Y ha ricevuto il nome della variabile.



Personalizziamo i **titoli degli assi**: `xlab="testo"` e `ylab="testo"` per X e Y; diamo un **titolo all'intero grafico**: `main="testo"`:

```
plot(a$BDI_II_depressione,  xlab="soggetti",  ylab="BDI",
   main="Depressione nel campione")
```



È un po' triste. Potremmo: **ingrandire i simboli**: `cex=` **da 1** in su; **cambiare i simboli**: `pch=` **da 0 a 25** (in dispensa sono elencati i 25 simboli); **colorarli**: `col="colore"`.

I colori si indicano per **nome** ("`red`", "`light blue`", "`purple`", ecc,) o per codice esadecimale **RGB**: digitate `demo(colors)` per conoscere tutti i colori disponibili.

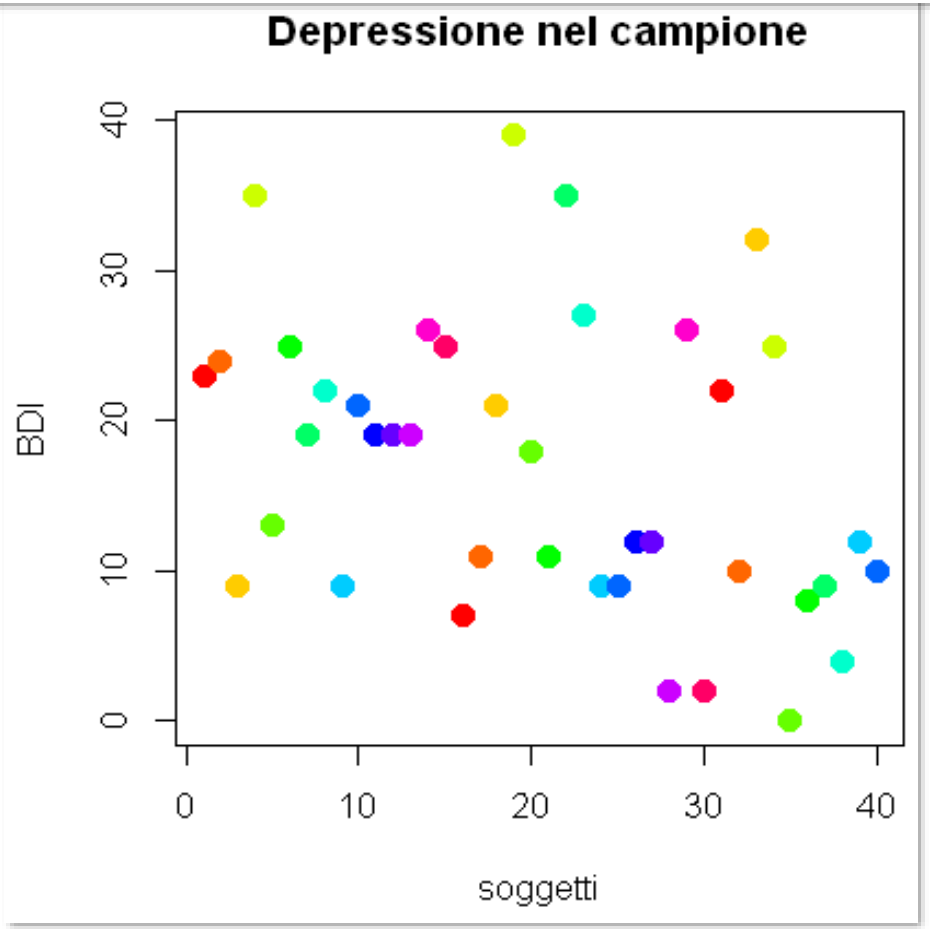
Per una coloritura rapida, usate `col=rainbow(numero di sfumature)`.

Aggiungiamo colore al plot `col=rainbow(15)`, ingrandiamone i simboli `cex=1.5` e cambiamo il simbolo da cerchio vuoto a cerchio **pieno**: `pch= 19`

```
plot(a$BDI_II_depressione, xlab = "soggetti", ylab="BDI", main="Depressione nel campione",
   cex=1.5, pch=19, col=rainbow(15))
```

I punteggi 20-29 indicano depressione moderata, quelli >30 grave depressione: per evidenziare questi cut off, **aggiungiamo due linee** con `abline`: linea **blu continua** per la depressione moderata, linea **rossa tratteggiata** per la depressione grave.

Dobbiamo quindi specificare coordinate, tipo e colore delle linee.

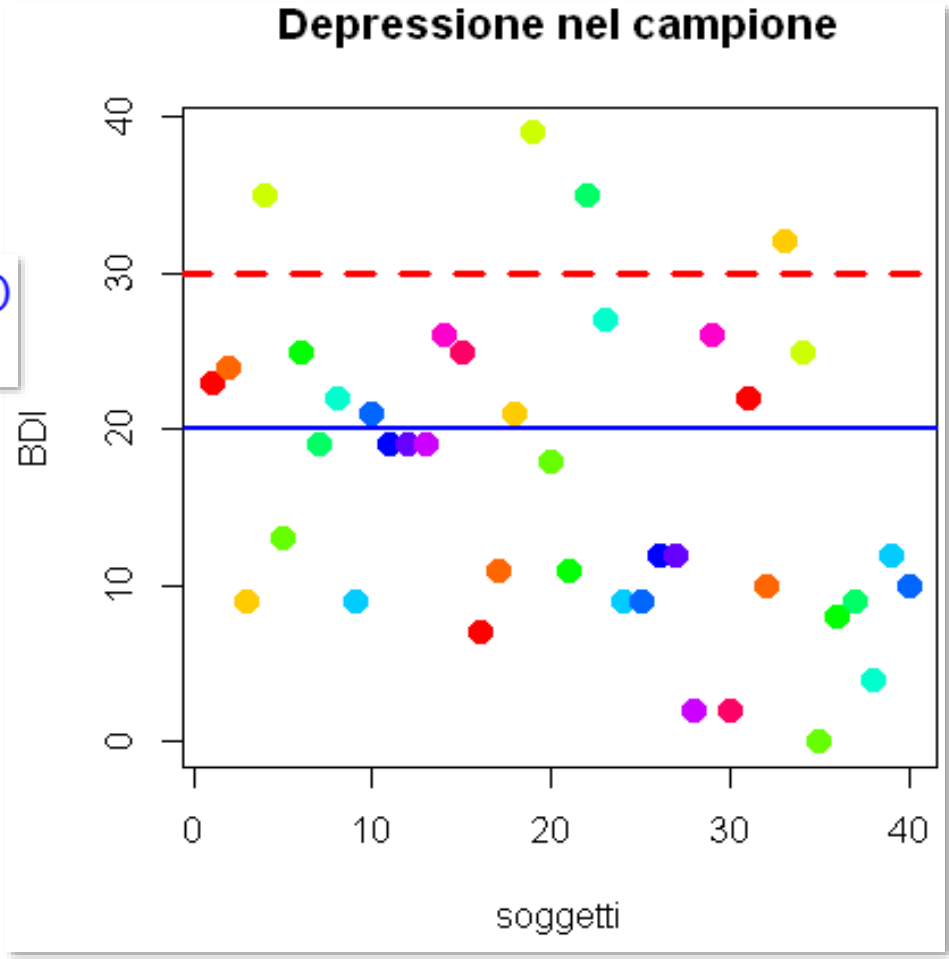


In `abline` inseriamo le coordinate: `h=valore in Y` per linee orizzontali, o `v= valore in X` per linee verticali. `lty= valore` indica il tipo di linea: `1` (default) continua, `2` tratteggiata, `3` punteggiata, `4` tratti e punti, `5` tratti lunghi, `6` tratti doppi. `lwd= valore` indica lo spessore della linea (1 di default).

Dopo aver creato il plot con la funzione precedente, aggiungiamo:

```
abline(h = 20, col="blue", lty=1, lwd=2.5)
abline(h=30, col="red", lty=2, lwd=3)
```

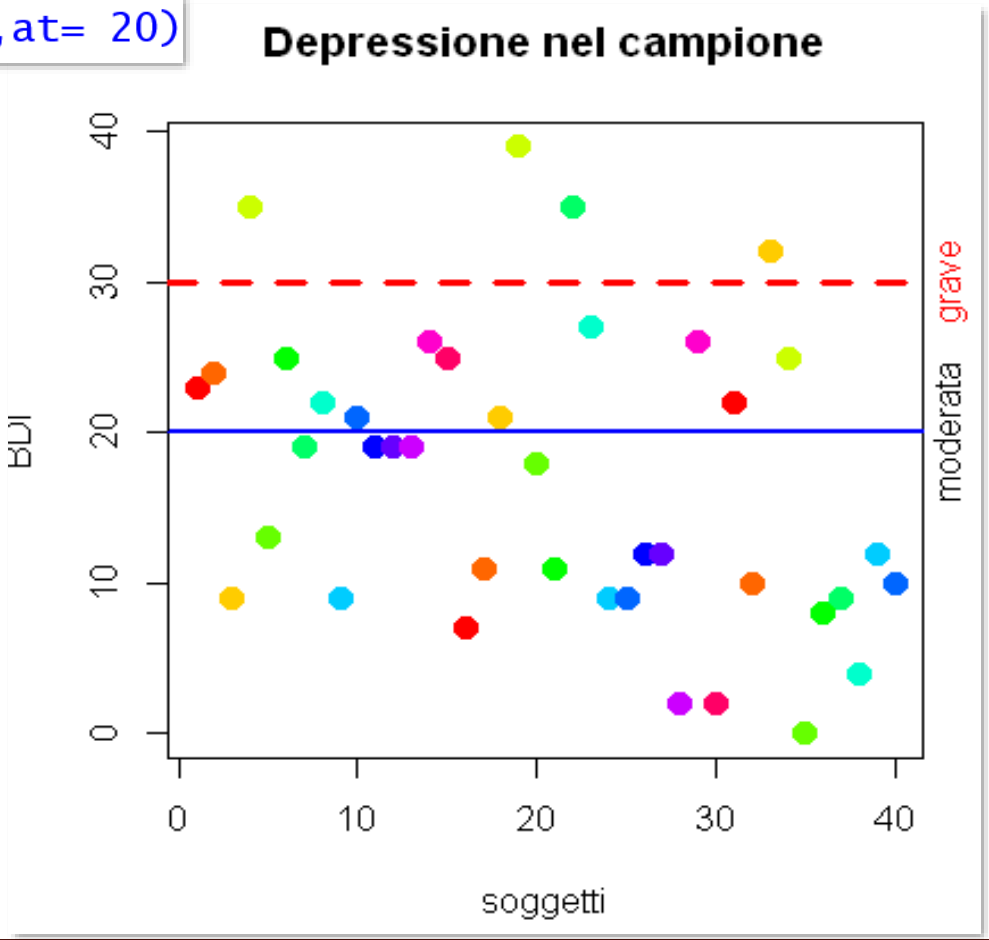
Si può **aggiungere del testo**; per aggiungerlo **all'esterno** usiamo `mtext,` i cui argomenti – base sono: `text="cose da scrivere"`, `side= valore` che indica in quale margine scriverle, `at= valore` della coordinata in cui inserire il testo



Scriviamo “**moderata**” in corrispondenza del margine **destro**, altezza in $Y = 20$, e “**grave**” in corrispondenza del margine **destro**, altezza in $Y = 30$, in **rosso**. I valori dei margini in `side=` sono **1-basso, 2-sinistra, 3-alto, 4-destra**.

Dopo aver creato il plot e inserito le linee con le funzioni precedenti, aggiungiamo:

```
mtext(text = "moderata", side = 4,at= 20)
mtext(text = "grave", side=4,
  at=30, col = "red")
```



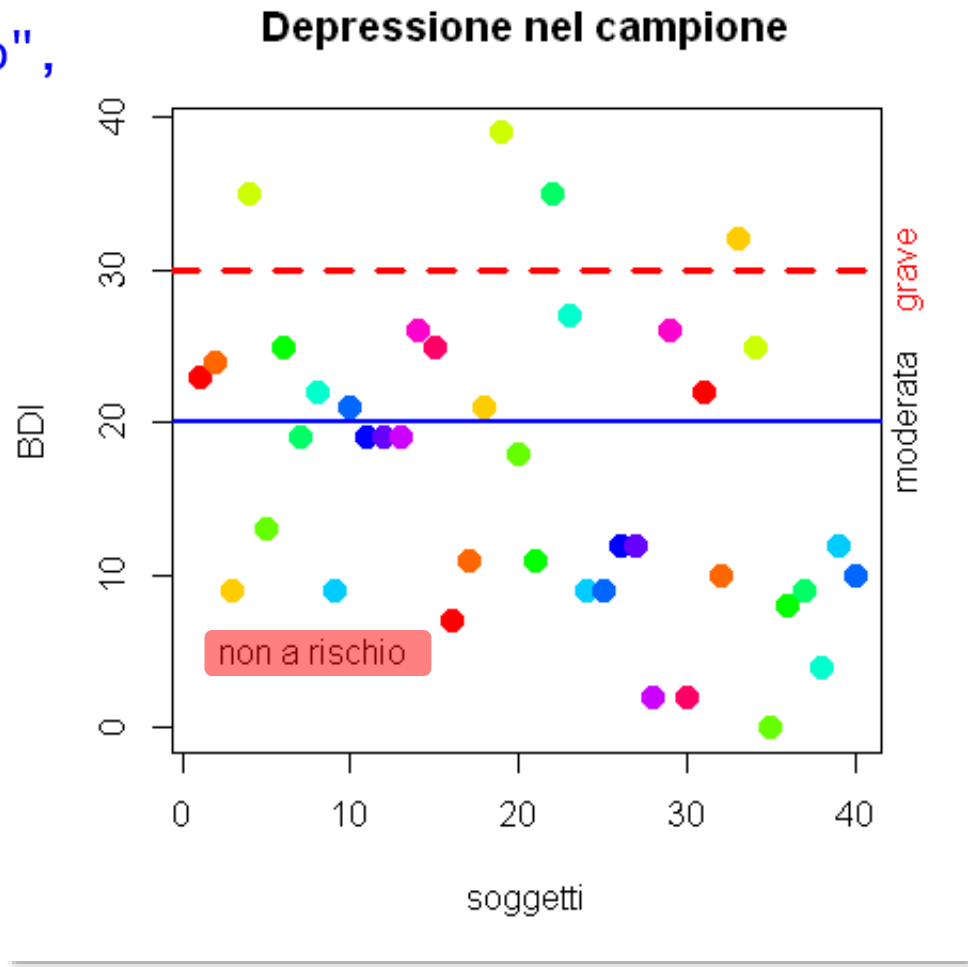
Per aggiungere testo **all’interno** usiamo `text,` i cui argomenti – base sono: `x= valore in X` e `y=valore in Y` come coordinate **da cui** scrivere il testo, `pos= valore` per la posizione in cui scrivere il testo rispetto alle coordinate (1 sotto, 2 a sinistra, 3 sopra, 4 a destra).

Scriviamo **“non a rischio”** nel settore **inferiore alla linea blu**; lo spazio libero è in fondo a sinistra, quindi indicheremo **come coordinate**, da cui **tracciare verso destra** il testo, il valore **1 in X e 5 in Y**.

Dopo aver fatto tutte le cose precendenti, digi

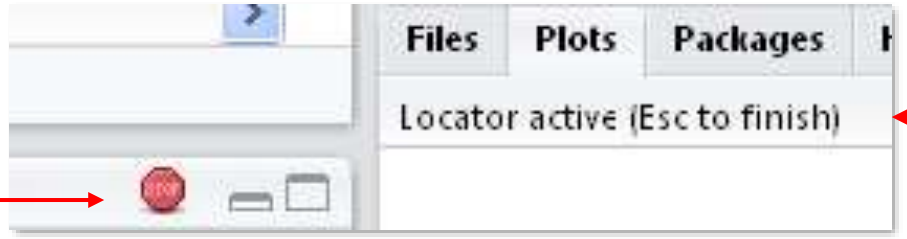
```
text(x=1,y=5,labels="non a rischio",
  pos= 4)
```

Aggiungiamo un **elemento interattivo**: con `identify(variabile in X)` **identifichiamo i soggetti gravemente depressi**. Dopo aver digitato `identify()` e dato **Invio**, clicchiamo con il mouse sui quattro punti sopra la linea rossa.



Dopo aver scritto `identify(attaccamento$BDI_II_depressione:`

In Rstudio compare questo simbolo, che indica che R sta lavorando



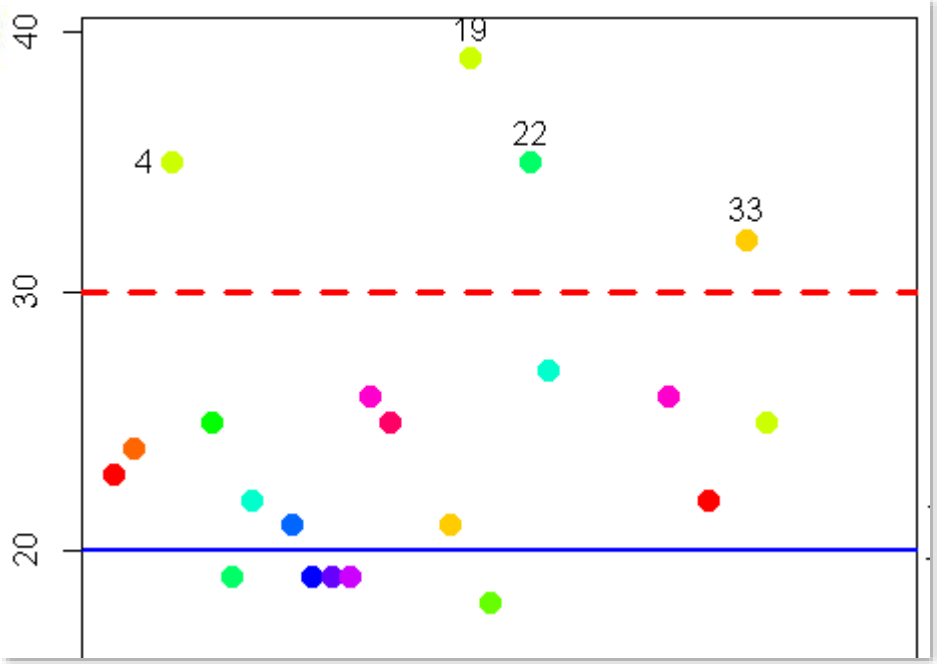
... e che è in attesa di input,

In R tradizionale non appaiono simboli, ma in entrambi i casi si prosegue **cliccando con il mouse su tutti i punti** da identificare, **poi si preme Esc** per terminare.

```
identify(attaccamento$BDI_II_depressione)
[1] 4   19   22   33
```

Sul grafico e in console (non in R) sono riportati numeri di riga dei casi:

Quale altro modo conoscete per identificare questi casi senza usare un grafico?



Riassumendo, lo script dell'ultimo grafico è:

```
plot(attaccamento$BDI_II_depressione, xlab =
  "soggetti", ylab="BDI", main="Depressione nel
  campione",cex=1.5, pch=19, col=rainbow(15))

abline(h = 20, col="blue",lty=1, lwd=2.5);
  abline(h=30, col="red", lty=2, lwd=3)

mtext(text = "moderata",side = 4,at= 20);
  mtext(text = "grave", side=4, at=30, col =
  "red")

text(x= 1, y=5, labels = "non a rischio", pos=4)

identify(attaccamento$BDI_II_depressione)
```

Grafici

per distribuzioni di densità di frequenza e di frequenza

Per rappresentare la distribuzione della **densità di frequenza** assoluta di variabili **continue suddivise in classi** (***bin***), si possono usare gli **istogrammi**, in genere usati per contare frequenze e mostrare la distribuzione di una variabile.

Usiamo `hist(variabile continua)`: in X è rappresentata la **misura**, divisa in classi, in Y la sua **densità di frequenza**, indicata dall'argomento `freq=FALSE`; alternativamente, in Y possiamo avere la **frequenza assoluta** (di default: `freq=TRUE`).

Se il numero di classi non è indicato nell'argomento `breaks= numero`, R le stima usando la regola di Sturges (di default: `breaks="Sturges"`):

$$\textbf{Numero di classi: } K = 1 + 3.322 \times log_{10}(N)$$

Non è un metodo ottimale se $N < 30$ o se la distribuzione è asimmetrica: è quindi opportuno costruire più istogrammi con diverso K, per valutare quale dia la rappresentazione migliore, modificando l'argomento `breaks=`.

Notate che R **accetta solo suggerimenti**, modificando il K indicato se non lo ritiene adeguato.

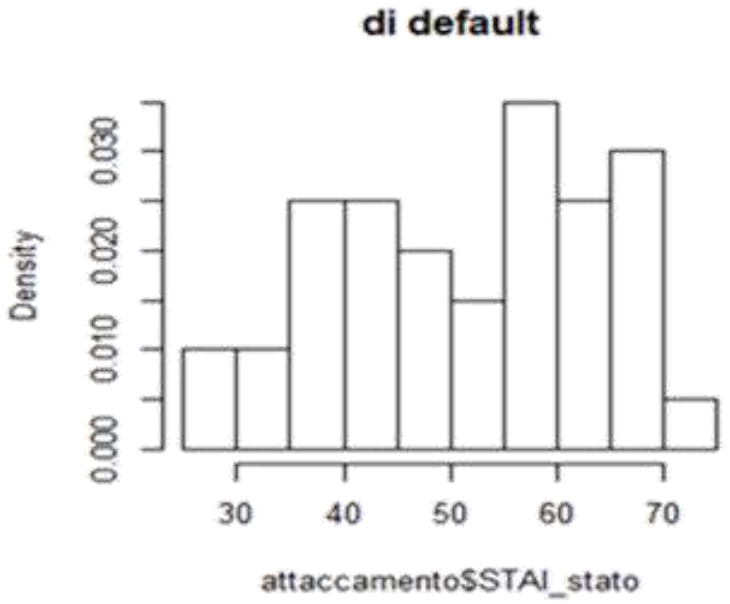
Una regola pratica per identificare il numero ideale di classi è **arrotondare all'intero più vicino la radice quadrata del numero di osservazioni**: nel dataframe `attaccamento` abbiamo 40 soggetti e la radice quadrata di 40 è 6.32, quindi "**sei classi" è un buon suggerimento**

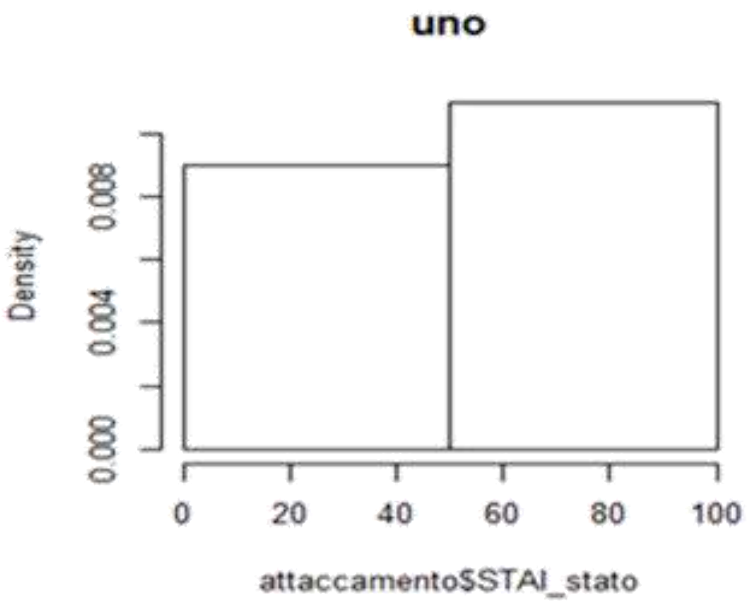
Rappresentiamo la distribuzione di **densità di frequenza dell'ansia di stato**, impostando **quattro diversi k**: quello di default , $k = 1$, $\boldsymbol{k = 6}$ e $k = 13$.

