2. Statistiche e modelli univariati

TECNICHE DI ANALISI DI DATI I

"Le statistiche sostengono che più tempo si passa in automobile sulle strade e più aumenta la probabilità di incidenti; la prudenza consiglia quindi di possedere un'auto molto veloce e di correre a tavoletta"

"Ci sono tre professori di Statistica che vanno a caccia di lepri. Ad un tratto ne vedono una. Il primo spara ... un metro a destra. Il secondo spara ... un metro a sinistra. Il terzo esclama: 'L'abbiamo presa!'"

"I numeri sono come le persone: torturali abbastanza ed essi ti diranno qualsiasi cosa"

La statistica

Una delle possibili definizioni (Wilcoxon, nel 1935, ne conta 115) la descrive come: "[...] la disciplina che elabora i principi e le metodologie che presiedono al processo di rilevazione e **raccolta dei dati**, alla **rappresentazione sintetica** e alla **interpretazione** dei dati stessi e, **laddove ve ne siano le condizioni**, alla **generalizzazione** delle evidenze osservate.

Non ci occuperemo in questo esame di **raccogliere dati**: useremo i dataframe a disposizione su Elly, composti da **unità statistiche** (**caso**, **soggetto**), che compongono il **collettivo statistico** (**campione**) su cui sono stati rilevati **caratteri** (aspetto elementare oggetto di rilevazione; **variabile**), ciascuno dei quali presente in diverse **modalità**. Di questi dati faremo rappresentazioni sintetiche, ovvero **modelli**, li **interpreteremo** e , dove possibile, li generalizzeremo alla popolazione da cui sono tratti,

Unità statistiche e collettivi

Unità statistica: il **caso individuale** componente del *collettivo statistico* (**campione**). In psicologia sono spesso definiti ***soggetti*** o ***partecipanti***.

	sogg	genere	eta	stato_civile	istruzione
1	S1	F	23	single	diploma superiore
2	S2	M	21	single	laurea
3	S3	F	68	coniugato	laurea
4	S4	M	18	single	diploma superiore
5	S5	F	23	single	diploma superiore
6	S6	M	68	coniugato	laurea
7	S7	F	55	coniugato	laurea

Caratteri: si chiama carattere ogni **aspetto elementare oggetto di rilevazione** nelle unità statistiche. Spesso si usa il termine ***variabile***, in senso generale.

La Statistica **usa numeri e formule per descrivere la realtà**, seguendo **regole rigorose** nel passare dalla realtà ai numeri: si usa il sistema numerico come se avesse le stesse caratteristiche del sistema empirico che deve rappresentare → **omomorfismo**.

I numeri descrivono **qualità** o **quantità**: le regole di trasposizione dal sistema empirico al sistema numerico definiscono la **scala di misura** del dato

Misura ed errore di misura

La misurazione consiste nell'applicazione della matematica a eventi

---

Misura ciò che è misurabile, e rendi misurabile ciò che non lo è.
Galileo Galilei

[…] quando puoi misurare quello di cui stai parlando, ed esprimerlo in numeri, allora puoi dire di conoscere qualcosa su di lui; ma quando non puoi misurarlo, quando non puoi esprimerlo in numeri, la tua conoscenza è scarsa è insoddisfacente: può forse essere l'inizio della conoscenza, ma sei consapevole di essere avanzato ben poco sul piano della scienza, qualunque possa essere la materia
Kelvin, 1891

Misurare non necessariamente significa progresso. Se fosse impossibile misurare quello che si desidera, l'avidità di misurare potrebbe, per esempio, semplicemente sfociare nel misurare qualcos'altro – e forse nel dimenticare la differenza -, o nell'ignorare qualcosa solo perché non può essere misurato.
Yule, 1921

La misurazione

Non si identifica *tout court* con la scienza, caratterizzata dall'uso di osservazioni **sistematiche** e **controllate**, e dal tentativo di **falsificarle**. In psicologia, la combinazione di disegno sperimentale e metodo statistico introdotta da Fisher ha reso perlopiù possibile la combinazione di controllo sperimentale e controllo statistico.