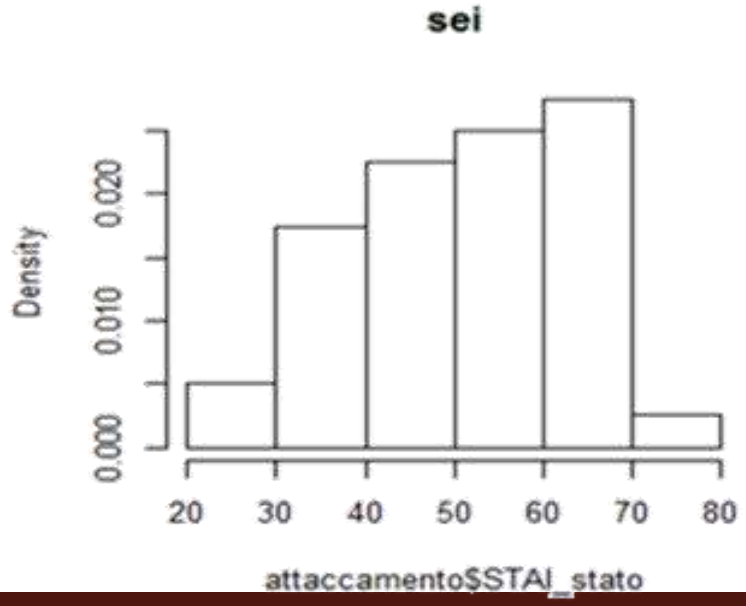
Ottimizziamo la presentazione dei grafici con un `par`: **`mfrow(c(righe,colonne)`** ripartisce la finestra dei grafici secondo il **numero di righe** e **colonne** specificato negli argomenti. Dato che predispone l'ambiente su cui saranno stampati i grafici, deve essere impostato **prima** di eseguire i grafici stessi.

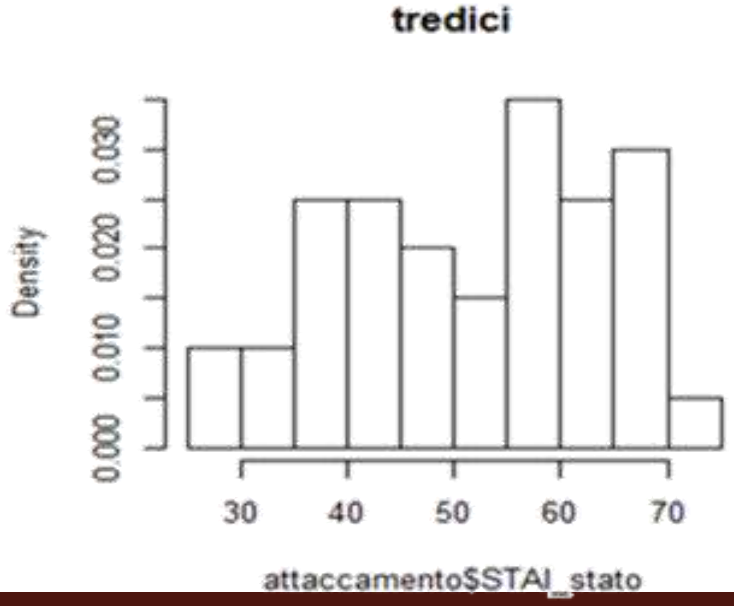
```
par(mfrow = c(2,2))
```

```
hist(attaccamento$STAI_stato,freq=FALSE, main="di default")
hist(attaccamento$STAI_stato,freq=FALSE, main="uno", breaks = 1)
hist(attaccamento$STAI_stato,freq=FALSE, main="sei", breaks = 6)
hist(attaccamento$STAI_stato,freq=FALSE, main="tredici", breaks = 13)
```


di default
Density
0.000
0.010
0.020
0.030
30
40
50
60
70
attaccamento$STAI_stato

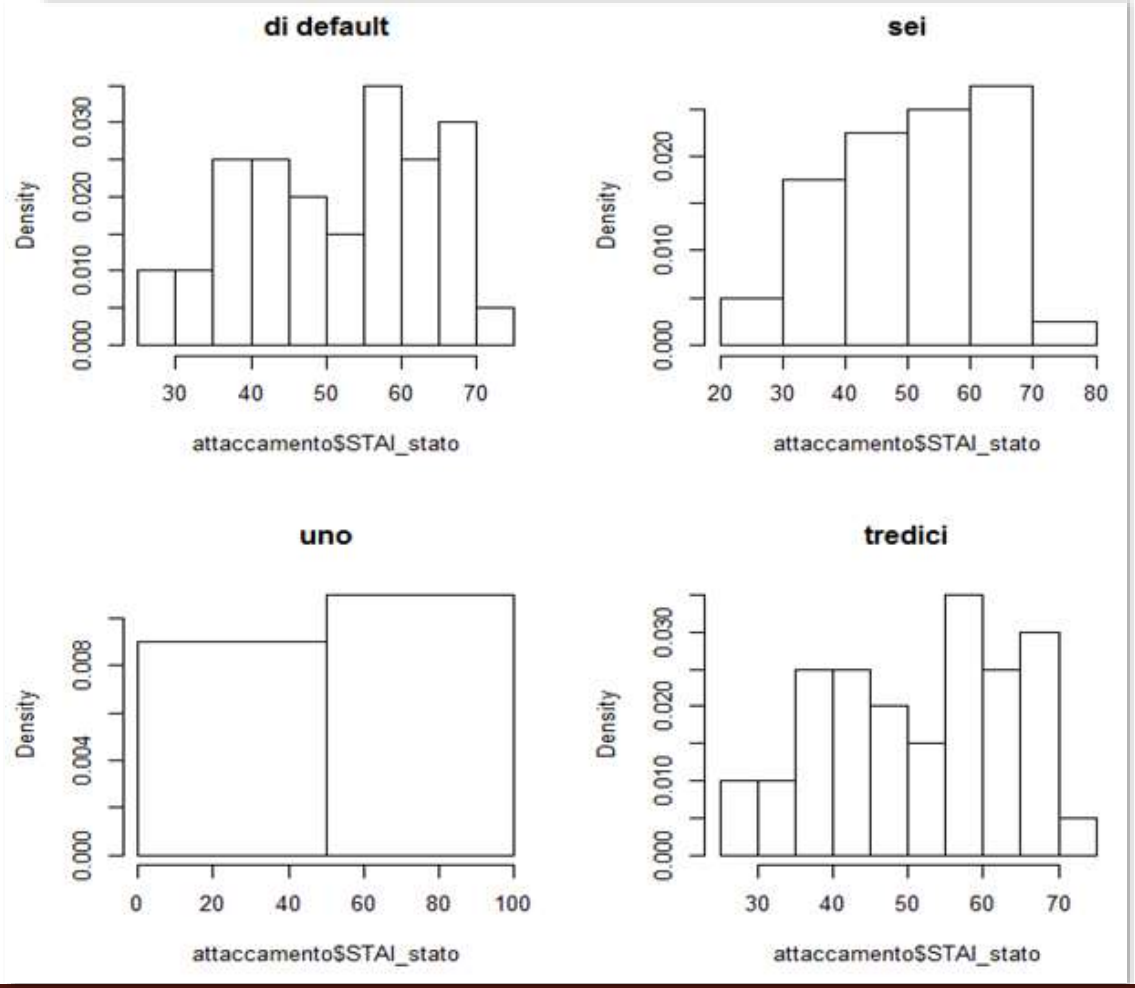

uno
Density
0.000
0.004
0.008
0
20
40
60
80
100
attaccamento$STAI_stato


sei
Density
0.000
0.010
0.020
20
30
40
50
60
70
80
attaccamento$STAI_stato


tredici
Density
0.000
0.010
0.020
0.030
30
40
50
60
70
attaccamento$STAI_stato

Il parametro **par(mfcol=numero righe, numero colonne)** predispone il layout della finestra dei grafici ordinandoli per colonna:

```
hist(attaccamento$STAI_stato,freq=FALSE, main="di default")
hist(attaccamento$STAI_stato,freq=FALSE, main="uno", breaks = 1)
hist(attaccamento$STAI_stato,freq=FALSE, main="sei", breaks = 6)
hist(attaccamento$STAI_stato,freq=FALSE, main="tredici", breaks = 13)
```



La finestra resta ripartita secondo le indicazioni di par(mfrow) o par(mfcol); per tornare a visualizzare un solo grafico per finestra, richiedetelo:

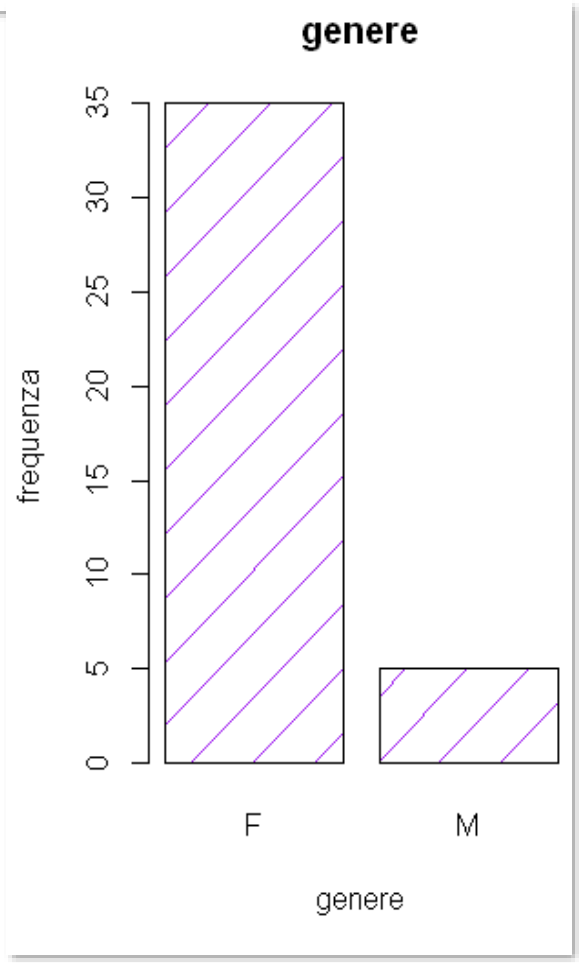
```
par(mfrow = c(1,1))
```

Per rappresentare la distribuzione della **frequenza** assoluta di variabili **categoriali** si può usare `barplot(table(distribuzione))`. Vediamo quella del **genere:**

```
barplot(table(a$genere), main="genere", xlab="genere", ylab="frequenza",
  col= "purple", density= 3)
```

`density=valore` riempie i rettangoli con linee diagonali più o meno fitte (da 1 in su), colorate con `col="colore"`

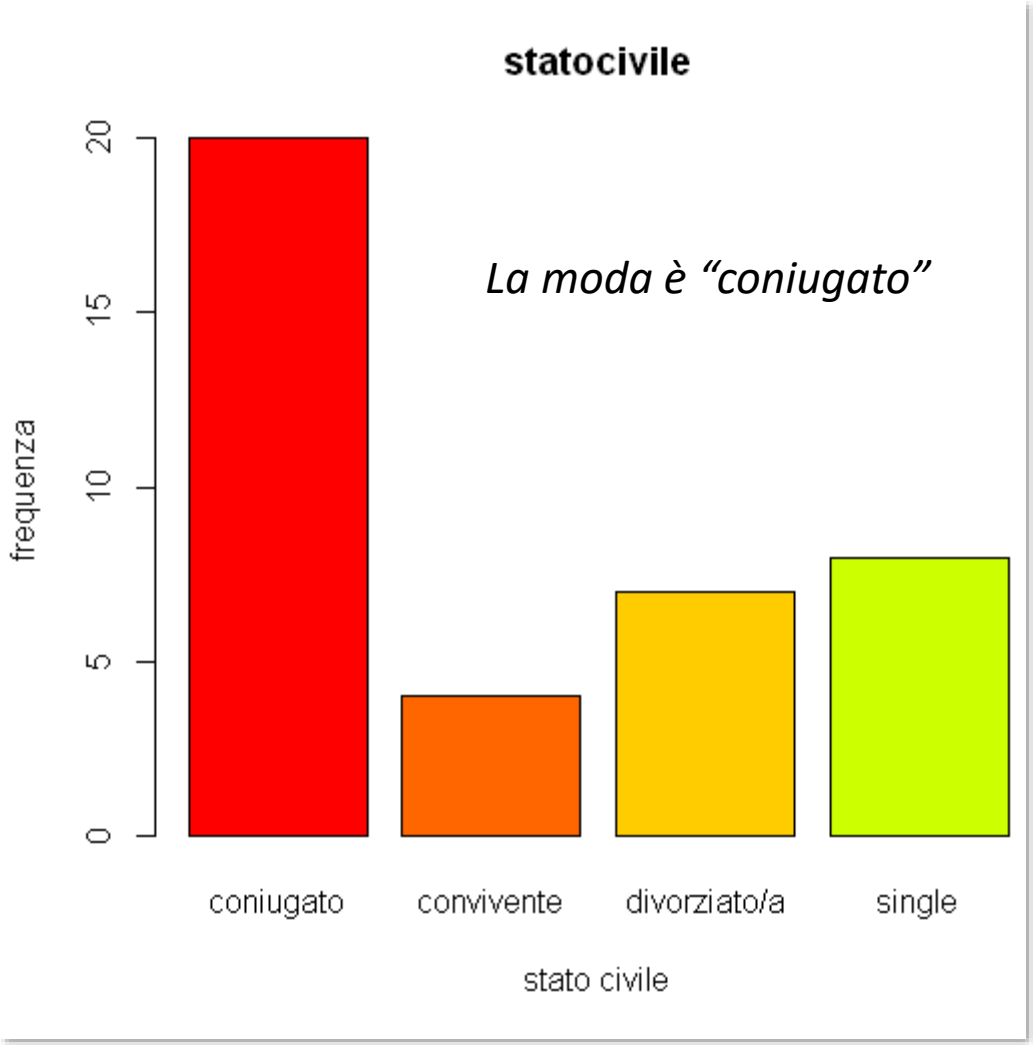
Dal grafico, cosa potete inferire della relazione tra caregiving e genere?



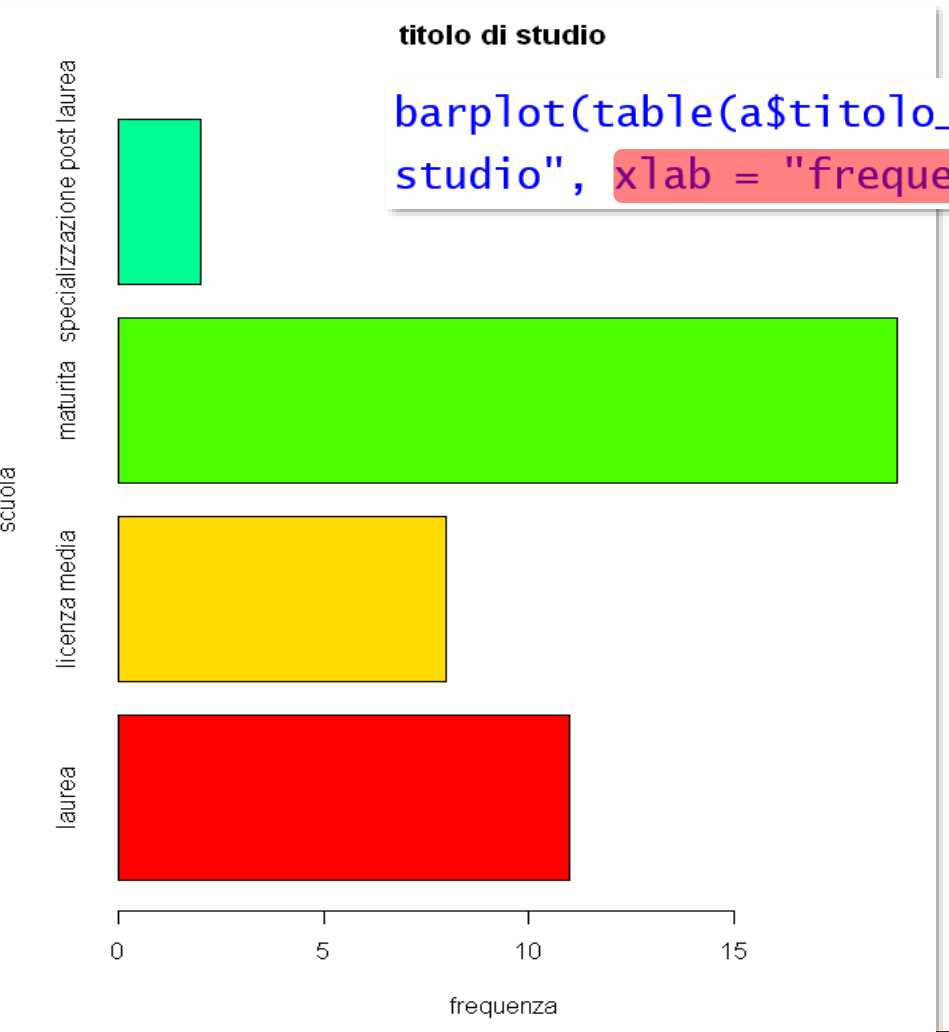
Vediamo la distribuzione dello **stato civile**:

```
barplot(table(a$stato_civile), main= "statocivile", xlab= "stato civile",
  ylab= "frequenza", col= rainbow(15))
```

Le barre sono separate, perché rappresentano categorie **discrete** (di default, l'argomento è `beside= FALSE`)



Se le **barre** sono disposte **orizzontalmente** invece che verticalmente, il grafico si definisce **a pila**: si imposta l'argomento logico **`horiz=TRUE.`**




```
barplot(table(a$titolo_studio),horiz = TRUE, main="titolo di
studio", xlab = "frequenza", ylab = "scuola", col=rainbow(7))
```

Attenti alle etichette di X e Y!

Il titolo di studio prevalente è piuttosto alto: le coorti cambiano...

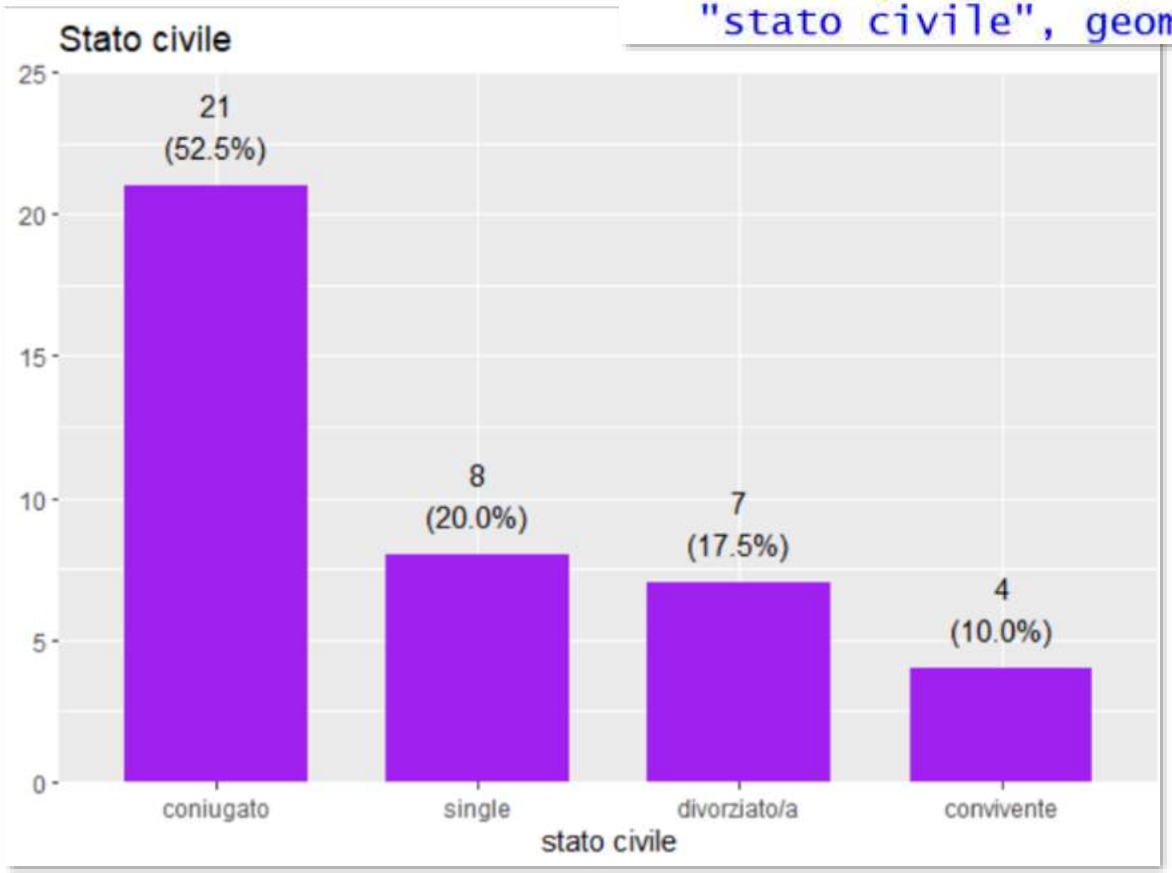
`plot_frq(variabile, type="bar")` di `sjPlot` produce un bel barplot, le cui barre possono essere ordinate per frequenza, in senso ascendente o discendente: `sort.frq = "desc"` o `"asc"`.

```
plot_frq(a$stato_civile, title = "Stato civile",
   sort.frq = "desc", type = "bar", axis.title =
   "stato civile", geom.colors = "purple")
```

titolo del grafico

colore

nome dell'asse X



Istogramma condizionale o co-plot

`histogram(~variabile | fattore)` di `lattice` (installato con R, va caricato nel workspace) è utile per **rappresentare le diverse distribuzioni di frequenza nei livelli di un fattore**. La **tilde ~** indica a R che vogliamo "**una cosa in funzione di un'altra**". Compone la **formula** del tipo $Y \sim X$ (Y **in funzione di** X), con cui lavoreremo moltissimo. Non è un carattere di tastiera, ma può essere richiamata **come carattere ASCII:**

- ✓ Se usate un dispositivo con tastierino numerico, in **Windows** si ottiene con la combinazione di tasti **Alt + 126**, in **MacOS con Alt+5**.

- ✓ Si può **inserire come simbolo in un editor di testi**, copiarlo e incollarlo in uno script di R per copincollarlo quando serve.

- ✓ Potete creare l'oggetto tilde e copiare e incollare il suo output nelle formule:

```
(tilde<-rawToChar(as.raw(126)))
[1] "~"
```

Vediamo la distribuzione di frequenza dello **stress derivante dalla fatica fisica** nei **caregiver che assistono il paziente a casa** rispetto a quella dei **caregiver con pazienti ricoverato**:

```
histogram(~a$CBI_burden_fisico | a$domicilio_assistito, xlab="domicilio assistito",
  ylab="carico fisico", col=rainbow(15))
```

Chi ha il paziente in RSA mostra prevalentemente bassi punteggi di burden fisico (asimmetria sinistra), ma tra chi assiste il paziente in casa ci sono pochissimi punteggi bassi e una prevalenza di punteggi alti.

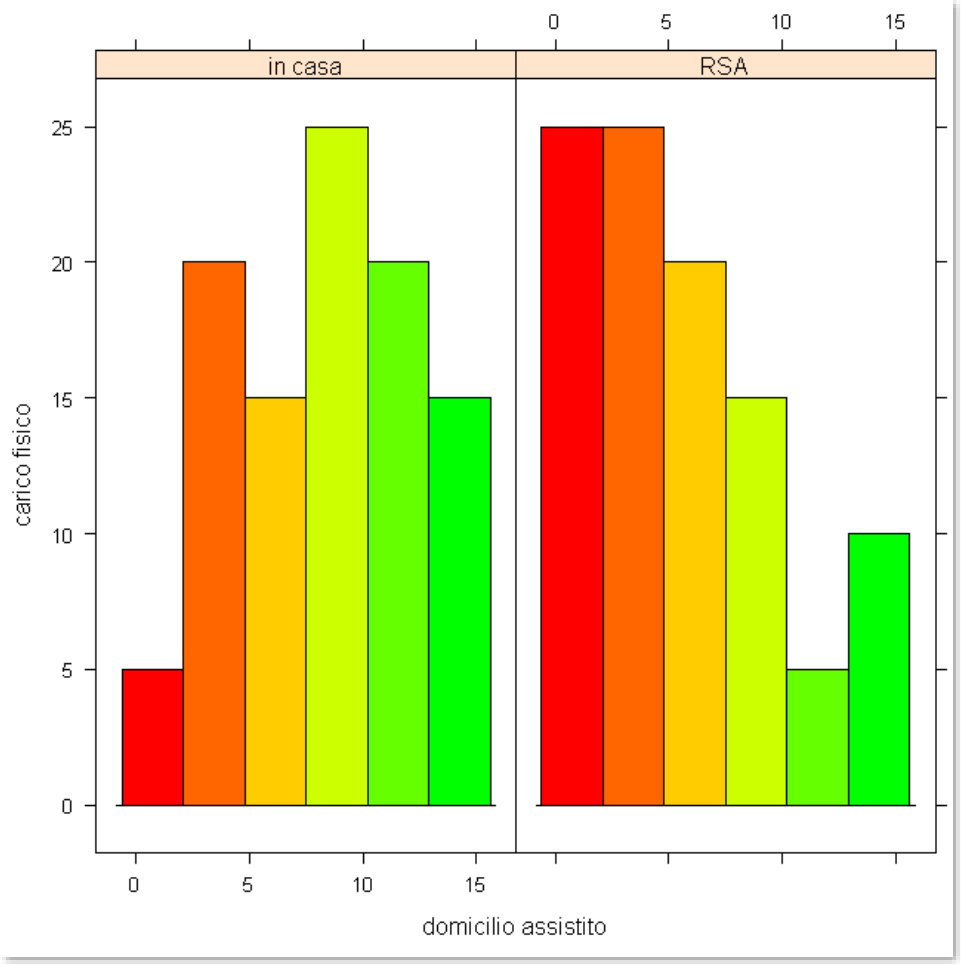


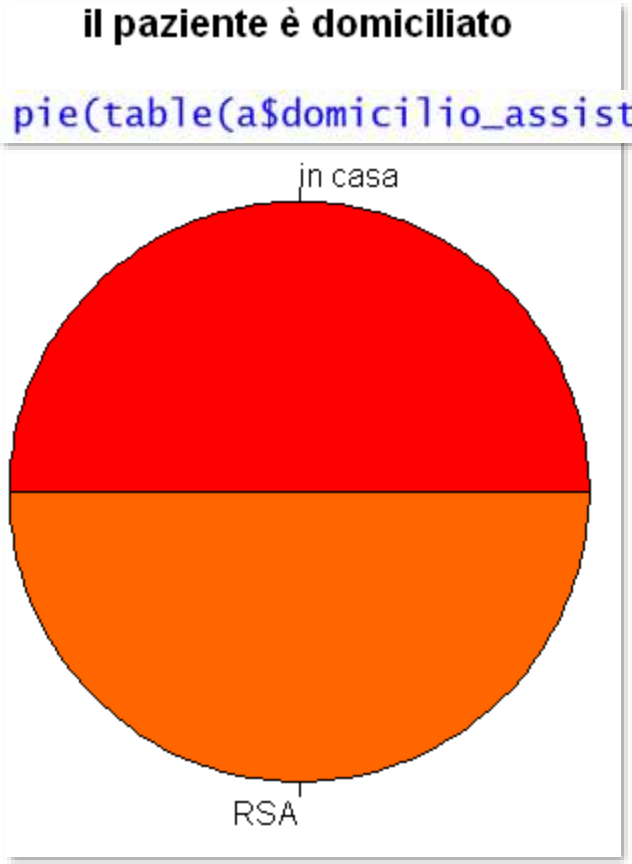
Grafico a torta o pie chart

Nel grafico a torta o ***pie***, **l'area del cerchio corrisponde al totale** dei casi e i suoi **settori** (le "fette" sono **proporzionali alle frequenze.**

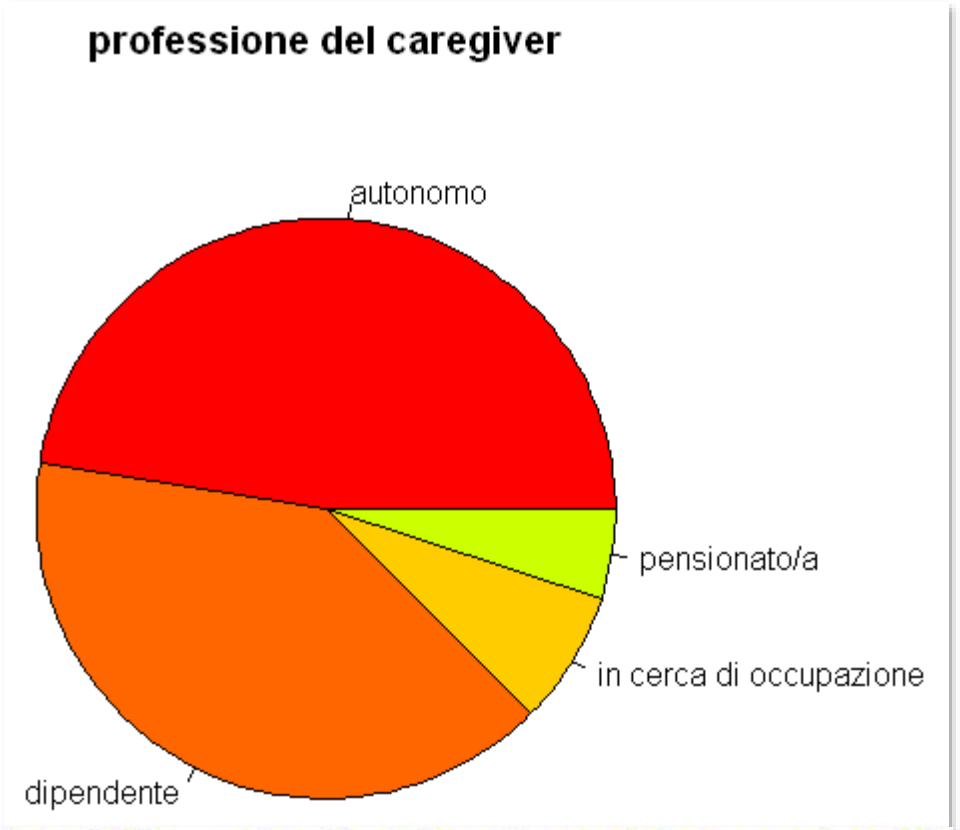
È una **tipologia con pochi estimatori***: "le tabelle di contingenza sono preferibili per dati poco numerosi. Una tabella è sempre preferibile a uno stupido grafico a torta; l'unica presentazione grafica peggiore di un grafico a torta è una lunga serie di grafici a torta [...]"* (Tufte, 1983)*; "il grafico a torta è completamente inutile"* (Bertin, 1981); *"i grafici a torta sono i meno utili tra tutte le forme grafiche"* (Wainer, 1977).

Peccato, perché sono molto facili. In R si usa `pie(table(distribuzione))`; l'argomento `clockwise=TRUE` (di default) o `FALSE` dispone i settori in senso orario (o antiorario) secondo l'ordinamento alfanumerico delle etichette.

Vediamo la distribuzione di frequenza della **collocazione dell'assistito** e della **professione del caregiver**:




```
pie(table(a$domicilio_assistito), col=rainbow(15), main="il paziente è domiciliato")
```




```
pie(table(a$occupazione_attuale), col=rainbow(15), main="professione del caregiver")
```

Grafici specifici

per distribuzioni a livello

ordinale e metrico

Grafico a scatola o boxplot

È uno dei grafici che useremo di più (**Tukey**, **1977**); vi sono rappresentati:

✓**range interquartilico (IR)**: q_1 e q_3 ***(fourths)*** tracciano **il bordo inferiore e superiore** della scatola, la cui **area** è quindi **proporzionale al range interquartilico**: distribuzioni compatte creano scatole corte, distribuzioni con valori molto dispersi scatole più lunghe.

✓**mediana**: indicata dalla **linea spessa** interna alla scatola.

✓**due "baffi" o wiskers**: il "baffo"**superiore** (w_s) si calcola **aggiungendo** a q_3 **una volta e mezzo l'IR,** quello **inferiore** (w_i) **sottraendo a** q_1 **una volta e mezzo l'IR**: Se **eccedono il valore minimo o massimo** della distribuzione, nel **grafico sono fissati al valore minimo e/o massimo**. Altrimenti, sono fissati al loro vero valore e nel grafico compaiono come **cerchietti** i potenziali **outlier** (**outside values**), cioè casi con valori $> w_s$ o $< w_i$)

*"**Because 1 would be too small and 2 would be too large**" (De Veaux et al., 2008)*

Se la mediana è esattamente metà della scatola e i baffi hanno uguale lunghezza, la

distribuzione è affine alla normale teorica (simmetrica e con code simili).

Vediamo un esempio **senza valori anomali**: la distribuzione dello **stress** del caregiver **derivante dall'avere poco tempo per sé,** sottratto dagli impegni dell'assistenza: `$CBI_burden_restrizione_tempo`

`Summary` e `fivenum` riportano gli elementi del boxplot:

```
summary(a$CBI_burden_restrizione_tempo)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0.0     8.0    15.0    12.9    18.0    20.0
```

```
fivenum(a$CBI_burden_restrizione_tempo)
[1]  0  8 15 18 20
```

I baffi si calcolano facilmente, sapendo che:

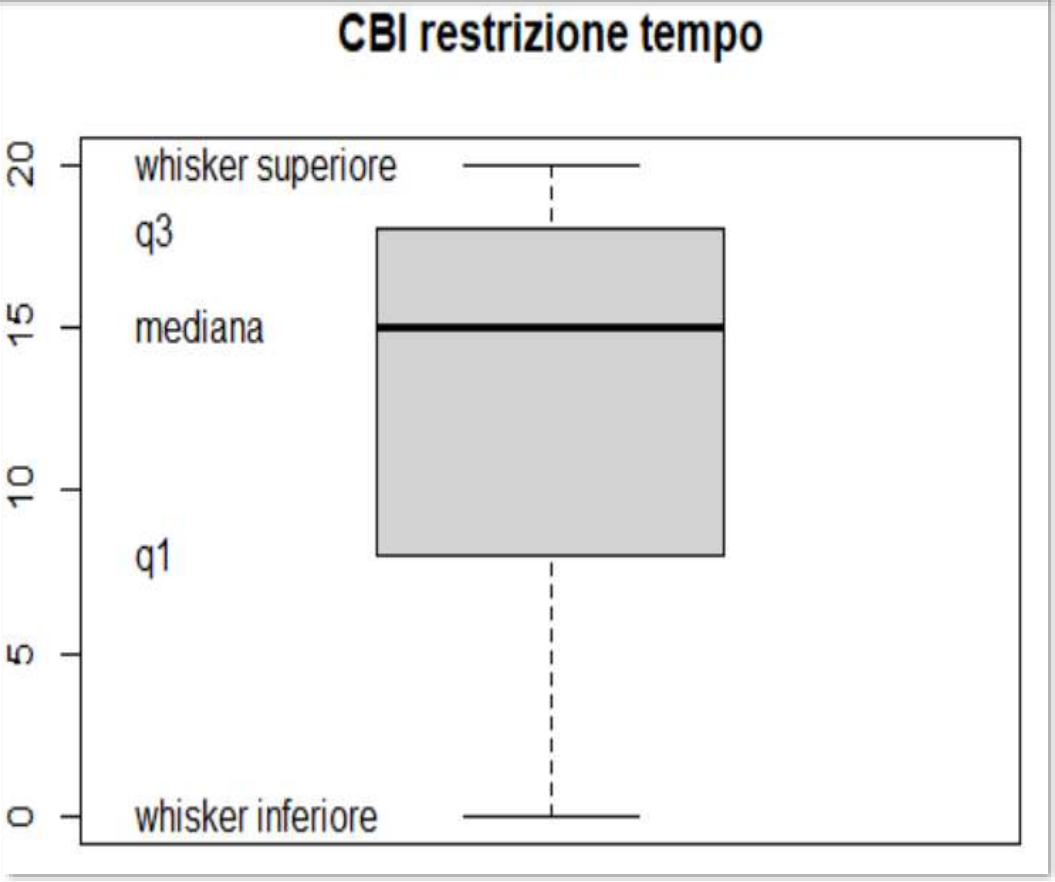
$$w_i = q_1 - 1.5 \times (q_3 - q_1) \qquad w_s = q_3 + 1.5 \times (q_3 - q_1)$$

```
IR<-18-8
inferiore<-8-(1.5*IR)
superiore<-18+(1.5*IR)
c(inferiore, superiore)
[1] -7 33
```

Poiché i **whiskers eccedono 0 e 20**, nel grafico li vedremo impostati al **valore minimo e massimo della distribuzione**.

Rappresentiamo questi valori in **boxplot(variabile)**: l'estensione del "baffo" è gestita da **coef= valore** (di default 1.5). Aggiungiamo la descrizione degli elementi con **text**.

```
boxplot(attaccamento$CBI_burden_restrizione_tempo, main="CBI restrizione tempo")
text(x = .5, y=c(0, 8,15, 18, 20), labels=c("whisker inferiore", "q1", "mediana",
q3", "whisker superiore"), pos = 4)
```



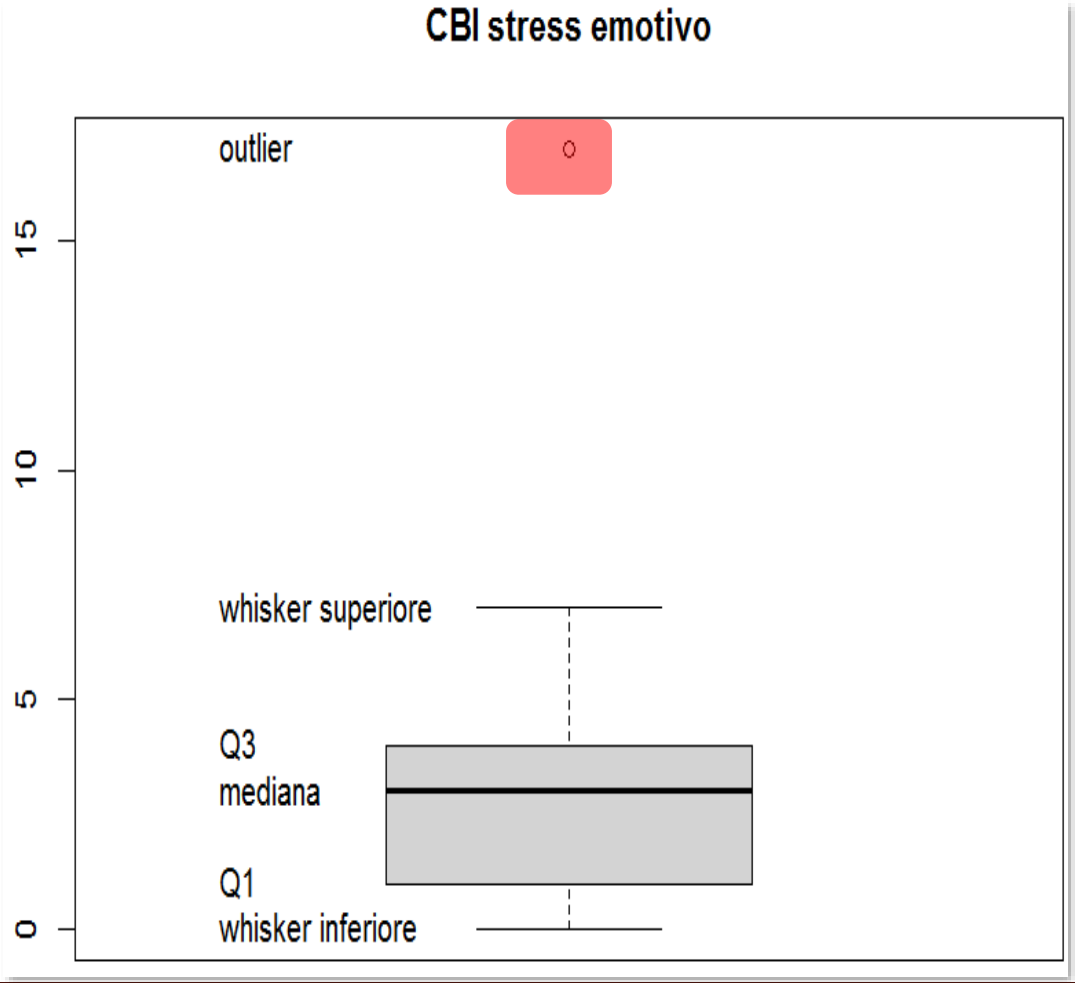
Lo stress derivante dall'avere poco tempo per sé è ***un problema rilevante per molti****: la distribuzione è ampia e variabile, ma la maggior parte dei punteggi si addensa nella parte superiore della scala: metà dei soggetti ha punteggi superiori a 15*

Vediamo un esempio **con valori anomali**: la distribuzione dello **stress** del caregiver **derivante dalla risonanza emotiva negativa** dovuta a comportamenti inappropriati (burden emotivo) : `$CBI_burden_emotivo`. Abbiamo **almeno un outlier superiore** a w_s, cioè il /i caregiver che ha/hanno ottenuto il valore massimo.

La distribuzione è molto più compatta di quella precedente, con punteggi prevalentemente bassi: i caregiver nel complesso non lamentano molto stress per questo aspetto specifico – tranne naturalmente almeno un caso lassù...

... Con `which` identifichiamo i valori anomali:

```
which(a$CBI_burden_emotivo>7)
[1] 2
```



La persona con maggior carico emotivo è il ***caregiver numero 2****: potete descrivere le sue caratteristiche, cioè* ***tracciare un suo profilo****?*

Segnala disagio emotivo di altro tipo?

A quale gruppo appartiene?

Chi lo/la aiuta?

I casi anomali: precisiamo

È importante **evidenziare gli outlier/outside values**, per motivi statistici e interpretativi.

Dal punto di vista statistico, possono **danneggiare il fit di un modello**: per questi casi gli errori del modello (gli scarti dalla media, per esempio) saranno gravi, e la loro eliminazione dal dataframe potrebbe produrre modelli migliori. Ci sono **comunque regole piuttosto rigide** per eliminare gli outlier: ne parleremo diffusamente nella **regressione**.

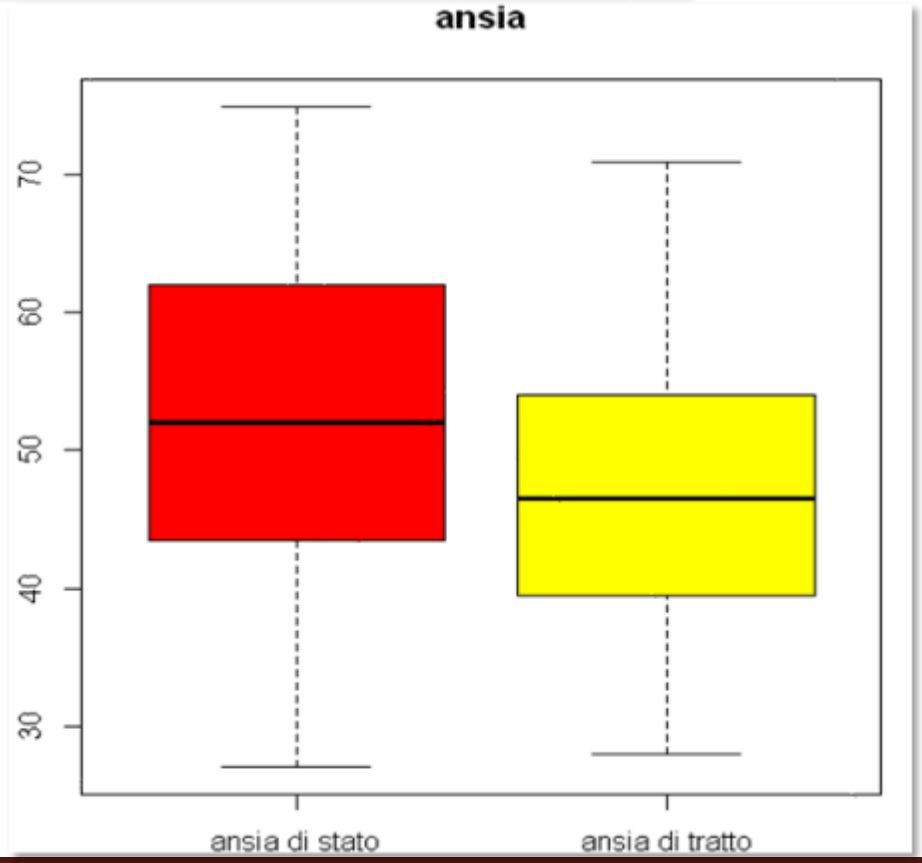
Dal punto di vista interpretativo, possono rappresentare **errori di campionamento** (potrebbero appartenere a una popolazione diversa: soggetti con disturbi emotivi non diagnosticati, reclutati in un campione normativo), o **casi all'estremo di una coda della popolazione** di riferimento (soggetti particolarmente dotati, o al contrario particolarmente a rischio, da segnalare al committente della ricerca)-

*Gli outlier univariati e multivariati di cui parleremo più avanti sono identificati dai loro scarti **dalla media** ($\pm|2|sd$dalla media), mentre il riferimento degli outside values è il range interquartilico. Comunque, potremo usare "outlier" per identificare genericamente valori "molto, molto anomali"*

Nello **stesso grafico** possono stare **due o più boxplot**, se le variabili hanno stessa **unità di misura** e stesso **range teorico** min-max: si **separarano con ","**. Dovremo **specificare i loro nomi** con `names=c("nome1","nome2")`.

Ecco le due **scale di ansia di stato e di tratto:**

```
boxplot(a$STAI_stato, a$STAI_tratto, main="ansia",
  col=rainbow(6), names=c("ansia di stato", "ansia di tratto"))
```

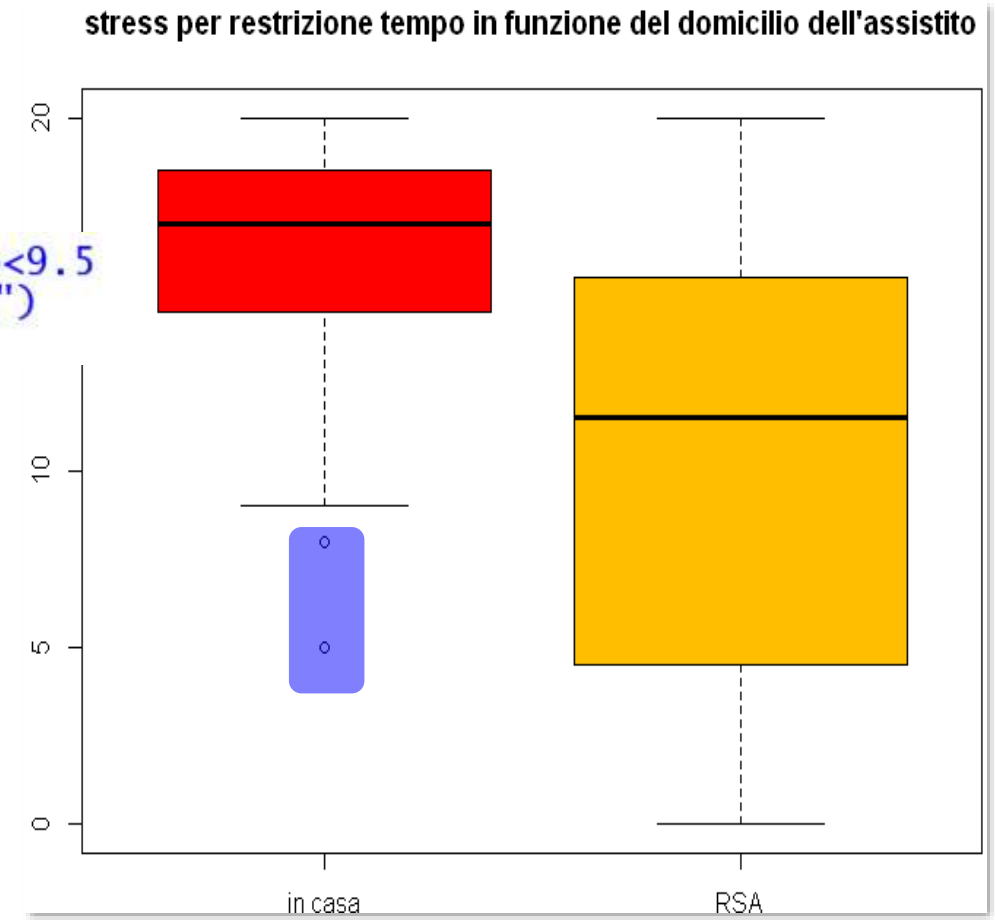


*Si direbbe che l'ansia dovuta alla **situazione contingente prevalga**, rispetto alla predisposizione ansiosa.*

Possiamo rappresentare la **distribuzione di una misura in funzione di un fattore**, usando **~**.

Per l'analogo con descrittori numerici, ricordiamoci **tapply**.

```
boxplot(a$CBI_burden_restrizione_tempo~a$domicilio_assistito, col=rainbow(15),
  main="stress per restrizione tempo in funzone del domicilio del paziente")
```



Identifichiamo gli outlier "in casa" :

```
which(a$CBI_burden_restrizione_tempo<9.5
   & a$domicilio_assistito=="in casa")
[1]  1  2 20
```

*Toh, sono **tre** e non due. I loro punteggi nella variabile (10_a colonna nel dataframe) sono:*

```
a[c(1,2,20),10]
[1] 5 9 8
```

I cerchietti dei punteggi 8 e 9 sono sovrapposti!

Verificate la distribuzione delle ***altre variabili di burden nei due livelli della variabile $domicilio_assistito****: anche le altre dimensioni di stress sembrano risentire del gruppo di appartenenza? Ricoverare temporaneamente l'assistito allevia anche altre dimensioni di carico assistenziale?*

Perché?

RcmdrMisc e gplots

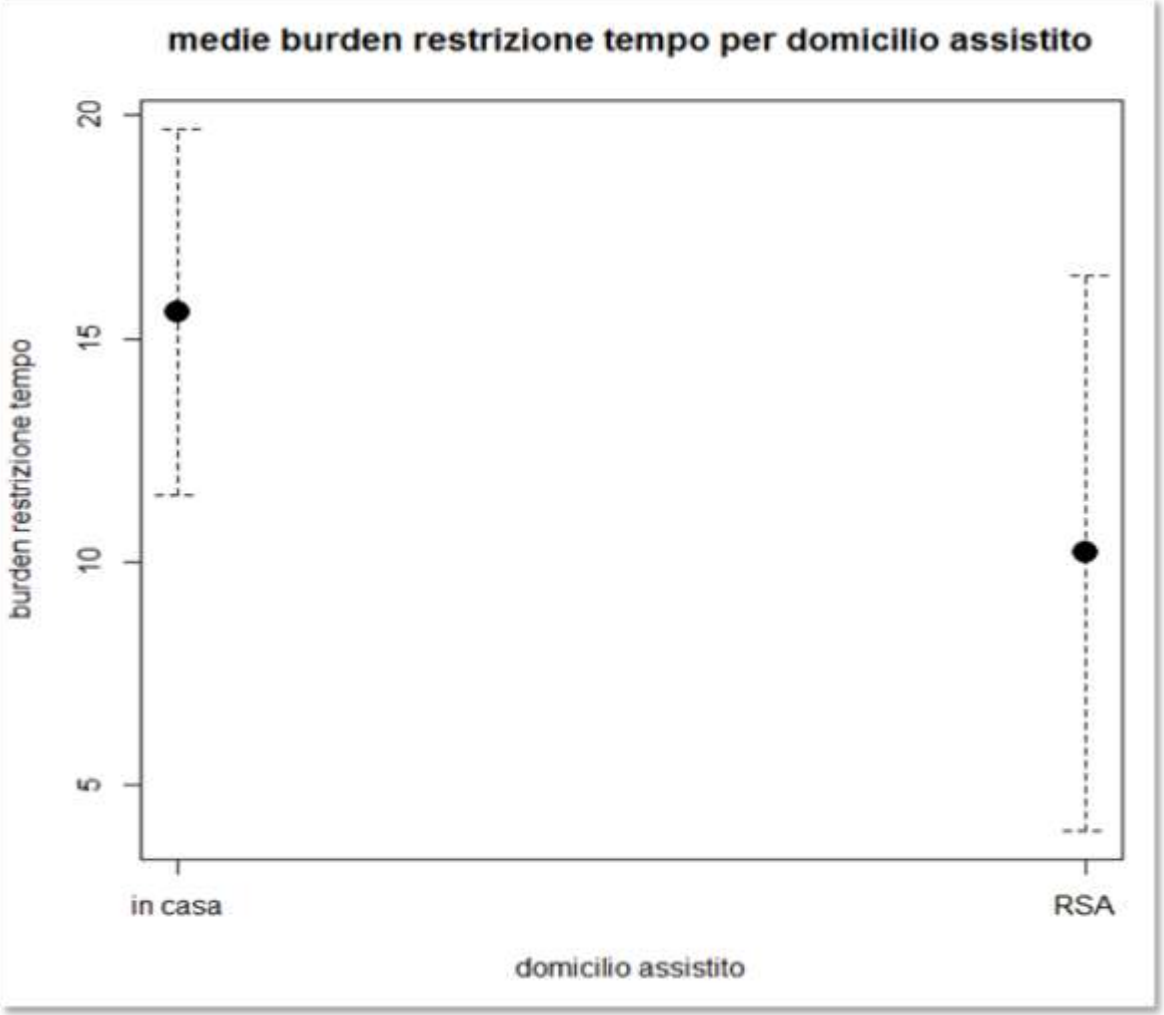
Nel package `RcmdrMisc` che viene installato insieme a `RCommander` , ma può essere anche **scaricato a parte**, ci sono funzioni che possono ampliare il repertorio di grafici a disposizione.

Per visualizzare le medie di una variabile a seconda dei livelli di un fattore si usa `plotMeans(misura, fattore, barre di errore`, cioè indici di dispersione attorno alla media).

`response=` indica la misura, `factor1=` fattore per i livelli dei quali sono rappresentate le medie della misura, `error.bars=` un indice di dispersione da scegliere tra **deviazione standard** (=`"sd"`), o **errore standard della media** (=`"se"`) o **intervallo di fiducia** (=`"conf.int"`) che affronteremo tra un po'. I punti che rappresentano le medie sono di default connessi da una linea, che può essere omessa con `connect=FALSE`.

```
plotMeans(response= a$CBI_burden_restrizione_tempo,factor1= a$domicilio_assistito,
  pch = 19,xlab="domicilio assistito", ylab="burden restrizione tempo", main="restrizione
  tempo per domicilio assistito", error.bars = "sd", connect=FALSE)
```

Vediamo le **medie** del burden derivante dal poco tempo a disposizione; usiamo la **sd** come indice di dispersione:



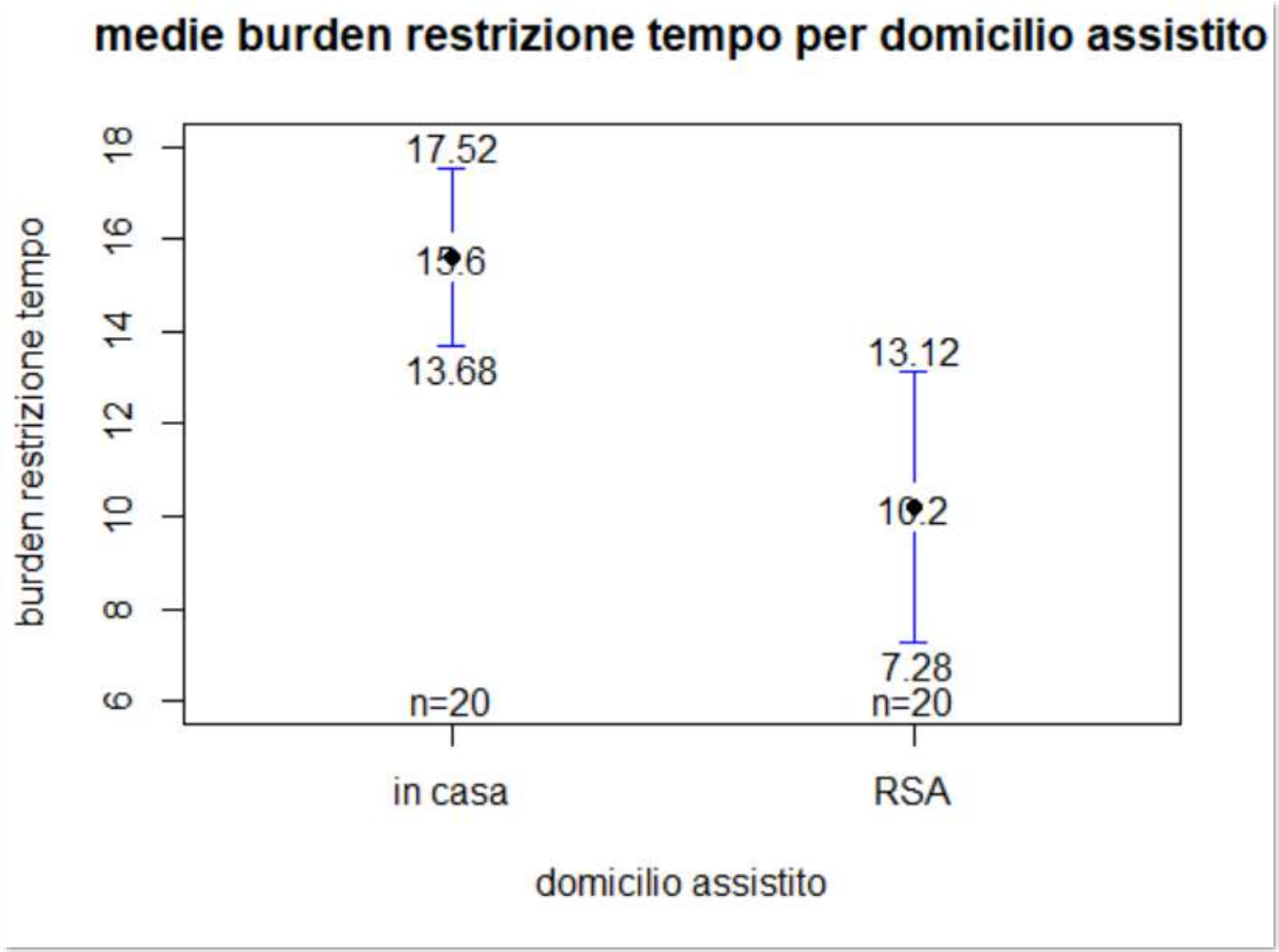
`plotmeans(formula= Y~fattore)` di **gplots** crea un grafico come il precedente, tracciando, come barre di errore, l'intervallo di fiducia CI attorno alle medie (`bar= "confint"`). La useremo molto nelle statistiche inferenziali per il confronto tra medie.

Gli argomenti `n.label= TRUE/FALSE`, `mean.labels= TRUE/FALSE` e `ci.label= TRUE/FALSE` aggiungono la N, la $\bar{x}$ e i limiti del CI al 95% per ogni gruppo. Limitiamo i decimali specificandone il numero in `digits=`. `connect= FALSE` non collega le medie (gruppi indipendenti).

Per creare il graficodel burden dovuto al poco tempo a disposizione in funzione del domicilio dell'assistito, scriveremo:

```
plotmeans(a$CBI_burden_restrizione_tempo ~ a$domicilio_assistito,
  pch = 19,xlab="domicilio assistito", ylab="burden restrizione tempo", main="medie burden
  restrizione tempo per domicilio assistito", connect=FALSE, n.label=TRUE, mean.labels=TRUE,
  ci.label=TRUE, digits=2, ylim=c(6,18), bars=TRUE)
```

`n.label` e `bars` sono TRUE di default.


medie burden restrizione tempo per domicilio assistito
burden restrizione tempo
6
8
10
12
14
16
18
17.52
15.6
13.68
13.12
10.2
7.28
n=20
n=20
in casa
RSA
domicilio assistito

plotMeans e *plotmeans* mostrano le medie di una distribuzione, con relativa barra di errore, solo in funzione dei livelli di un fattore. Come potremmo usare una o l'altra di queste due funzioni per rappresentare la media di un campione nel suo complesso, senza suddivisione in gruppi? Per esempio, come potremmo rappresentare la media del burden dovuto alla restrizione del tempo nel campione complessivo, con relativa barra di errore?

Ci vuole un piccolo trucco, sapreste indovinare quale?

Indici di forma

Dopo aver usato ***medie e mediane*** avere un'idea dell'ordine di grandezza del fenomeno e gli ***indici di dispersione*** per segnalare il grado di diversità tra le sue singole manifestazioni, uniremo i grafici (`hist`, in particolare) a indici numerici (**indici di forma**) per completare il quadro delle tecniche per la comprensione delle caratteristiche di una distribuzione statistica

Dalla distribuzione gaussiana...

"The supreme law of Unreason"
Galton, 1855

Le distribuzioni di frequenza possono assumere molte forme diverse. Per esempio, i dati di una variabile **continua** potrebbero essere distribuiti simmetricamente attorno alla tendenza centrale della distribuzione: se tracciassimo una linea verticale per il centro della distribuzione, questa sarebbe speculare, ovvero **simmetrica** ai due lati della linea.

Questa è una caratteristica della **distribuzione normale**. La sua storia viene da lontano:

- ✓ **Laplace** (1812) lavora sulle distribuzioni di probabilità associate al gioco d'azzardo e interpreta la curva come **legge dell'errore**
- ✓ **Gauss** (1855) sistematizza le osservazioni di Laplace nella funzione della "curva gaussiana" o distribuzione di Laplace – Gauss;
- ✓ Nel corso del XIX secolo la gaussiana si diffonde, anche grazie alla crescita delle compagnie assicurative e all'applicazione di un approccio statistico alle **scienze biologiche e sociali**.

… alla distribuzione normale

Il **passaggio alla denominazione della curva da gaussiana a "normale"** si deve all'osservazione che **molte variabili biologiche**, **se misurate in grandi gruppi**, hanno distribuzioni di frequenza **strettamente approssimate** alla curva normale.

*In realtà, la "vera" **distribuzione normale è solo teorica**: l'istogramma di frequenza, per quanto piccole possano essere le classi, è una curva **discontinua**, non continua.*

Quetelet (1835) descrive l'***homme moyen***: è l'idea che la Natura ha di come deve essere un uomo, un **ideale che corrisponde a un valore misurato medio**.

Però la **Natura commette errori**, **creando** così **la variabilità osservata** nei tratti fisici e psicologici dell'uomo. **L'estensione e la frequenza di questi errori** della Natura si **conformano** alla legge della frequenza degli errori, ovvero alla **distribuzione normale**.

Galton, grandemente impressionato da questa concezione e con il concorso dell'allievo Karl Pearson, impianta decisamente e forse definitivamente la **curva normale** nella psicologia:

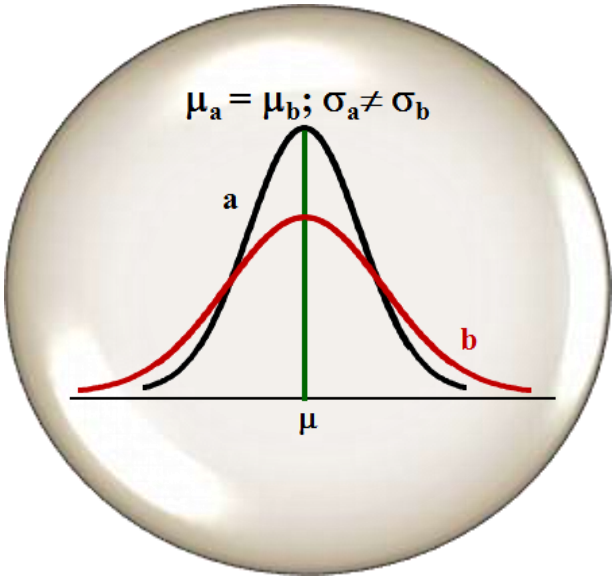
Conosco ben poche cose così capaci di colpire l'immaginazione come la meravigliosa forma di ordine cosmico espressa dalla "legge di frequenza degli errori". La ***legge sarebbe stata personificata dai Greci e deificata****, se l'avessero conosciuta. Regna con* ***serenità*** *e in volontaria discrezione in mezzo alla più selvaggia confusione.*

Tanto più smisurata è la moltitudine e tanto più grande è l'anarchia, quanto più perfetta appare la sua regola.

È la suprema legge dell'Irrazionale*. Ogni volta che una grande massa di elementi caotici viene raccolta e questi elementi sono schierati secondo l'ordine della loro grandezza, un'insospettabile e splendida forma di regolarità dimostra di essere stata latente fin dall'inizio"*

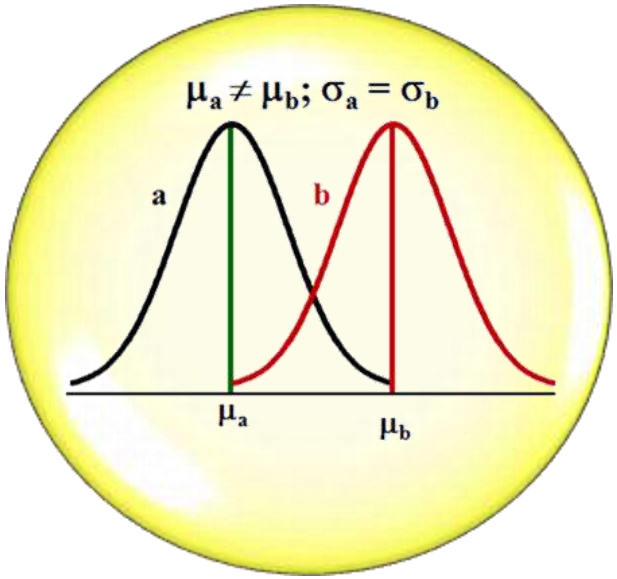
Galton, Natural inheritance, 1889, pag. 66

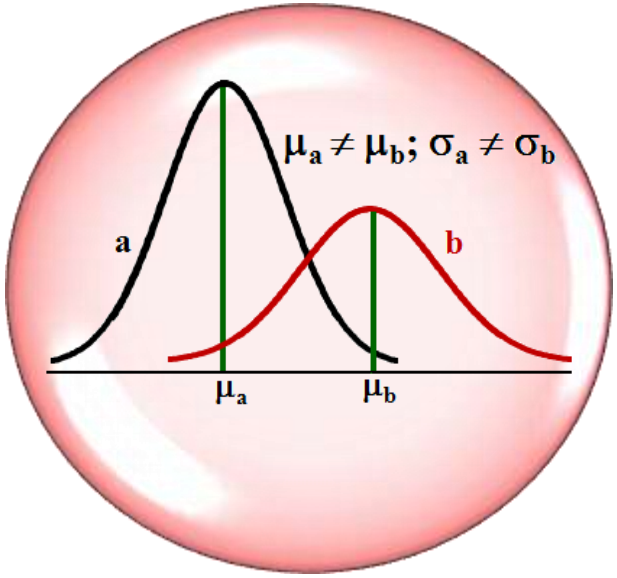
Noi avremo un approccio più laico alla distribuzione normale, che è in realtà una **famiglia** di distribuzioni, definite da **due parametri:** media μ e deviazione standard σ:



in X i valori della distribuzione,

in Y la loro frequenza





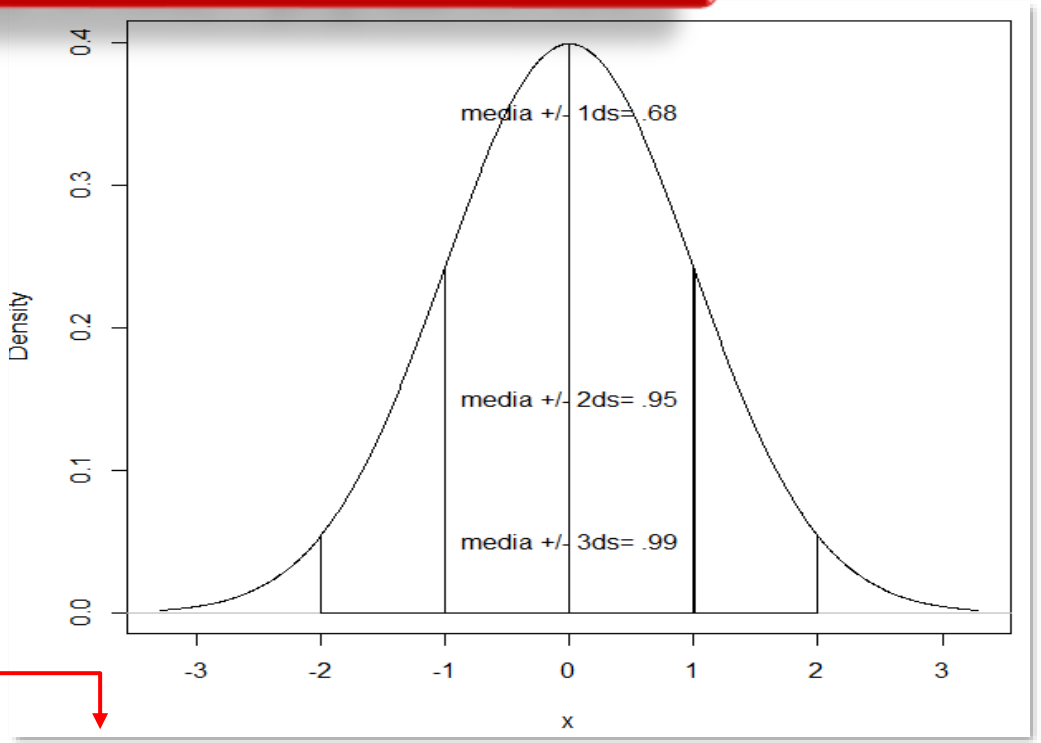
La distribuzione **normale** è *asintotica*: non tocca mai l'ascissa X, se non in corrispondenza di $\pm\infty$. **Moda, mediana e media coincidono nel valore centrale**; ogni metà della curva presenta punti di flesso, in cui la curva cambia direzione, corrispondenti a $\pm\boldsymbol{\sigma}$.

Il 100% dei casi è compreso nell'area delimitata dalla curva: l'area a essa sottesa è quindi = 1. Per **qualunque valore di μ e σ**, l'area corrispondente a intervalli definiti, ovvero la **proporzione di casi compresi sotto la curva è sempre la stessa**:

$$\mu \pm 1\sigma = .68; \quad \mu \pm 2\sigma = .95; \quad \mu \pm 3\sigma = .99$$

Impareremo a rappresentarla con R tra poco.

Disegnarla con Rcommander è facile: *Distribuzioni → Distribuzione continue → distribuzione normale → disegna distribuzione normale, più la funzione* `text` *usando lo script*




```
x <- seq(-3.291, 3.291, length.out=1000)
plotDistr(x, dnorm(x, mean=0, sd=1), cdf=FALSE, xlab="x", ylab="Density",
   main=paste("Normal Distribution:  Mean=0, Standard deviation=1"),
   regions=list(c(-2, -1), c(-1, 0), c(0,1), c(1,2)), col=c('#FFFFFF',
   '#FFFFFF','#FFFFFF
text(x =-1, y=c(0.35, 0.15, 0.05), labels = c("media +/- 1ds= .68", "media +/- 2ds=
   .95", "media +/- 3ds= .99"), pos = 4)
```

L'asimmetria

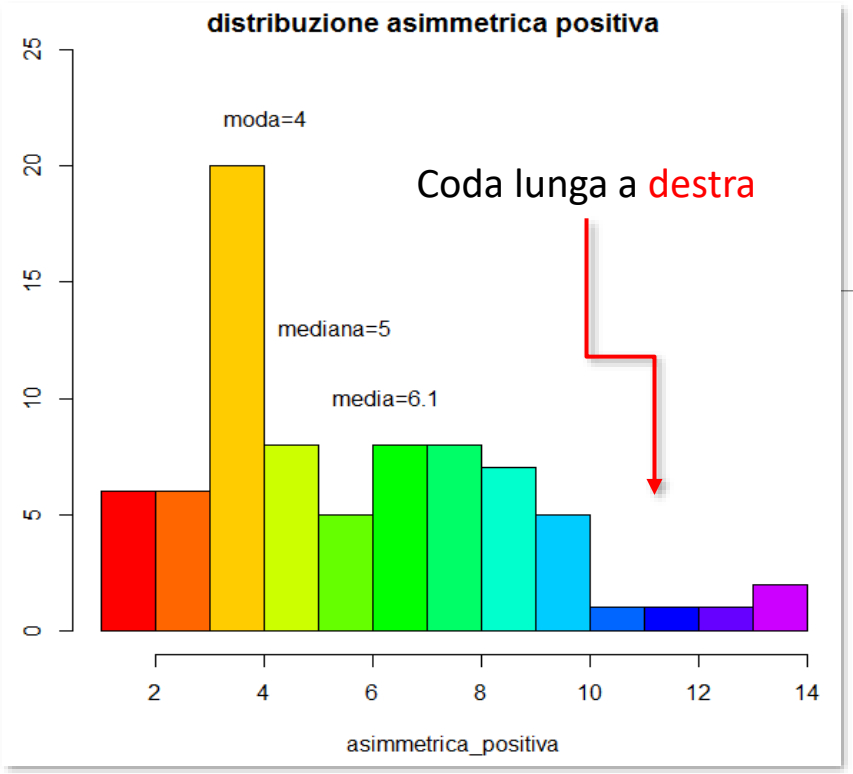
Gli indici di forma della distribuzione normale teorica sono le pietre di paragone per gli indici di forma delle distribuzioni campionarie.

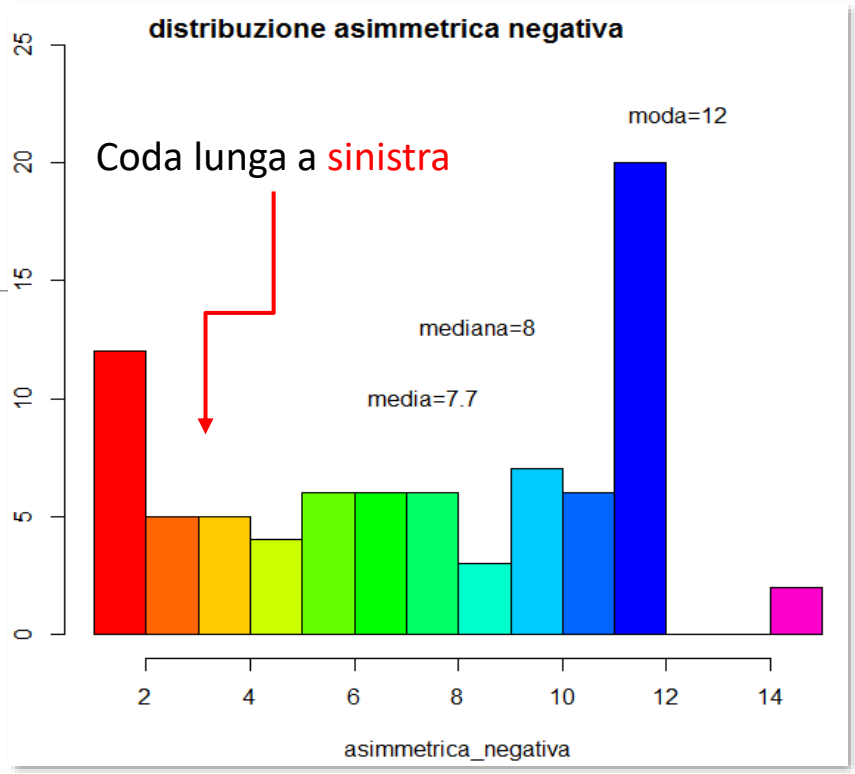
La **coincidenza di moda, mediana e media** rende la curva **simmetrica**: quando non sono equivalenti, la distribuzione è **asimmetrica**. Nella distribuzione normale, $\boldsymbol{asimmetria = 0}$.

L'**asimmetria** (**skewness**) può essere positiva ($> \mathbf{0}$) o negativa ($< \mathbf{0}$).

Quando $\boldsymbol{media > mediana > moda}$, **l'asimmetria è positiva**: la distribuzione presenta un **maggior numero di casi con valori bassi** della distribuzione → **coda più lunga a destra**.

Quando $\boldsymbol{media < mediana < moda}$, **l'asimmetria è negativa**: la distribuzione presenta un **maggior numero di casi con valori alti** della distribuzione → **coda più lunga a sinistra**.






```
mean(asimmetrica_positiva)
[1] 6.089744
median(asimmetrica_positiva)
[1] 5
table(asimmetrica_positiva)
asimmetrica_positiva
 1  2  3  4  5  6  7  8  9 10 11 12 13 14
 2  4  6 20  8  5  8  8  7  5  1  1  1  2
```

```
mean(asimmetrica_negativa)
[1] 7.731707
median(asimmetrica_negativa)
[1] 8
table(asimmetrica_negativa)
asimmetrica_negativa
 1  2  3  4  5  6  7  8  9 10 11 12 15
 4  8  5  5  4  6  6  6  3  7  6 20  2
```

La curtosi

La **curtosi** (**kurtosis**) indica **quanto le code della distribuzione siano più o meno addensate** rispetto alle code di una distribuzione normale.

Nelle formule di Pearson, asimmetria e curtosi di una normale teorica risultavano = 3; le formule sono state successivamente riadattate in modo che **entrambi gli indici di forma della distribuzione normale teorica siano =0**, facilitandone l'interpretazione.

Una $\boldsymbol{curtosi = 0}$ indica **coincidenza con la gaussiana** (distribuzione **mesocurtica**).

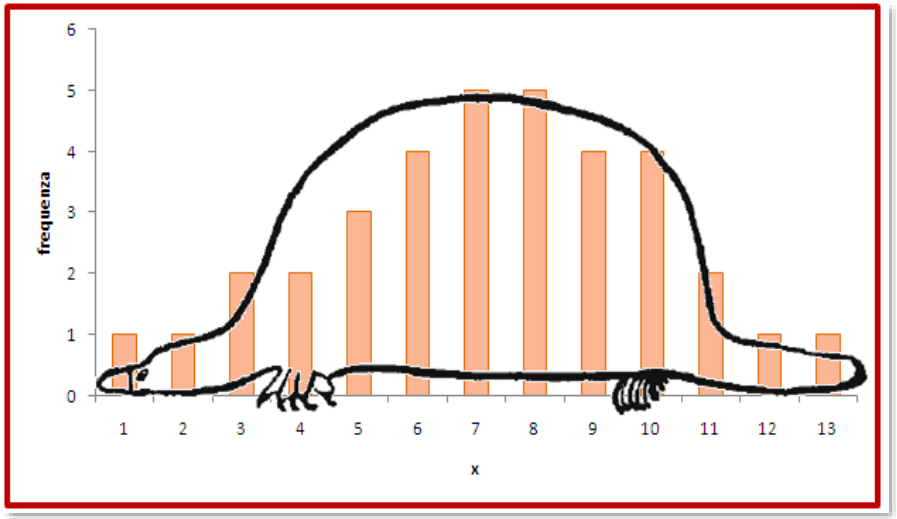
Una curtosi **negativa** (< 0) indica un eccesso relativo di osservazioni nelle zone intermedie a destra e a sinistra del centro, quindi **code scarsamente differenziate** dai valori centrali: distribuzione **platicurtica** (platùs= piatto).

Una curtosi **positiva** (> 0) si ritrova in una distribuzione con **code sottili, chiaramente differenziate** dai valori centrali: **leptocurtica** (leptòs: sottile).

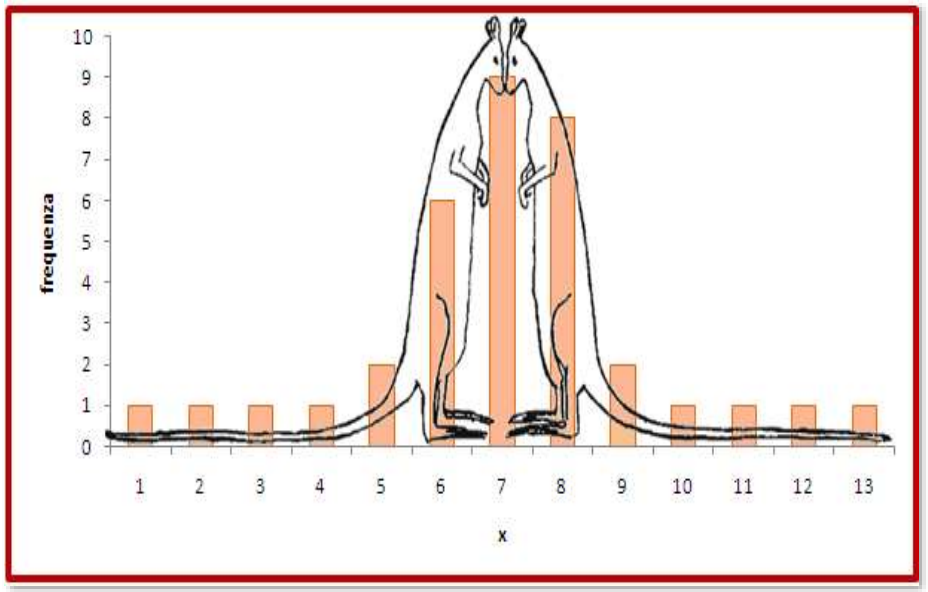
Gosset (meglio noto come Student) ha proposto una curiosa mnemotecnica per la curtosi:

> The important property which follows from this is that platykurtic curves have shorter "tails" than the normal curve of error and leptokurtic longer "tails." I myself bear in mind the meaning of the words by the above *memoria technica*, where the first figure represents platypus, and the second kangaroos, noted for "lepping," though, perhaps, with equal reason they should be hares!

Errors of routine analysis, Biometrika,19,1927



Convenzionalmente, asimmetria e curtosi in un range da -1 a $+1$ (per altri, da -1.5 a $+1.5$) indicano deviazioni **trascurabili** rispetto agli indici di una distribuzione normale teorica.



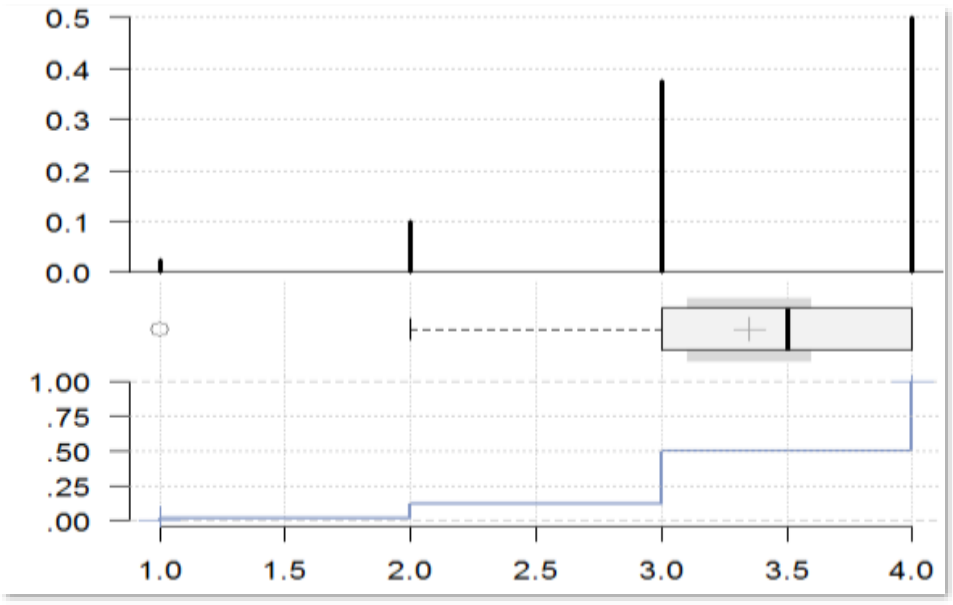
Skewness e curtosi **non** sono statistiche di base in R: usiamo `Desc(variabile)` di `DescTools`

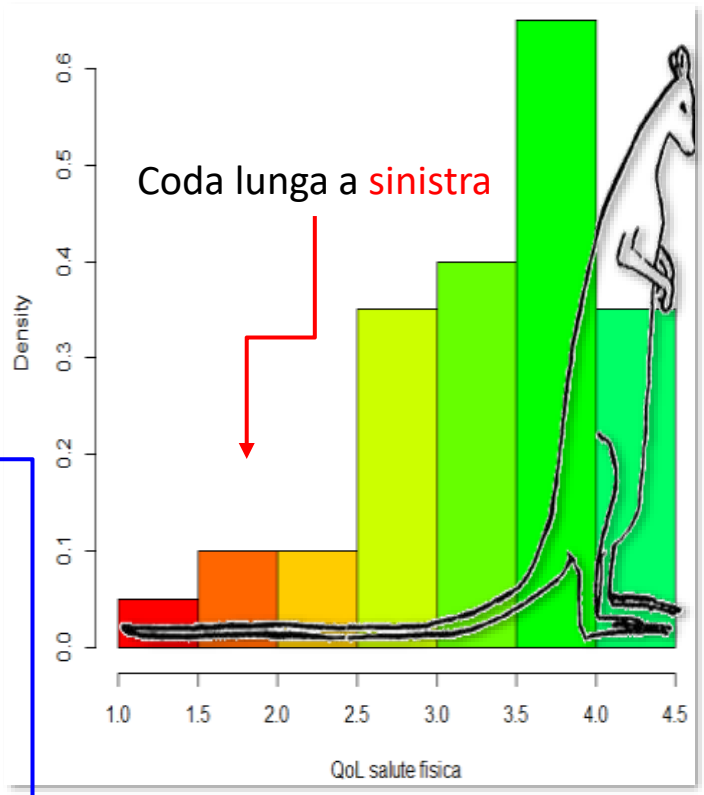
```
Desc(attaccamento$QOL_salute_fisica)
-----------------------------------------------------------
attaccamento$QOL_salute_fisica (integer)

  length       n     NAs  unique      0s    mean   meanCI'
      40      40       0       4       0    3.35      3.10
          100.0%    0.0%            0.0%                3.60

     .05     .10     .25  median     .75     .90      .95
    2.00    2.00    3.00    3.50    4.00    4.00     4.00

   range      sd   vcoef     mad     IQR    skew     kurt
    3.00    0.77    0.23    0.74    1.00   -0.99     0.39
```





***curtosi positiva**: distribuzione leptocurtica, con coda ben distinguibile*

***Asimmetria negativa**: valori addensati verso la coda sinistra*

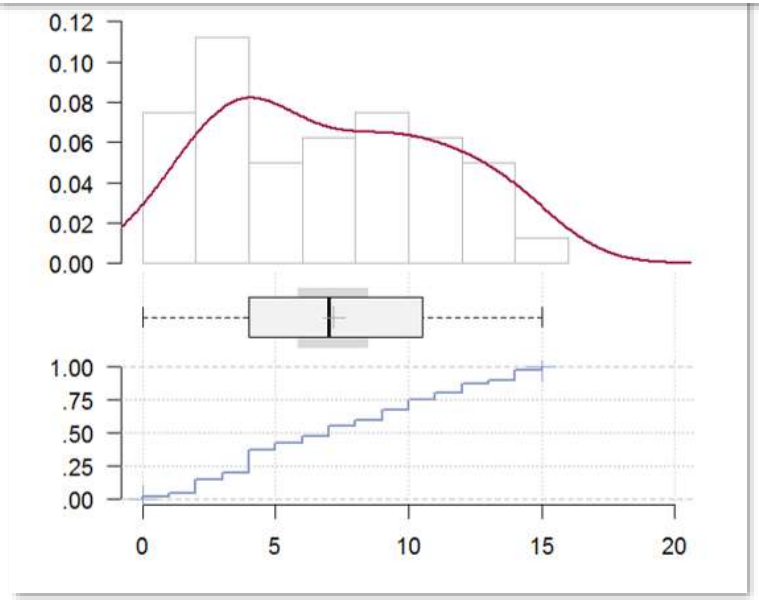
La distribuzione dello **stress derivante dall'onere fisico dell'assistenza** è diversa:

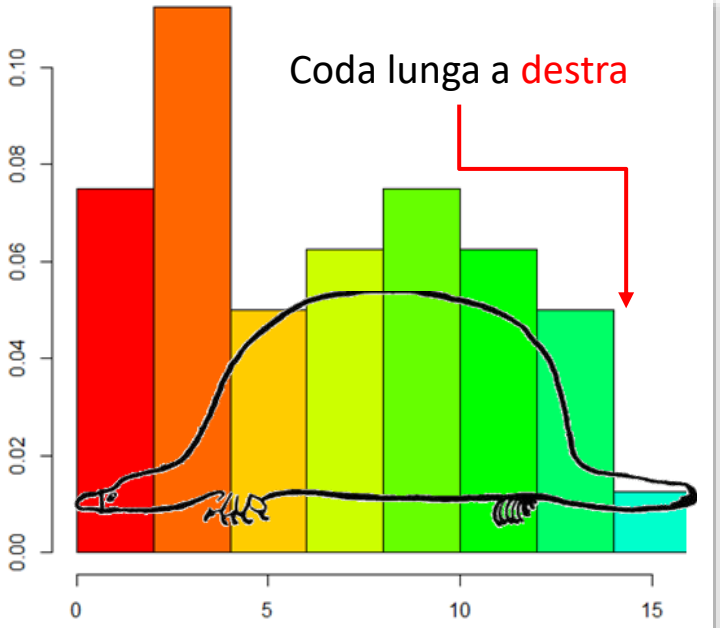
```
Desc(attaccamento$CBI_burden_fisico)
----------------------------------------------------------------
attaccamento$CBI_burden_fisico (integer)

  length         n       NAs    unique        0s      mean     meanCI'
      40        40         0        16         1      7.17        5.84
          100.0%     0.0%                 2.5%                    8.51

     .05       .10       .25    median       .75       .90       .95
    1.95      2.00      4.00      7.00     10.25     13.10     14.00

   range        sd     vcoef       mad       IQR      skew      kurt
   15.00      4.19      0.58      4.45      6.25      0.21     -1.21
```





***curtosi negativa**: distribuzione platicurtica, con code poco differenziate*

***Asimmetria positiva**: valori addensati verso la coda destra*

Per i **soli valori** di asimmetria e curtosi, ci sono anche `Skew(variabile)` e `Kurt(variabile)` di `DescTools`.

Quali considerazioni possiamo fare sulla salute fisica e sullo stress derivante dall'assistenza fisica dei caregiver?

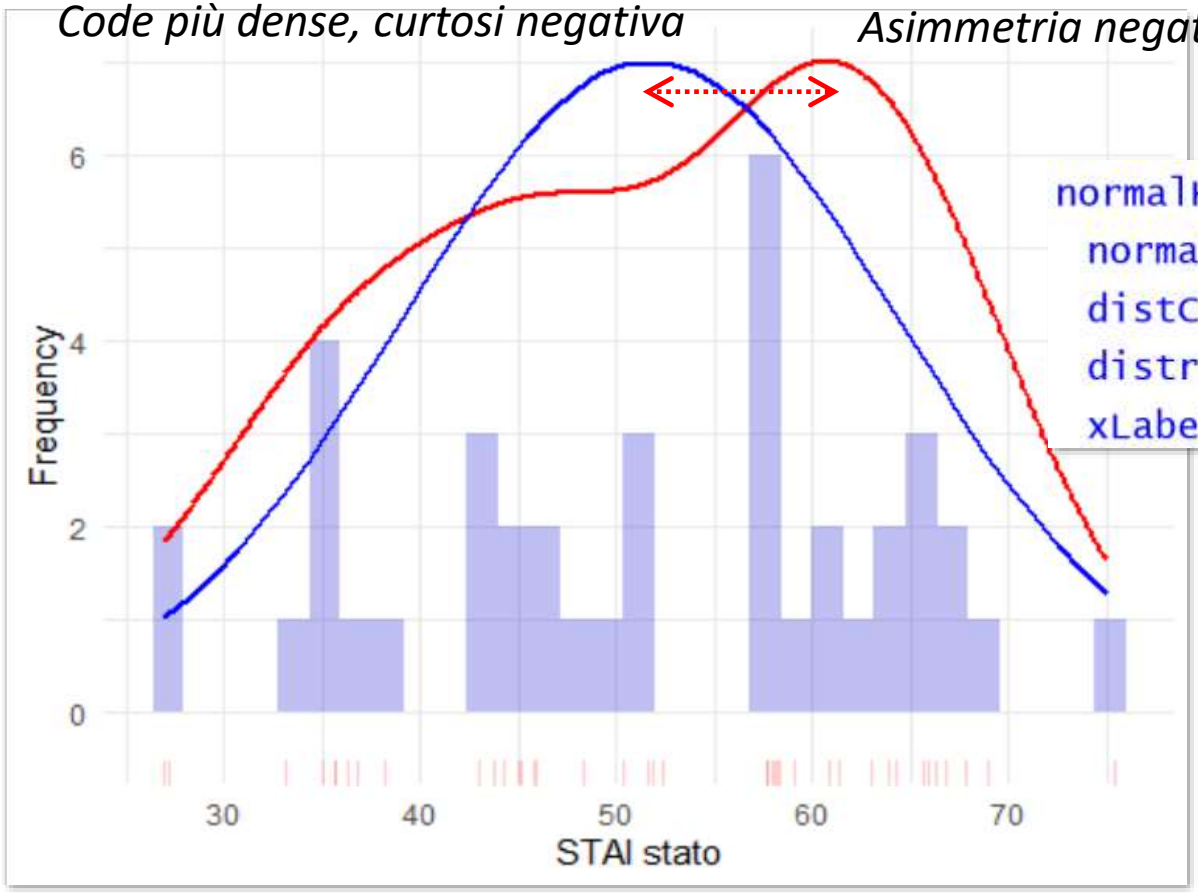
*Quale **covariata** non dovremmo trascurare nei commenti?*

Come metterla alla prova?

Per un confronto più facile tra istogramma di frequenze osservate e distribuzione normale teorica, potete usare `normalHist(vector= distribuzione, normalCurve= TRUE, distCurve= TRUE)` del package `ufs`. La distribuzione empirica *smoothed* (`distCurve`) e la normale teorica (`normalCurve`) sono mostrate di default.

Code più dense, curtosi negativa

Asimmetria negativa



Questa è **l'ansia di stato**.

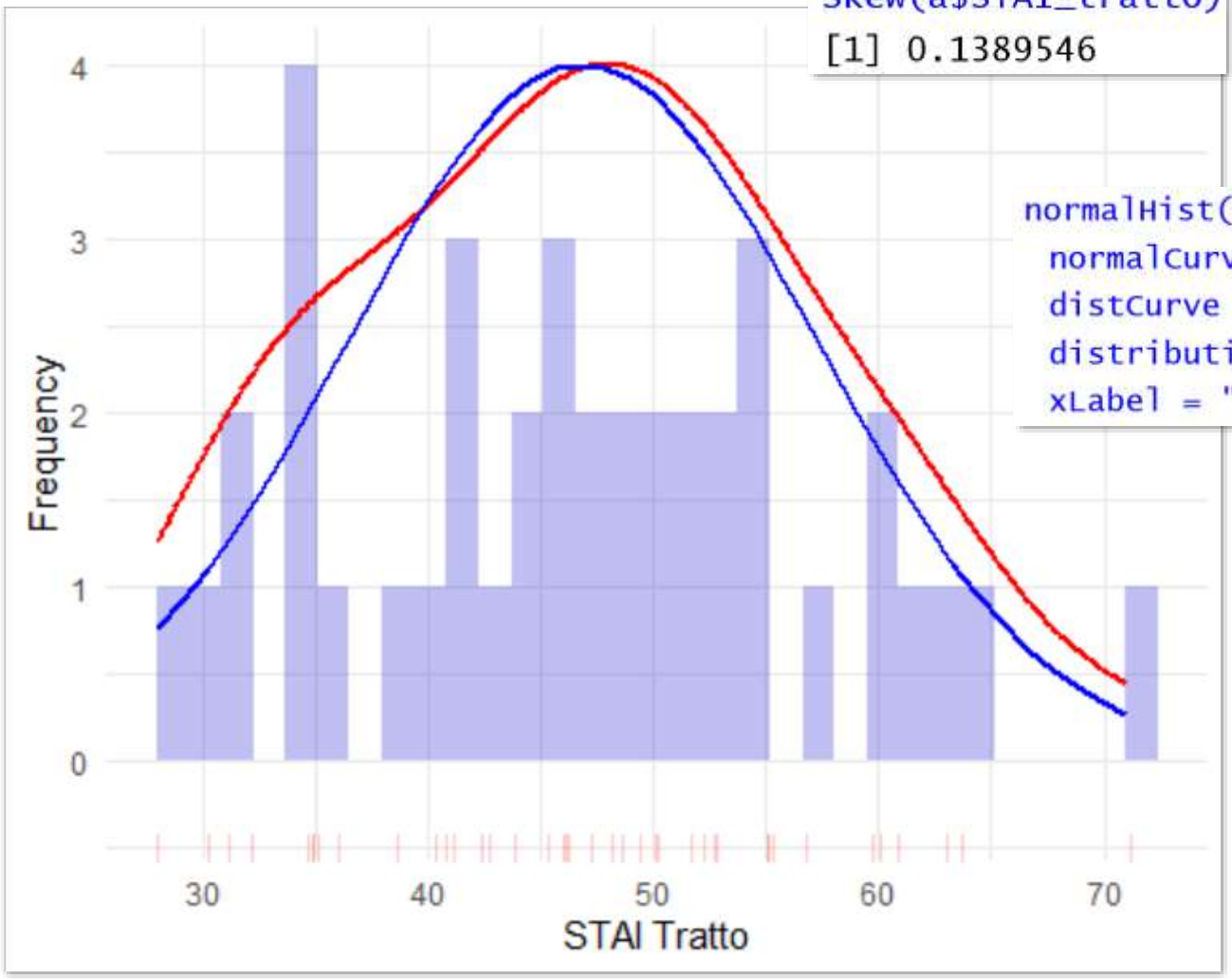
```
normalHist(vector=a$STAI_stato,
  normalCurve=TRUE,
  distCurve =TRUE, normalColor= "blue",
  distributionColor= "red",
  xLabel= "STAI stato")
```

```
Skew(a$STAI_stato)
[1] -0.2313227
Kurt(a$STAI_stato)
[1] -1.071975
```

L'asimmetria dell'**ansia di tratto** è molto più ridotta, così come anche la sua curtosi, anche se ancora negativa, è molto più vicina a quella della normale teorica:

```
skew(a$STAI_tratto)
[1] 0.1389546
```

```
Kurt(a$STAI_tratto)
[1] -0.7128255
```




```
normalHist(vector= a$STAI_tratto,
 normalCurve= TRUE,
 distCurve = TRUE, normalColor = "blue",
 distributionColor = "red",
 xLabel = "STAI Tratto")
```

Potete provare a spiegare la diversa distribuzione delle due scale di ansia, Stato e Tratto?

Il Q-Q plot

Il **Q-Q plot** (grafico quantile – quantile) confronta i **quantili** di due distribuzioni X e Y: ogni punto del Q-Q plot ha come coordinata l'n-esimo quantile di X e il corrispondente quantile di Y.

Se X e Y hanno un andamento simile, i punti del Q-Q plot si dispongono approssimativamente su una retta $X = Y$.

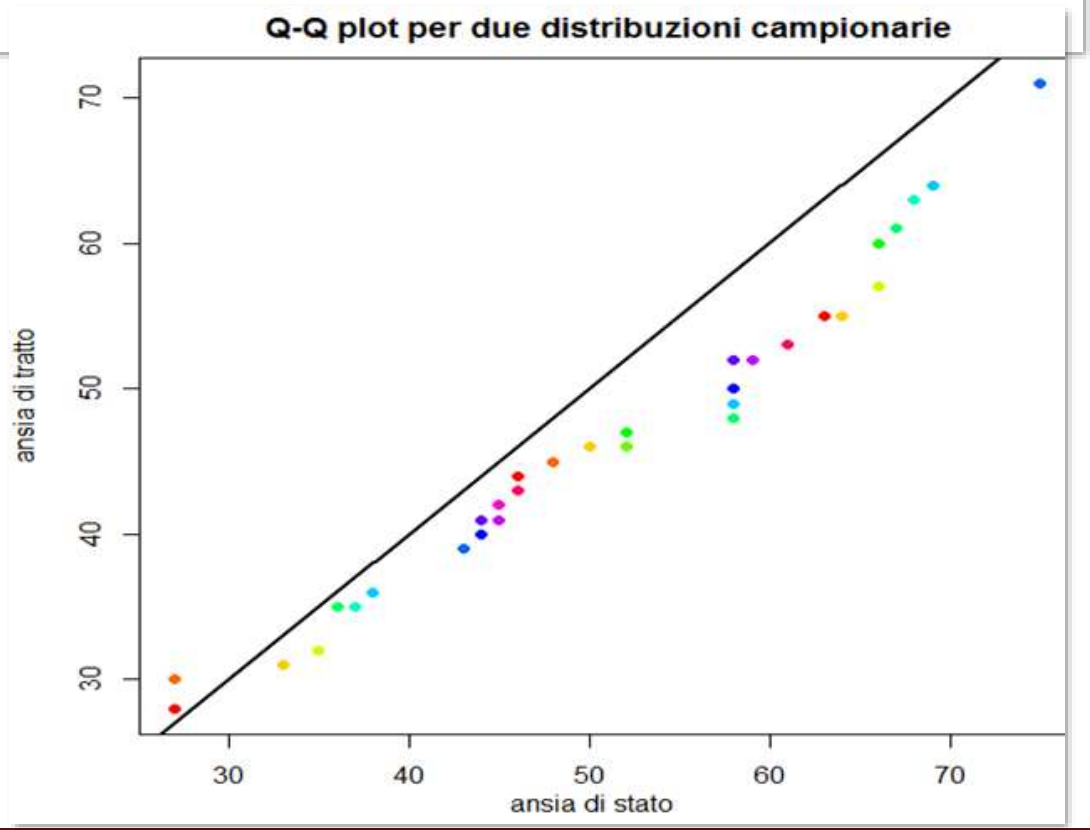
Il Q-Q plot confronta la forma di **due distribuzioni campionarie** oppure, più spesso, **la forma di una distribuzione campionaria con una attesa** (**normale**, di solito): più le distribuzioni si assomigliano, più i punti del Q-Q plot **si disporranno ordinatamente sulla retta di riferimento**.

Il modo in cui **non si dispongono** sulla retta dà informazioni sulla natura della distorsione: **asimmetria** destra o sinistra, **curtosi** positiva o negativa.

`qqplot (X, Y)` confronta la **forma di due distribuzioni campionarie.** Per aggiungere la retta di riferimento, si usa `abline(a=0, b=1)`, dove `a= 0` è **l'intercetta** (origine della retta) e `b= 1` il **coefficiente angolare** (variazione unitaria in Y al variare di una unità in X: ne parleremo diffusamente nella regressione.

```
qqplot(a$STAI_stato, a$STAI_tratto, pch=19, col=rainbow(15), xlab = "ansia di stato", ylab="ansia
  di tratto", main="Q-Q plot per due distribuzioni campionarie")
abline(0,1, lwd=2)
```

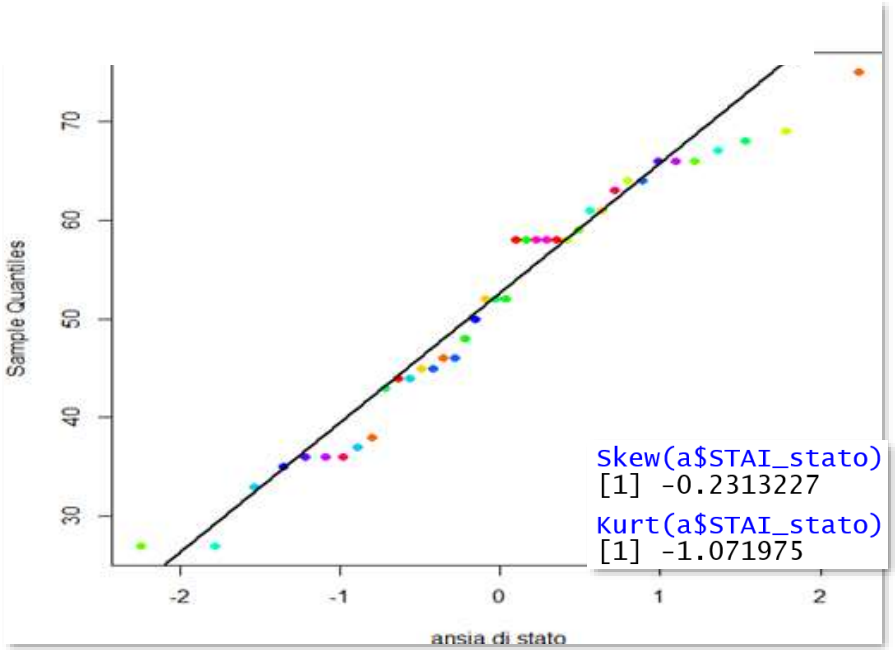
Ansia di stato e ansia di tratto non sembrano avere una forma molto simile, in effetti...



`qqnorm(X)` confronta la **forma di una distribuzione campionaria con la normale teorica (standardizzata: $\mu = 0, \sigma = 1$).** Per aggiungere la retta di riferimento: `qqline(X)`:

```
qqnorm(a$STAI_stato, pch=19, col=rainbow(15), xlab = "ansia di stato")
qqline(a$STAI_stato, lwd=2)
```

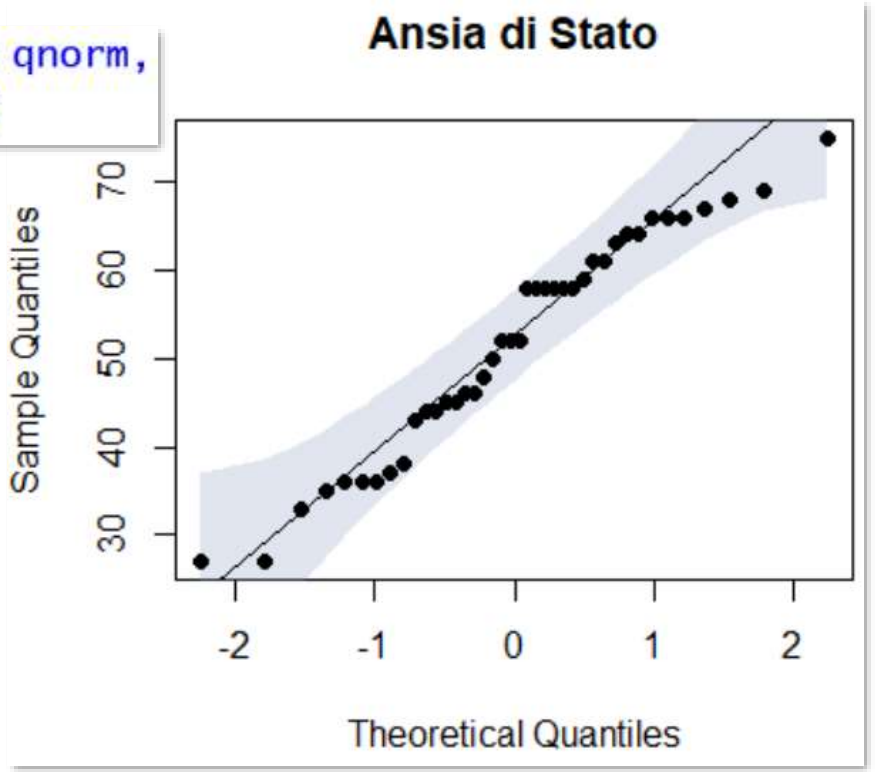
*Confermiamo che l'ansia di stato **non** sembra avere un andamento molto analogo alla normale*

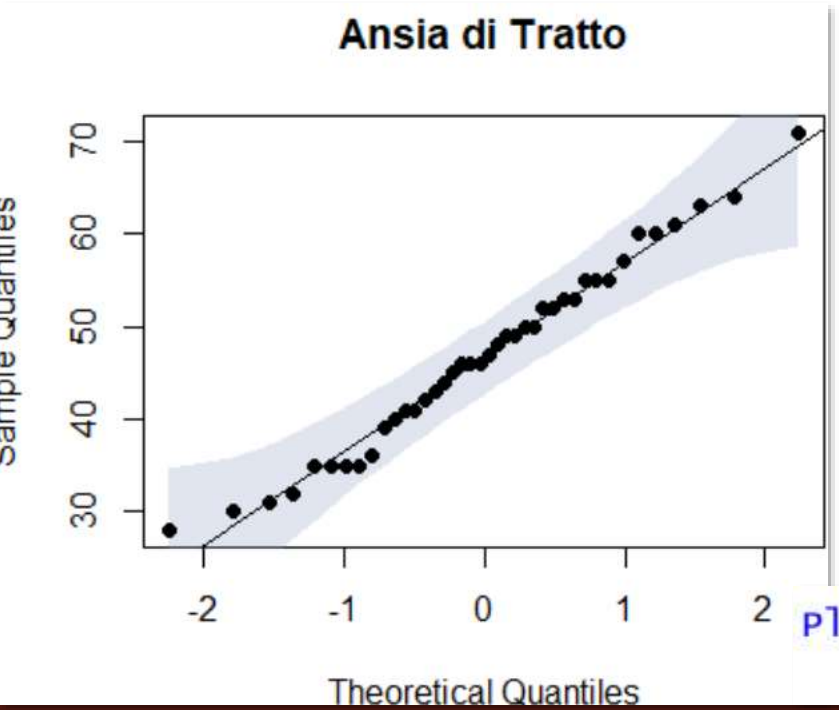


Andamento Q-Q plot	Forma della distribuzione
a U, concavità verso l'alto	Asimmetria positiva, a destra
A U rovesciato, concavità verso il basso	Asimmetria negativa, a sinistra
Parte iniziale sotto la linea, parte finale sopra la linea	Code lunghe, curtosi positiva
Parte iniziale sopra la linea, parte finale sotto la linea	Code corte, curtosi negativa

`PlotQQ(x=X, qdist= quantili della distribuzione teorica)` di `DescTools` aggiunge la linea di riferimento di default e produce la sua ***confidence band***: la vedremo bene nella regressione, per ora potete interpretarla come il range entro cui, in campionamenti ripetuti, la distribuzione delle coordinate si trova in popolazione, con una verosimiglianza prefissata. Per il confronto con la distribuzione normale, `qdist=qnorm`. (di default).

```
PlotQQ(a$STAI_tratto, qdist = qnorm,
pch=19, main="Ansia di Stato")
```






```
PlotQQ(a$STAI_tratto, qdist = qnorm,pch=19,
   main="Ansia di Tratto")
```

Correggere la forma della distribuzione:

trasformazioni non lineari

Una distribuzione non normale di per sé non è certo un problema per la descrizione: è un dato di realtà.

Però, **l'assunzione di normalità** è un prerequisito per l'applicazione di molti test inferenziali: test **parametrici**. Le conclusioni di un test parametrico sono affidabili se la distribuzione degli **errori** del modello segue determinati **parametri**, cioè se è **analoga alla normale**.

In caso di **violazione** dell'assunzione, possiamo **cambiare il test,** usando un test non parametrico (**robusto**)**: privilegeremo questo approccio** e vedremo diversi test robusti in Tecniche di analisi di dati II. In alternativa, possiamo **trasformare la distribuzione X** per renderla più "normale", sperando che questo normalizzi anche la distribuzione degli errori relativi a X, nonché per ridurre la varianza dell'errore. Le trasformazioni **non lineari cambiano la forma della distribuzione X** modificando ogni **elemento di X per quantità non costanti**.

Non possiamo comunque trascurare diverse obiezioni in merito (Games, 1984):

- **Teorema centrale del limite**: per grandi campioni la distribuzione campionaria tenderà a essere normale, quindi il **dibattito sulla necessità della trasformazione è realmente importante solo per campioni piccoli** (N > 40)

- Trasformando i dati **si cambiano le ipotesi che vengono testate**: per esempio, usando una trasformazione logaritmica su due distribuzioni e confrontandone le medie, staremmo confrontando medie geometriche, e non aritmetiche: l'interpretazione della differenza tra queste medie sarebbe ovviamente diversa (Gelman e Hill, 2007).

- In piccoli campioni è problematico stimare la normalità, **qualsiasi** modalità si usi.

- Le **conseguenze** sul modello statistico derivanti dall'applicazione di una **trasformazione inadeguata sarebbero peggiori** di quelle derivanti dall'analisi su dati non trasformati.

Le trasformazioni non lineari sono molte (*trasformazioni angolari, per proporzioni e percentuali: arcoseno, seno inverso, seno inverso iperbolico; tangente iperbolica, inversa per distribuzioni da* -1 *a* $+1$*; log-log e log-complementare per analisi di sopravvivenza et cetera*), oltre a quelle **più frequenti** che vediamo noi. Una volta scelto il tipo di trasformazione, deve essere applicato a **tutte** le variabili oggetto d'analisi.

Usiamo il metodo di **Box-Cox**, (Box e Cox, 1964) per individuare la migliore trasformazione non lineare per il tipo di variabile in analisi: valuta, con il metodo della **Maximum Likelihood** o Massima Verosimiglianza, la verosimiglianza (**log-likelihood**, $\boldsymbol{LL}$) di esponenti **lambda** λ (**da -5 a 5**), da applicare alla X da trasformare per ottenere la migliore normalizzazione possibile: $\boldsymbol{X_T = X^{\lambda}}$. λ è inserito in un **intervallo di fiducia** CI, all'interno del quale si sceglie il λ **intero più prossimo al** λ **lambda ottenuto**, corrispondente a una tra le trasformazioni più comuni.

Faremo la Massima Verosimiglianza in Tecniche di Analisi di dati II*, non preoccupatevi del calcolo. La funzione in R è molto semplice, si può usare senza approfondire il background teorico*

Riduzione dell'asimmetria positiva

Trasformazione lineare: la forma non cambia

Riduzione dell'asimmetria negativa

$\lambda = 0$ ***logaritmo***: $\log X$

logaritmo in base naturale di X, perché $X^0 = 1$

$\lambda < -3$ $X^{-\lambda}$

$\lambda = -3$ X^{-3} o ***reciproco*** $\frac{1}{X^3}$

$\lambda = -2$ $\frac{1}{X^2}$

$\lambda = -1$ $\frac{1}{X^1}$

$\lambda = -.5$ $\frac{1}{\sqrt[2]{X}}$

$\lambda = -.3$ $\frac{1}{\sqrt[3]{X}}$

$\lambda = .3$ $X^{\frac{1}{3}}$ o ***radice*** $\sqrt[3]{X}$

$\lambda = .5$ $X^{\frac{1}{2}}$ o $\sqrt[2]{X}$

$\lambda = 1$

$X^1 = X$

Non serve trasformare X, è sufficientemente normale

$\lambda = 2$ ***Esponente positivo***: X^2

$\lambda = 3$ X^3

$\lambda > 3$ X^{λ}

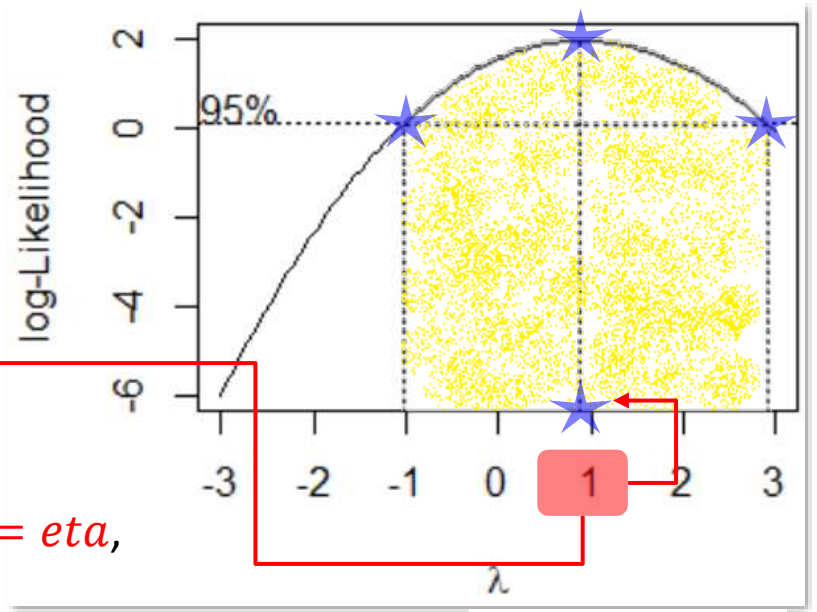
In R useremo `boxcox(variabile ~ 1, plotit=TRUE)` del package di base `MASS` che è stato scaricato durante l'installazione di R; dovete solo caricarlo nella sessione di lavoro:

```
library(MASS)
```

La **formula** `variabile ~1` indica che l'oggetto della funzione è un **modello lineare** $Y \sim X$, che per distribuzioni univariate contiene solo Y e **l'intercetta** (~1; modello nullo); ne parleremo nella regressione lineare. `plotit` (`=TRUE` di default) dà un grafico che mostra il **λ ottimale in un range di λ ugualmente verosimili** (intervallo di fiducia CI, in giallo). Il range dei λ messi alla prova (da -2 a +2 di default) si amplia con `lambda= seq(from= limite negativo, to= limite positivo, by=0.10)`, in cui `by` indica il passo del range dei λ.

Usiamo la **distribuzione dell'età** dei caregiver:

```
eta<- boxcox(a$eta~1, plotit=TRUE,
  lambda = seq(from=-3,to=3,by = 1/10))
```



Il λ (X) che ottimizza la funzione di verosimiglianza (Y) è ≅ **1** ($95\% CI[-1, 3]$, un po' ampio.

Quindi, la **trasformazione migliore** è $eta_T = eta^1 = eta$, cioè **non trasformare**!

In effetti, l'asimmetria di `a$eta` è già praticamente $= 0$

```
Skew(a$eta)
 [1] -0.0069
```

L'oggetto **eta** creato da **boxcox** è una lista:

```
class(eta)
[1] "list"
```

che contiene tutti i valori λ (eta$x) e i valori di verosimiglianza loro associati (eta$y).

```
round(head(eta$x,3),2); round(tail(eta$x, 3),2)
[1] -3.00 -2.94 -2.88
[1] 2.88 2.94 3.00
```

```
round(head(eta$y,3),2); round(tail(eta$y, 3),2)
[1] -6.00 -5.75 -5.50
[1]  0.18  0.07 -0.05
```

Possiamo usare queste informazioni per individuare il λ esatto per trasformare X: è la coordinata del λ in X che corrisponde alla massima verosimiglianza in Y. Usiamo **which.max**:

Tra i λ considerati dal test — *qual è il valore più grande* — *che corrisponde alla massima verosimiglianza di ottenere una distribuzione normale?*

```
eta$x  [which.max  (eta$y)]
[1] 0.8787879
```

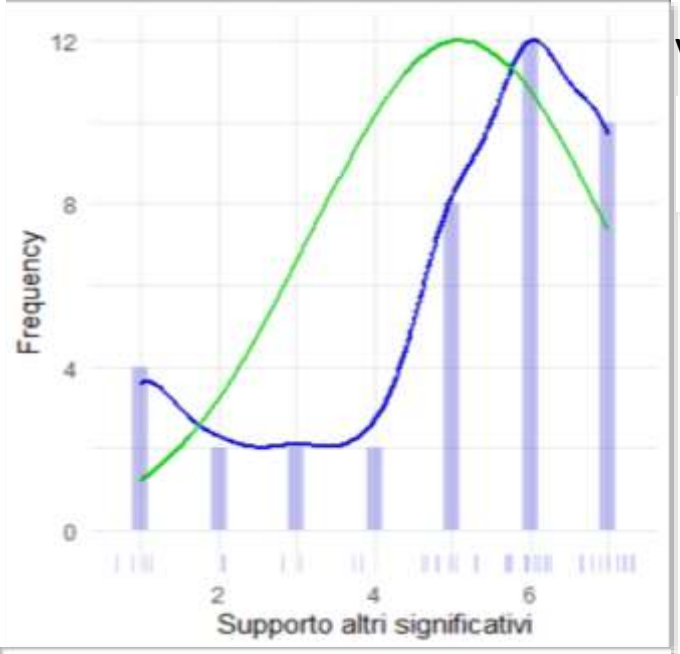
In effetti, la trasformazione $X_T = X^{.878}$ non migliorerebbe l'asimmetria di a$eta:

```
Skew(a$eta^.8787879)
[1] -0.04021103
```

```
Skew(a$eta)
 [1] -0.0069
```

In ogni caso, la trasformazione NON sempre funziona!

Bisogna sempre valutare l'effettiva approssimazione alla normalità di X_T

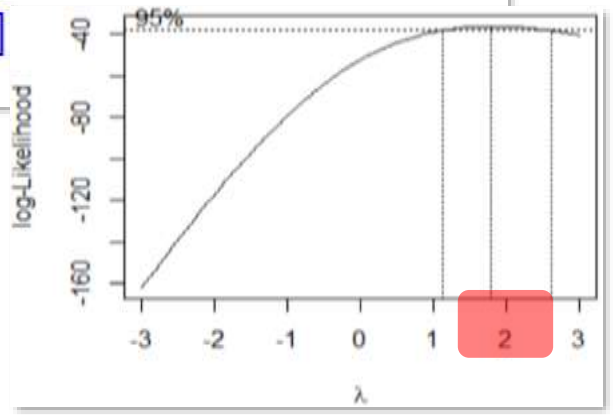


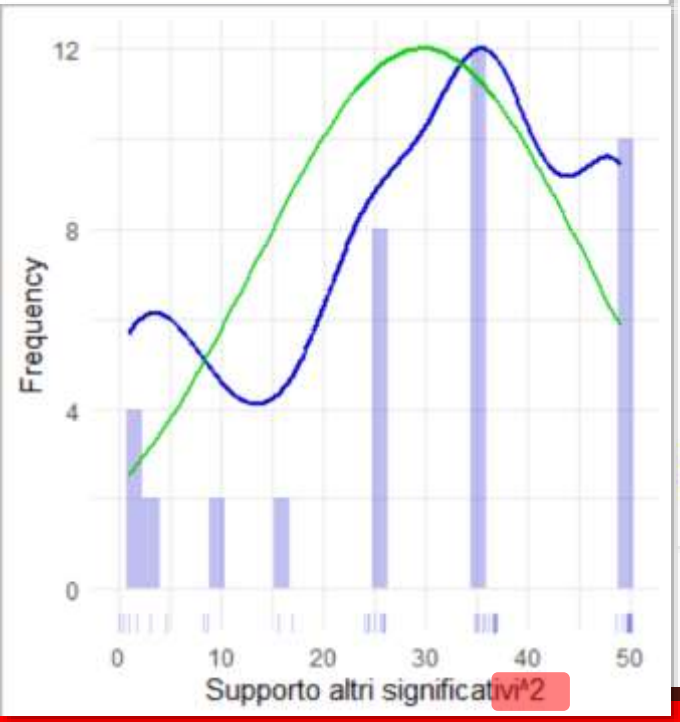
Vediamo un altro esempio: $zimet_altri_significativi:

```
Skew(a$Zimet_supporto_altri_significativi)
[1] -0.9876344
```

L'asimmetria è negativa, servirà un esponente positivo.

```
altri<-boxcox(a$Zimet_supporto_altri_significativi~1,
   lambda = seq(-3, 3, 1/10))
altri$x[which.max(altri$y)]
[1] 1.787879
```





Possiamo fare una **trasformazione esponenziale quadratica** (o usare come esponente $\lambda = 1.788$)

```
Skew(a$Zimet_supporto_altri_significativi^2, na.rm = TRUE)
[1] -0.4372951
```

Asimmetria ridotta, ma siamo lontani dall'ideale, soprattutto per la curtosi.

*Per fare qualche esempio delle diverse trasformazioni non lineari, usiamo **adolescenti**; scaricatelo da Elly e rinominatelo come **a**.*

*La ricerca era interessata alla percezione della gravità e alla **frequenza dei comportamenti a rischio**: in questi dati avete il **numero totale di comportamenti a rischio** per la salute (`a$comportamenti_rischio`) ammessi dai ragazzi.*

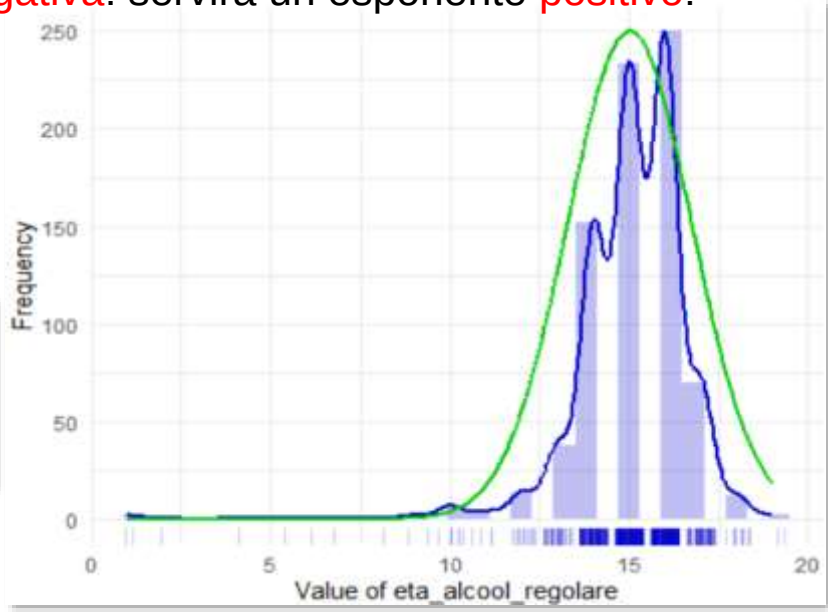
Prima di proseguire, descrivete i comportamenti a rischio dichiarati e interpretateli: anzitutto nel campione complessivo, poi separatamente per ragazzi e ragazze, infine separatamente per Istituto.

*Fate attenzione: è un campione molto **numeroso** (oltre 1000 casi), ma ci sono anche molti NA.*

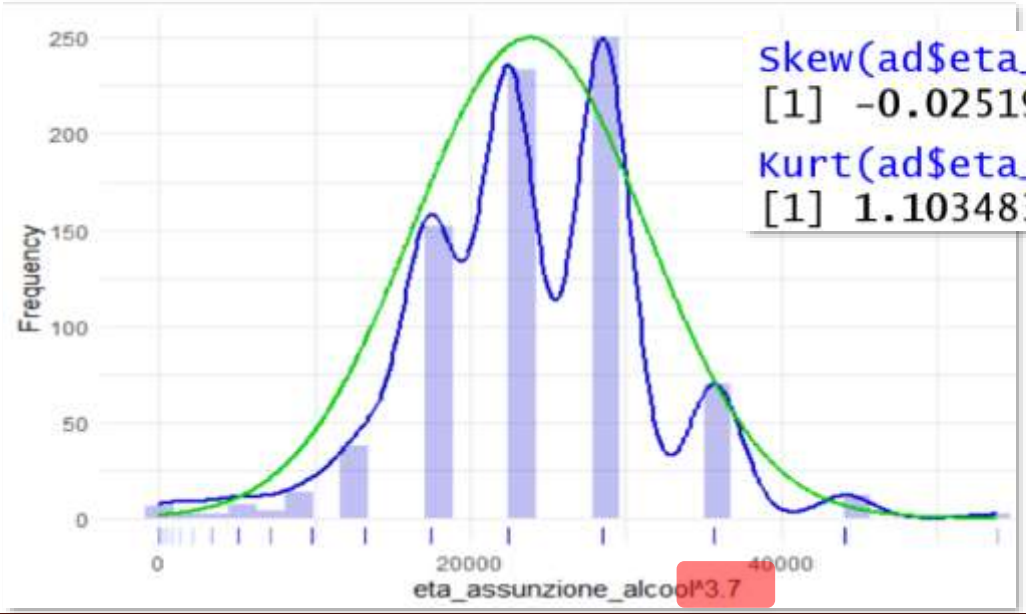
`$eta_alcool_regolare` ha una forte asimmetria negativa: servirà un esponente positivo:

```
Skew(ad$eta_alcool_regolare, na.rm = TRUE)
[1] -3.100543
Kurt(ad$eta_alcool_regolare, na.rm=TRUE)
[1] 18.74578

alcool<-boxcox(ad$eta_alcool_regolare~1, lambda = seq(-
  5,5,.10))
alcool$x[which.max(alcool$y)]
[1] 3.7
```

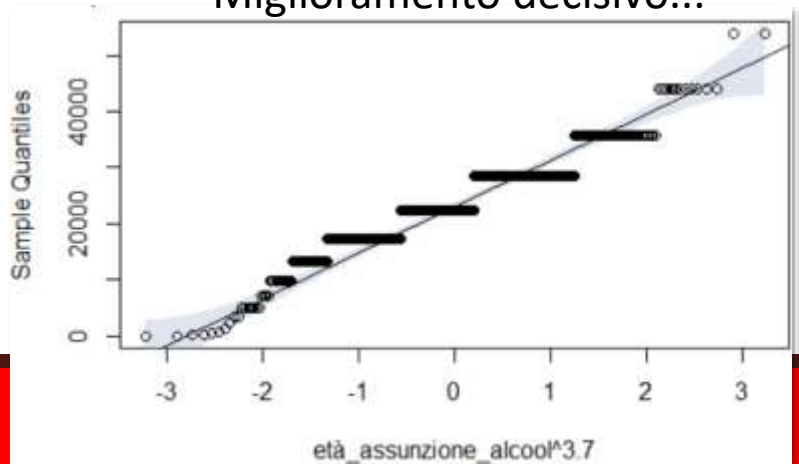


Facciamo una **trasformazione con esponente $= 3.7$**)

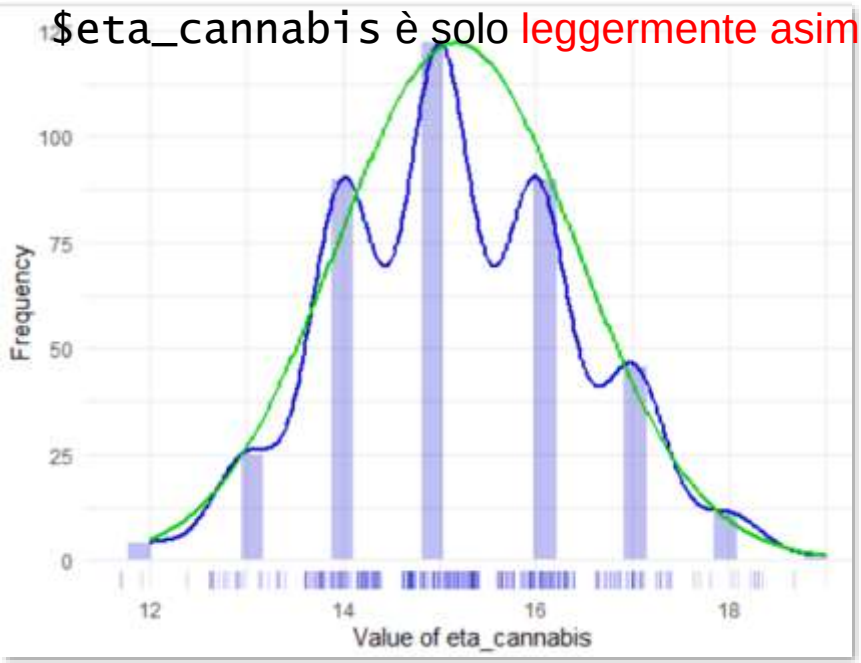



```
Skew(ad$eta_alcool_regolare^3.7, na.rm = TRUE)
[1] -0.02519031
Kurt(ad$eta_alcool_regolare^3.7, na.rm=TRUE)
[1] 1.103483
```

Miglioramento decisivo...



`$eta_cannabis` è solo leggermente asimmetrica positiva: servirà probabilmente una radice:

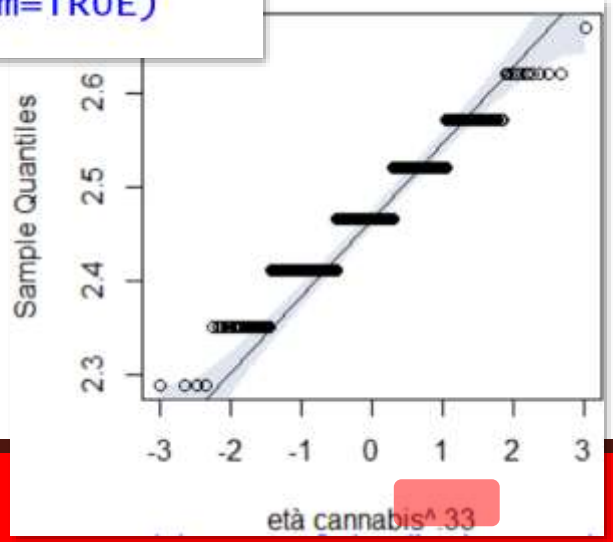



```
Skew(ad$eta_cannabis, na.rm = TRUE)
[1] 0.1420742
Kurt(ad$eta_cannabis, na.rm=TRUE)
[1] -0.2312118
```

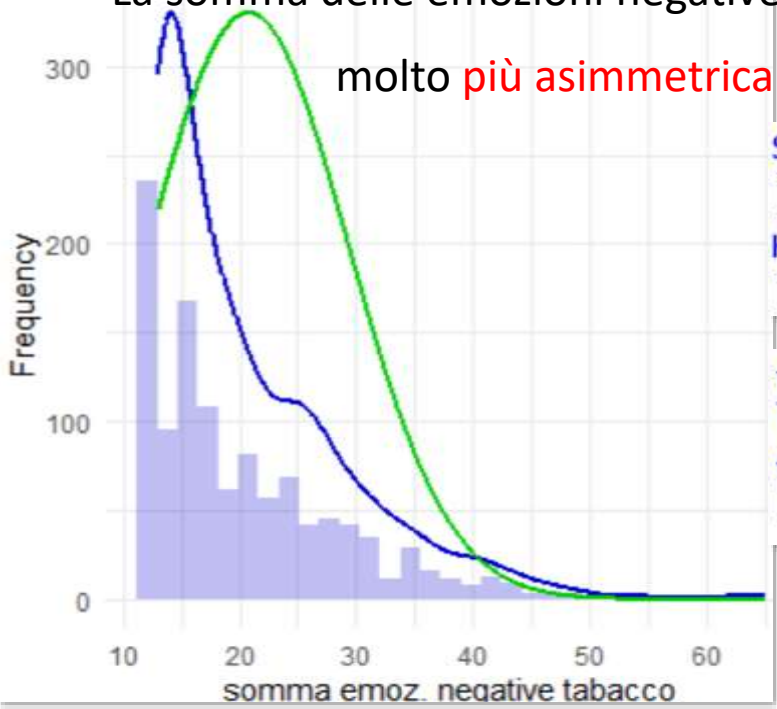
```
cannabis<-boxcox(ad$eta_cannabis~1, lambda = seq(-
  5,5,.10))
cannabis$x[which.max(cannabis$y)]
[1] 0.3
```

$\lambda = 0.3$: trasformazione in **radice terza**, ovvero $X_T = \sqrt[3]{X}$, che corrisponde a ***X* elevata a un terzo**: $X_T = \sqrt[3]{X} = X^{\frac{1}{3}} = X^{0.33}$


```
Skew(ad$eta_cannabis^(1/3), na.rm = TRUE)
[1] -0.00315167
Kurt(ad$eta_cannabis^.33, na.rm=TRUE)
[1] -0.2280347
```



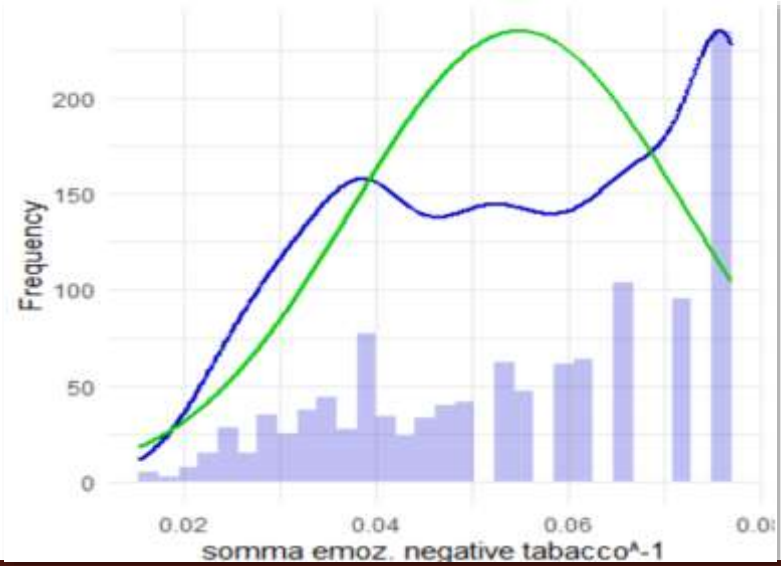
La somma delle emozioni negative associate all'uso di tabacco $somma_negative_fumo è molto più asimmetrica positiva dalla precedente, potrebbe servire un reciproco



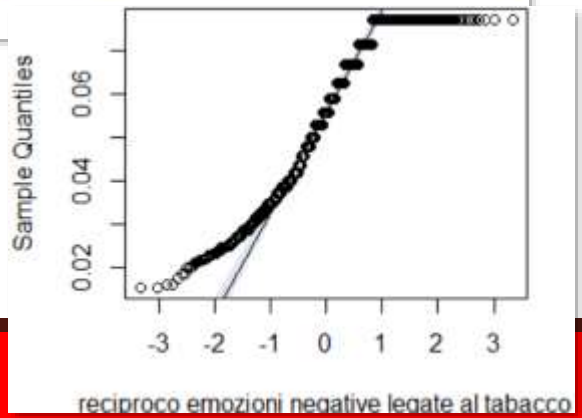

```
Skew(ad$somma_emozioni_negative_fumo, na.rm = TRUE)
[1] 1.546049
Kurt(ad$somma_emozioni_negative_fumo, na.rm=TRUE)
[1] 2.861756
```

```
fumo_nega<-boxcox(ad$somma_emozioni_negative_fumo~1,
lambda = seq(-5,5,.10))
fumo_nega$x[which.max(fumo_nega$y)]
[1] -1.1
```

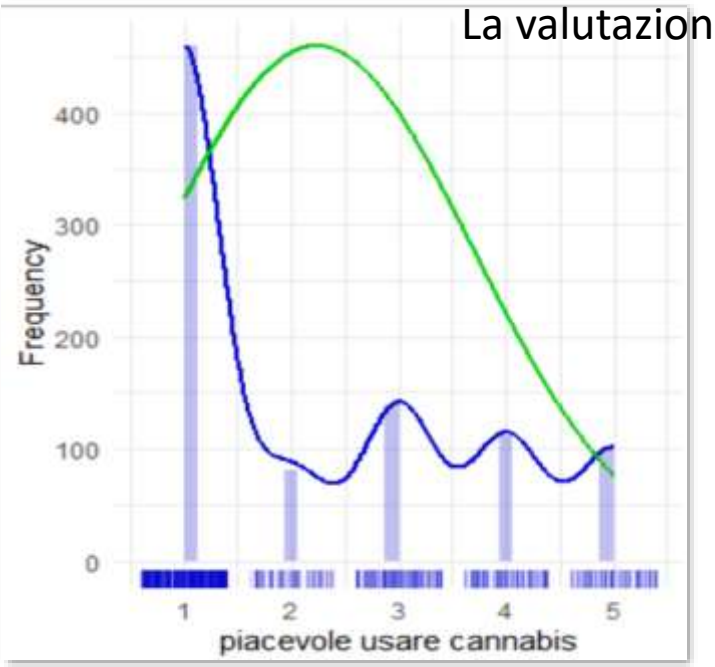
$\lambda = -1.1$: elevazione con esponente negativo $= -1$, $X_T = X^{-1}$, ovvero reciproco di X: $X_T = \boldsymbol{X^{-1}} = \frac{\mathbf{1}}{\boldsymbol{X^1}}$




```
Skew(ad$somma_emozioni_negative_fumo^(-1), na.rm =
  TRUE)
[1] -0.2303338
Skew(1/ad$somma_emozioni_negative_fumo, na.rm = TRUE)
[1] -0.2303338
```

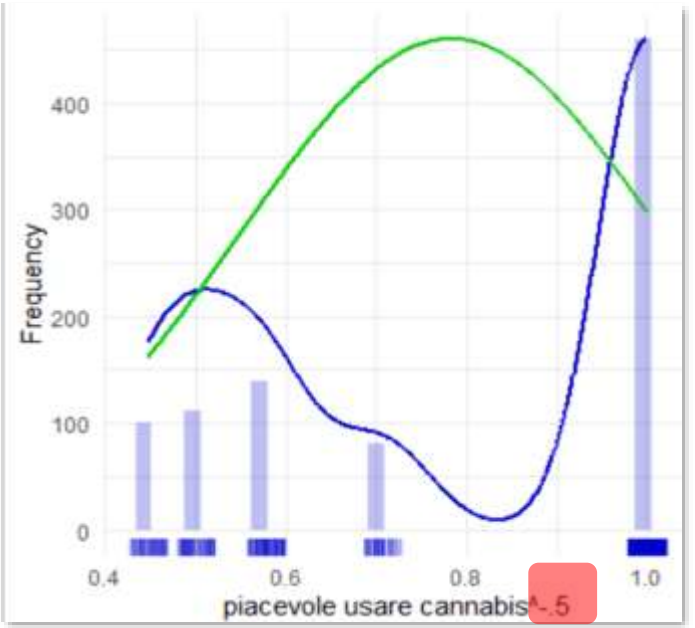


La valutazione single item (1-5) sulla piacevolezza di assumere cannabis è meno asimmetrica positiva, servirà un diverso reciproco



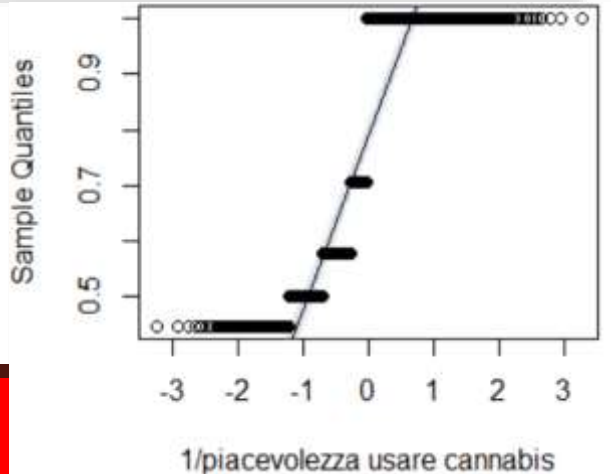

```
Skew(ad$picevole_cannabis, na.rm = TRUE)
[1] 0.7060653
Kurt(ad$picevole_cannabis, na.rm=TRUE)
[1] -1.003459
piacevole_cannabis<-boxcox(ad$picevole_cannabis~1,
lambda = seq(-5,5,.10))
piacevole_cannabis$x[which.max(piacevole_cannabis$y)]
[1] -0.6
```

$\lambda = -.6$: corrisponde a una elevazione con esponente negativo $= -0.5$, cioè il reciproco della radice quadrata di X: $X_T = \boldsymbol{X^{-.5}} = \frac{1}{\sqrt[2]{X}}$

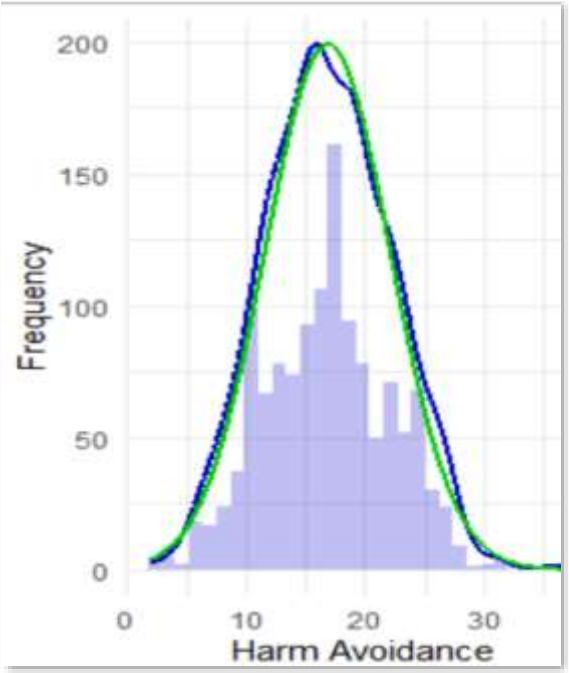



```
Skew(ad$picevole_cannabis^(-.5), na.rm = TRUE)
[1] -0.2500924
Skew(1/sqrt(ad$picevole_cannabis), na.rm = TRUE)
[1] -0.2500924
```

Miglioramento, ma non sufficiente...

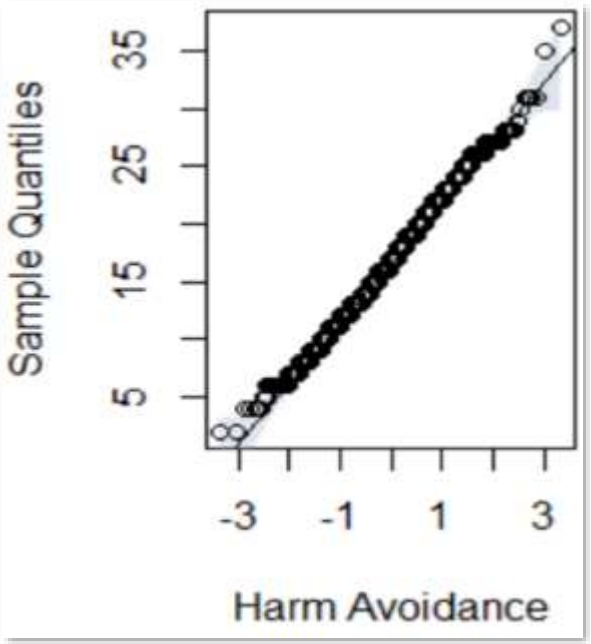


Il tratto **Harm Avoidance** – **HA**, come gli altri tratti di personalità, dovrebbe essere normalmente distribuito.



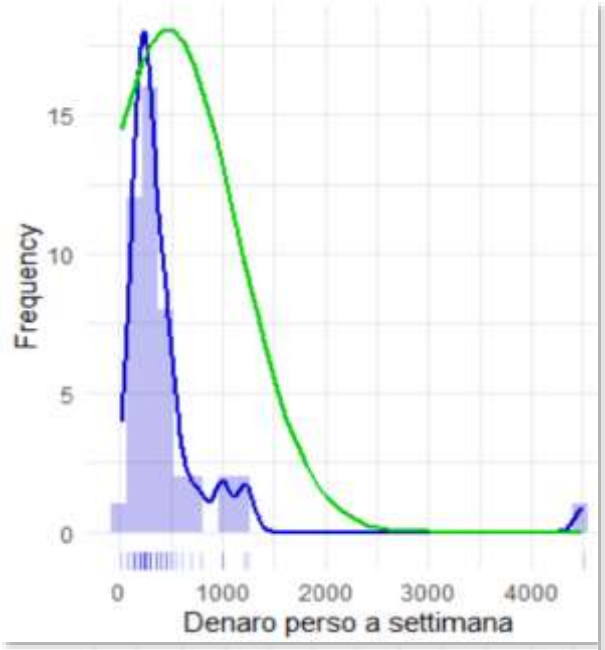

```
Skew(ad$HA_tot, na.rm = TRUE)
[1] 0.09166606
Kurt(ad$HA_tot, na.rm=TRUE)
[1] -0.2474728
```

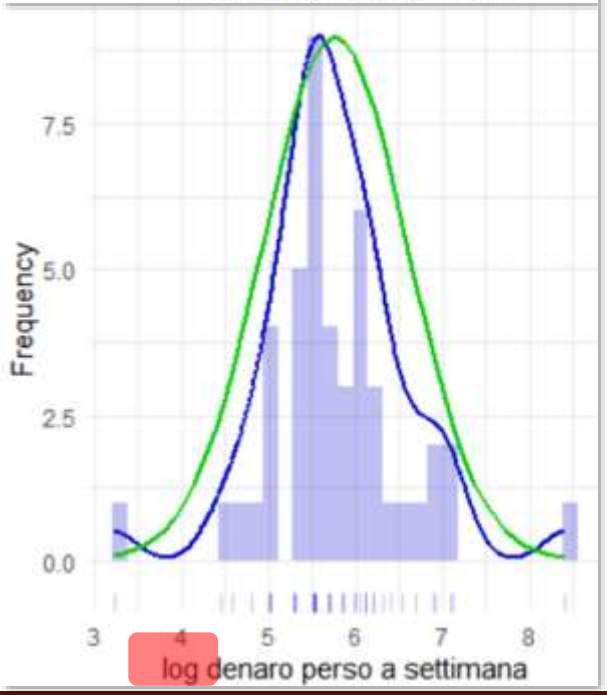
Simmetria quasi perfetta, curtosi buona; non dovremmo aver bisogno di trasformare




```
HA<-boxcox(ad$HA_tot~1, lambda = seq(-5,5,.10))
HA$x[which.max(HA$y)]
[1] 0.9
```

$\lambda = .9$ corrisponde a una elevazione di X alla prima, cioè alla trasformazione di X in se stessa: $X_T = X^1 = X$.





Nel dataframe `gamblers` troviamo pazienti in trattamento presso un SeRT per dipendenza da gioco d'azzardo (gambling). Ci concentriamo sulla quantità di **denaro che i pazienti dichiarano di aver perso alla settimana**, in media.

```
Skew(gamblers$euro_a_settimana, na.rm = TRUE)
[1] 4.807489
Kurt(gamblers$euro_a_settimana, na.rm=TRUE)
[1] 25.81551
```

Fortissima asimmetria positiva: servirà un logaritmo?

```
euro<-boxcox(gamblers$euro_a_settimana~1,lambda=seq(-5,5,.10))
euro$x[which.max(euro$y)]
[1] 0
```

$\lambda = 0$ implica la trasformazione in logaritmo di X: $X_T = \log X$.

Usiamo `log(x, base=)`, in cui non serve specificare la base (di default, base naturale

```
Skew(log(gamblers$euro_a_settimana), na.rm = TRUE)
[1] 0.2106762
Kurt(log(gamblers$euro_a_settimana), na.rm=TRUE)
[1] 2.121095
```

Ottimo miglioramento!

ATTENZIONE: non è possibile calcolare il reciproco di zero, la radice di numeri negativi e il logaritmo di zero e di numeri negativi

```
1/0
[1] Inf
```

```
sqrt(-10)
[1] NaN
(-10)^.33
[1] NaN
```

```
log(0)
[1] -Inf
log(-10)
[1] NaN
```

Se la distribuzione grezza contiene valori $= 0$ o negativi incompatibili con la trasformazione non lineare cui si deve sottoporre, dovremo **trasformare linearmente X, aggiungendo una costante a tutti i suoi valori** (di solito $+.5$ o $+1$), **prima di procedere alla trasformazione non lineare.**

$$X_T = \frac{1}{(X+1)}$$

$$X_T = \sqrt[n]{(X+1)}$$

$$X_T = \log(X+1)$$

Traslare la distribuzione: trasformazioni lineari

La standardizzazione

Come appena visto, può essere utile sommare (o sottrarre) un valore **costante** a tutti i dati della distribuzione: un'importantissima forma di **traslazione**, che **oltre a spostare il centro della distribuzione ne cambia anche l'unità di misura**, è la **standardizzazione**.

Standardizzare una variabile significa trasformarla in modo tale che, qualunque siano l'unità di misura e il range dei punteggi **grezzi**, la distribuzione trasformata avrà $\boldsymbol{media = 0}$ e $\boldsymbol{sd = 1}$.

Prima si **centrano i dati attorno a zero** sottraendo a ciascuno la media della distribuzione → la **media** della nuova distribuzione sarà **=0**. Poi, ogni dato è diviso per la deviazione standard → la **deviazione standard** della nuova distribuzione sarà **= 1**: la **nuova unità di misura della variabile standardizzata è quindi la deviazione standard**.

I dati trasformati sono chiamati **punteggi z (z scores)**:

$$z = \frac{x_i - \bar{X}}{s}$$

In R è facile: prima vediamo i passaggi del calcolo, poi la funzione dedicata. Standardizziamo la distribuzione **dell'ansia di stato**

```
> summary(attaccamento$STAI_stato); sd(attaccamento$STAI_stato)
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  27.00   43.75   52.00   51.72   61.50   75.00
[1] 12.61661
```

Prima **centriamo** la variabile sulla media sottraendola a ogni dato

```
> ansia_z<-attaccamento$STAI_stato-mean(attaccamento$STAI_stato)
> round(mean(ansia_z),2);sd(ansia_z)
[1] 0
[1] 12.61661
```

La media è = 0, ma la sd è ancora quella grezza; ora cambiamo l'unità di misura **rapportando i dati centrati alla deviazione standard:**

```
> ansia_z<-ansia_z/sd(attaccamento$STAI_stato)
> round(mean(ansia_z),2);sd(ansia_z)
[1] 0
[1] 1
```

Con `scale(x=distribuzione, center=TRUE, scale=TRUE)` è più veloce:

```
>ansia_z<-scale(attaccamento$STAI_stato)
```

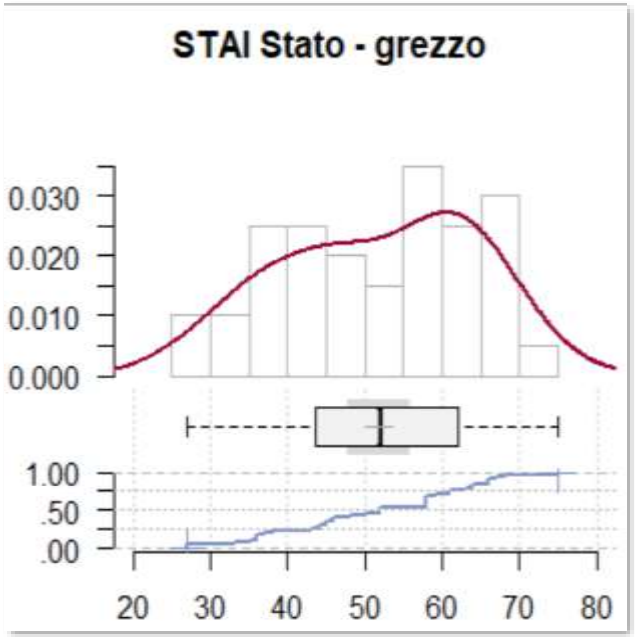
```
> Desc(attaccamento$STAI_stato, digits = 2, main="STAI Stato - grezzo")
```

STAI Stato - grezzo

length	n	NAs	unique	0s	mean	meanCI'
40	40	0	23	0	51.73	47.69
	100.0%	0.0%		0.0%		55.76

.05	.10	.25	median	.75	.90	.95
32.70	35.90	43.75	52.00	61.50	66.10	68.05

range	sd	vcoef	mad	IQR	skew	kurt
48.00	12.62	0.24	13.34	17.75	-0.23	-1.07



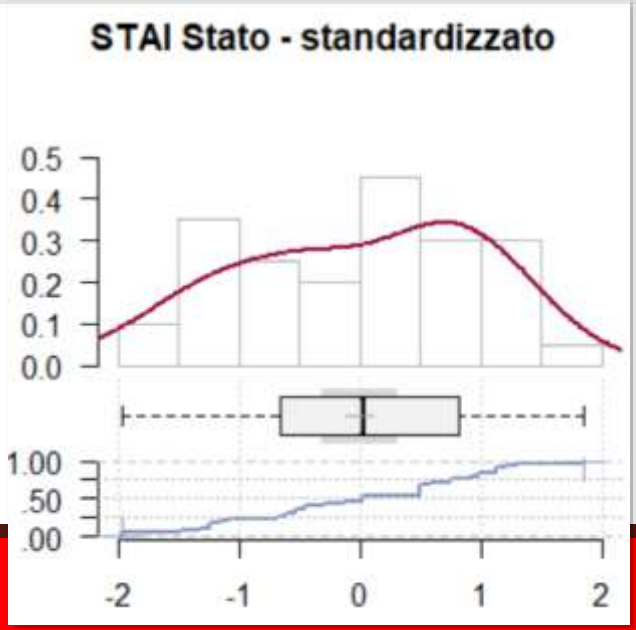

```
> Desc(ansia_z, digits = 2, main= "STAI Stato - standardizzato")
```

ansia_z (numeric)

length	n	NAs	unique	0s	mean	meanCI'
40	40	0	23	0	0.00	-0.32
	100.0%	0.0%		0.0%		0.32

.05	.10	.25	median	.75	.90	.95
-1.51	-1.25	-0.63	0.02	0.77	1.14	1.29

range	sd	vcoef	mad	IQR	skew	kurt
3.80	1.00	-1.64e+16	1.06	1.41	-0.23	-1.07



La standardizzazione ha cambiato la scala dei dati

lasciando inalterata la forma della distribuzione.

Gli outliers

La trasformazione in punti *z* rende molto facile individuare i soggetti **outlier univariati**:

Sono **outlier univariati** i soggetti il cui punteggio **si discosta di almeno** $|2|sd$ **dalla media**, e quindi i soggetti con $z \geq 2$ o $z \leq -2$.

Se la distribuzione segue un andamento normale, l'outlier cadrà nel 2.5% superiore o inferiore della totalità dei casi.

L'individuazione degli outlier è un passaggio importante della descrizione dei dati, sia dal punto **interpretativo**, sia per **la valutazione del fit della distribuzione**. Togliere gli outlier può **migliorare l'adeguamento della distribuzione alla normale** attesa (il caso outlier determina una coda "anomala") **e il fit del modello ai dati**: gli **outlier rappresentano i casi per cui il modello compie gli errori più grandi**, perciò, eliminandoli, la capacità del modello di descrivere la realtà dovrebbe migliorare.

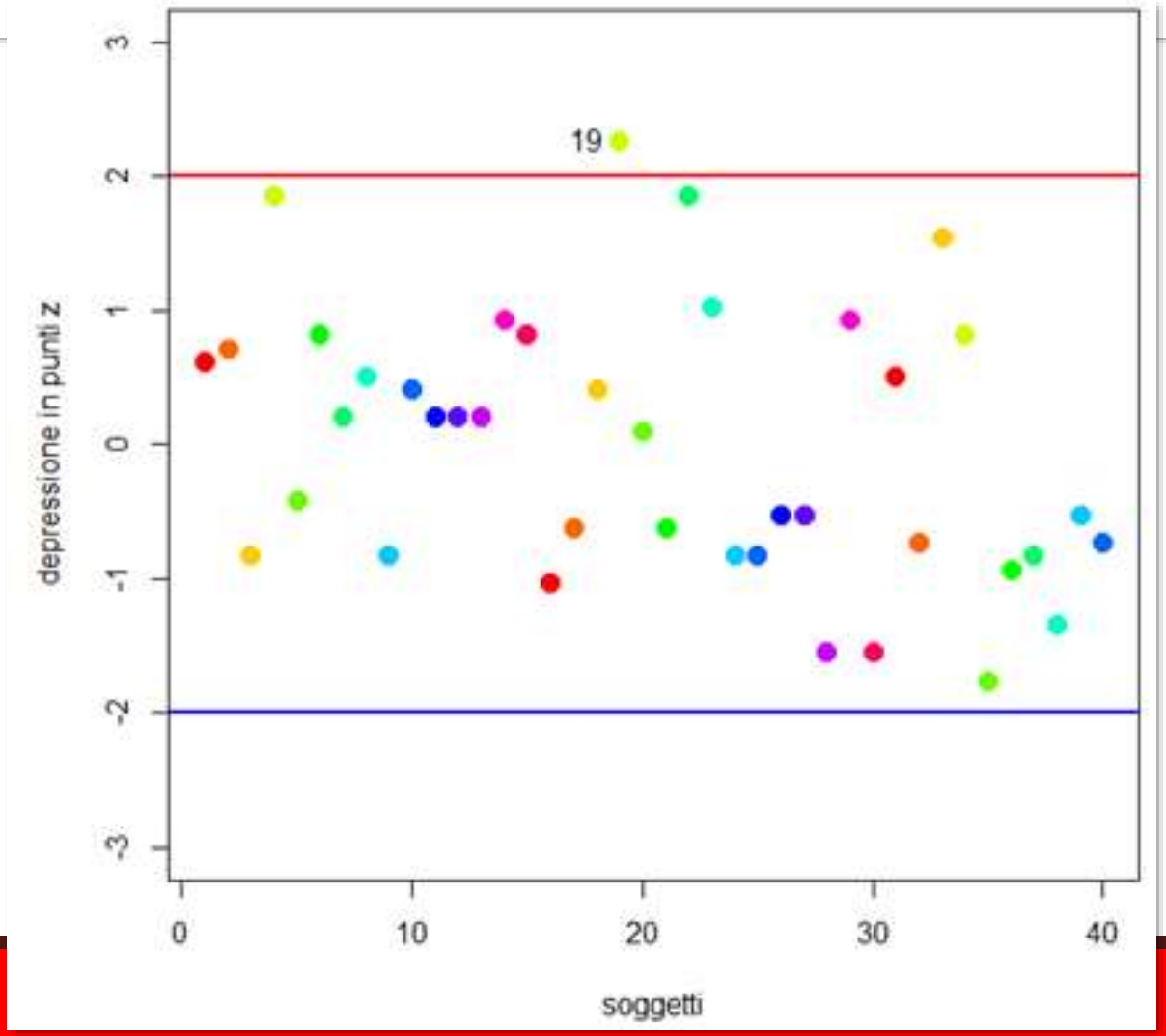
Gestiremo nella regressione gli outlier **bivariati.**

Vediamo la distribuzione z dei **punteggi al test BDI II** sulla depressione:

```
depressione_z<-scale(attaccamento$BDI_II_depressione)
```

```
> plot(depressione_z, col=rainbow(15), pch=19, cex=1.5, ylim=c(-3,3), xlab="soggetti",
ylab="depressione in punti z")
> abline(h=2, col="red", lwd=2)
> abline(h=-2, col="blue", lwd=2)
> identify(depressione_z)
```

Il soggetto 19 ha un punteggio di depressione >2ds dalla media: è, quindi, probabilmente un **outlier**, e **necessita di attenzione clinica**.

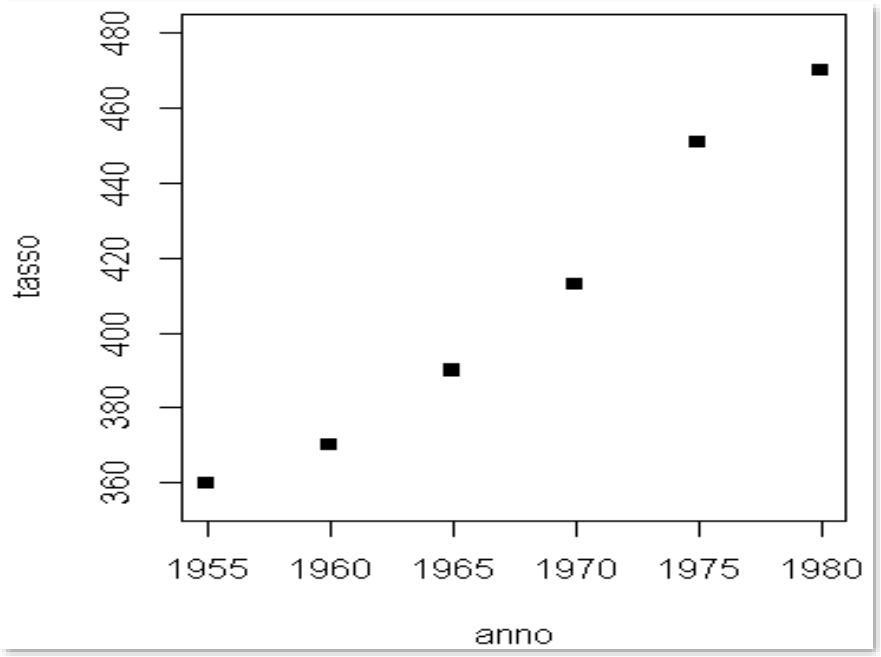


Bias nelle rappresentazioni grafiche

Proprio la chiarezza dei grafici può portare l'incauto lettore grafici a **equivoci interpretativi**.

Quando la **scala dell'ordinata Y è eccessivamente compressa**, l'informazione del grafico viene distorta.

Everitt (2001): tassi di mortalità per cancro al seno registrati dagli anni Cinquanta agli Anni Settanta negli Stati Uniti. **se non specifichiamo alcun limite ai valori dell'ordinata**, il tasso esprime un'ascesa vertiginosa.




```
>anno<-c(1955,1960,1965,1970, 1975, 1980)
>tasso<-c(360, 370, 390 ,413, 451, 470)
>plot(anno, tasso, ylim=c(355,480), pch=15)
```

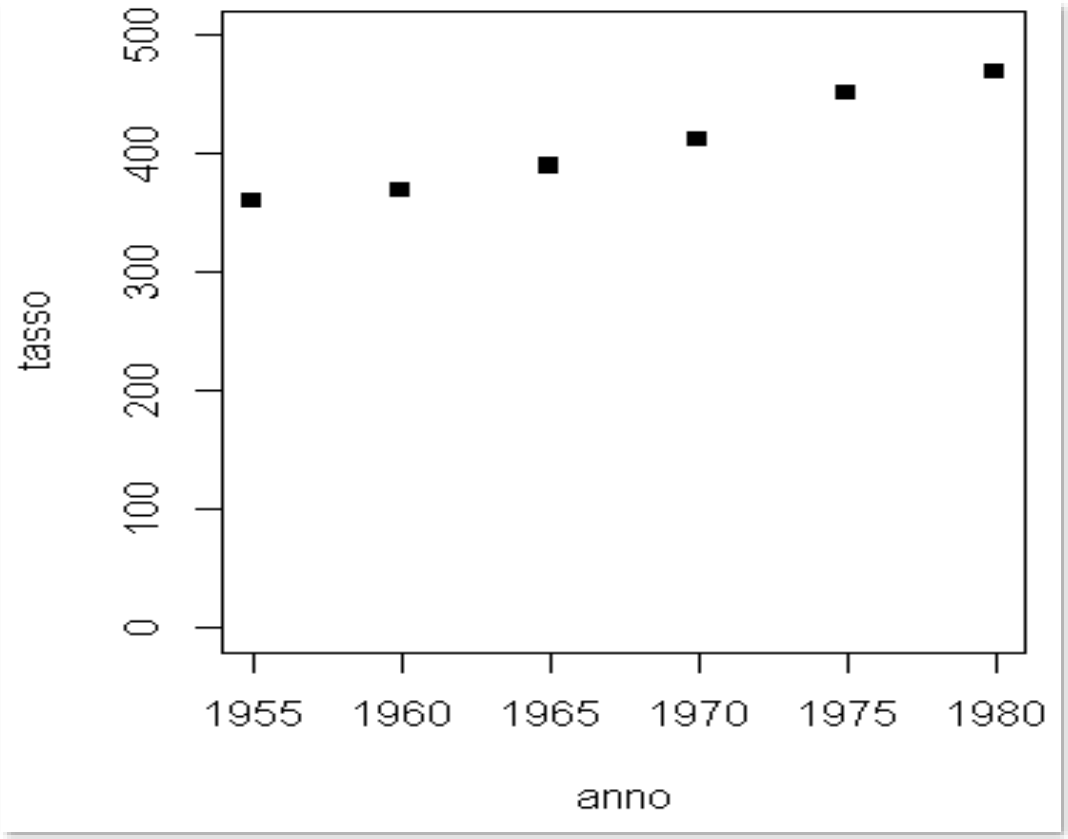
I limiti di Y calcolati da R sono 360 e 480: il range è quindi pari a 120.

Cambiamo i limiti di Y usando `ylim=c(limite inferiore, limite superiore)`:

facciamo **partire il grafico da 0**, quindi **allargando il range di Y**:

```
plot(anno, tasso, ylim=c(0,500), connect=TRUE, pch=15)
```

Le conclusioni del lettore sarebbero diverse, no?



Vedremo altri esempi di distorsioni quando affronteremo i grafici per distribuzioni bivariate