Usiamo numeri per **designare oggetti ed eventi e la relazione tra loro**. Gli oggetti possono essere **concreti** o **non tangibili** (l'intelligenza, la personalità, l'autostima) → le misure sono **operazionalizzazioni**, descrizioni del compito che riflettono il costrutto. In ogni caso, **la misura fornisce descrizioni precise, economiche e facilmente comunicabili**.

Tutte le misurazioni incontrano difficoltà pratiche → **errore di misura**, che aumenta la variabilità nei dati, diminuendo la precisione delle statistiche descrittive e delle inferenze.

Quetelet e Galton: la distribuzione degli errori è una **legge** di **natura**; **non può esistere un'assoluta accuratezza nella misurazione**, ma solo un **giudizio di accuratezza**, nei termini dell'intrinseca **variazione entro e tra gli individui**.

Le statistiche sono gli strumenti per valutare le proprietà di queste fluttuazioni casuali.

Scale di misura

Stevens (1946): “Possiamo dire che la misurazione, nel senso più ampio, consiste nell’attribuzione di numeri a oggetti o eventi seguendo determinate regole. Il fatto che si possano **assegnare dei numeri seguendo regole differenti** porta a **differenti tipi di scala e livelli di misurazione**”.

Lord (1953), **tra gli altri**, non è d’accordo:

> *"Since the numbers don’t remember where they came from, they always behave just the same way, regardless"*

Ogni scala o livello di misura mantiene le proprietà del livello precedente, aggiungendovi caratteristiche peculiari.

Posso dire se due elementi di una distribuzione sono uguali o diversi?

no → **Beh, allora dovrei chiedermi se sto misurando qualcosa**

sì

Posso ordinare fra loro due elementi diversi di una distribuzione?

no → **Scala nominale**

Equivalenza e non equivalenza simmetrica ($A = B \rightarrow B = A$; $A \neq B \rightarrow B \neq A$), transitività (se $A = B$ e $B = C \rightarrow A = C$)

sì

Posso calcolare una differenza fra due elementi di una distribuzione riconducibile a un'unità di misura?

no → **Scala ordinale**

Anche principio d'ordine (se $A < B \rightarrow B > A$; se $A > B$ e $B > C \rightarrow A > C$)

sì

Esiste uno zero assoluto?

no → **Scala a intervalli equivalenti**

Costanza del rapporto tra intervalli

Differenze tra valori equivalenti:

$$6 - 4 = 10 - 8$$

sì → **Scala a rapporti equivalenti**

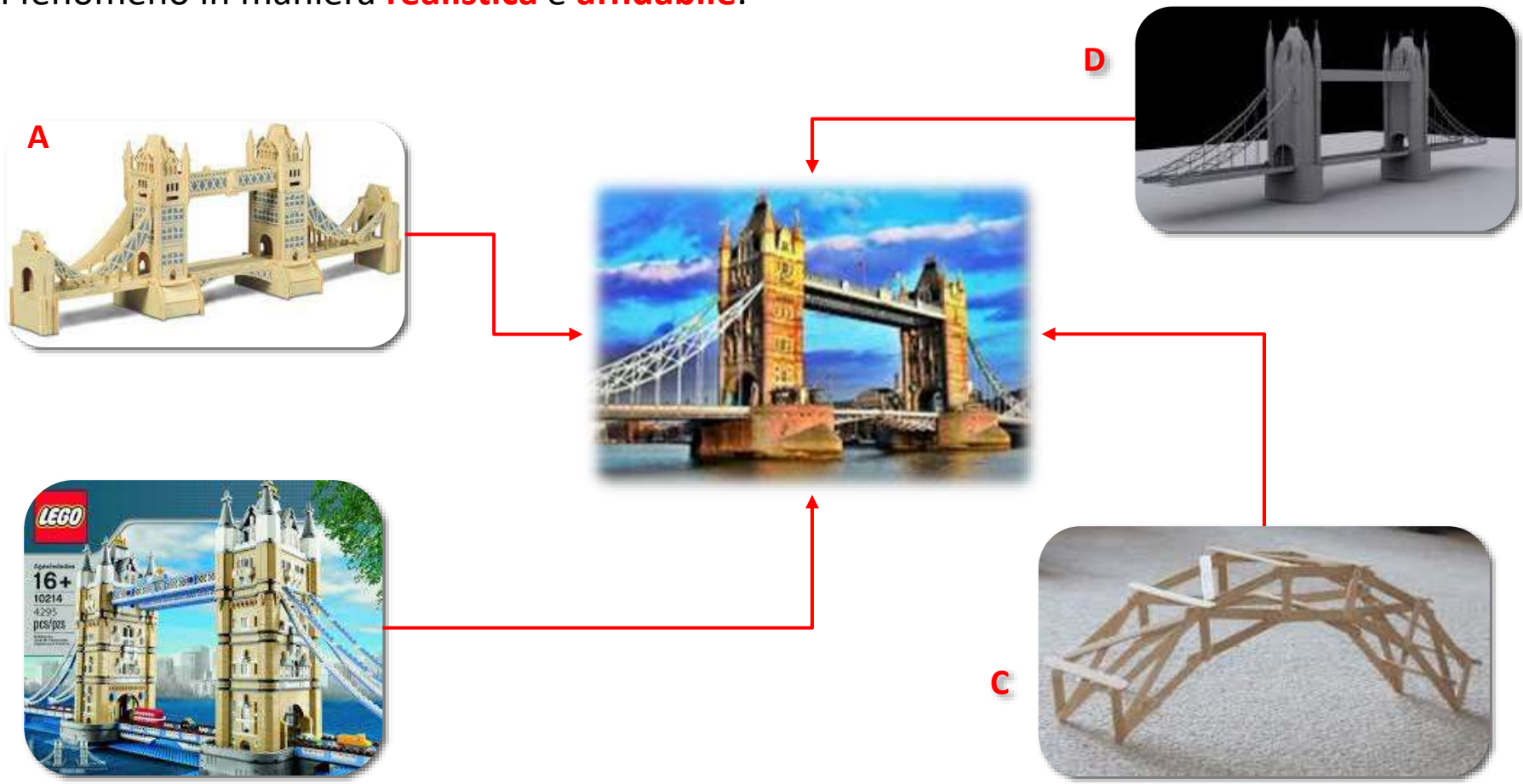
Anche costanza del rapporto tra valori:

$$20 / 10 = 10 / 5$$

Il modello

“The hallmark of good science is that it uses models and “theory”, but never believes them"

Un **modello** è una **riproduzione** di un qualsivoglia fenomeno, che consente di replicare in **scala**, risparmiando tempo ed energia, il fenomeno stesso e di valutare ipotesi e **previsioni** sul fenomeno in maniera **realistica** e **affidabile**.



C

D

I quattro modelli di ponti sono più o meno **affidabili** nella loro capacità di riflettere le caratteristiche del ponte reale: se volessimo **stimare** la capacità di resistenza al vento del London Bridge usando simulazioni con i ponti C e D e costruissimo il London Bridge usando tali stime, provocheremmo una strage di massa al primo temporale.

I due modelli **non si adattano** al fenomeno reale

I **modelli non** hanno un buon **fit**

Le **previsioni** sul fenomeno oggetto di studio basate su questi modelli **non sono affidabili**

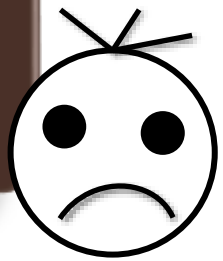
Oltre ai modellini con i Lego, la scienza usa modelli statistici per riprodurre il mondo reale in scala

Se i **modelli statistici si adattano** al fenomeno reale

i **modelli hanno** un buon **fit**

Le **previsioni** sul mondo reale basate su questi modelli **sono affidabili**

Se i **modelli statistici non si adattano** al fenomeno reale



i **modelli non** hanno un buon **fit**

Le **previsioni** sul mondo reale basate su questi modelli **non sono affidabili**

La goodness of fit di un modello è la sua capacità, quantificabile, di riprodurre il più semplicemente e fedelmente possibile un dato reale (di solito complesso).

Più precisamente:

Un modello statistico è una **funzione delle variabili esplicative X** (variabili indipendenti, predittori) il cui scopo è quello di **spiegare il meglio possibile la variazione nella variabile dipendente Y** (risposta).

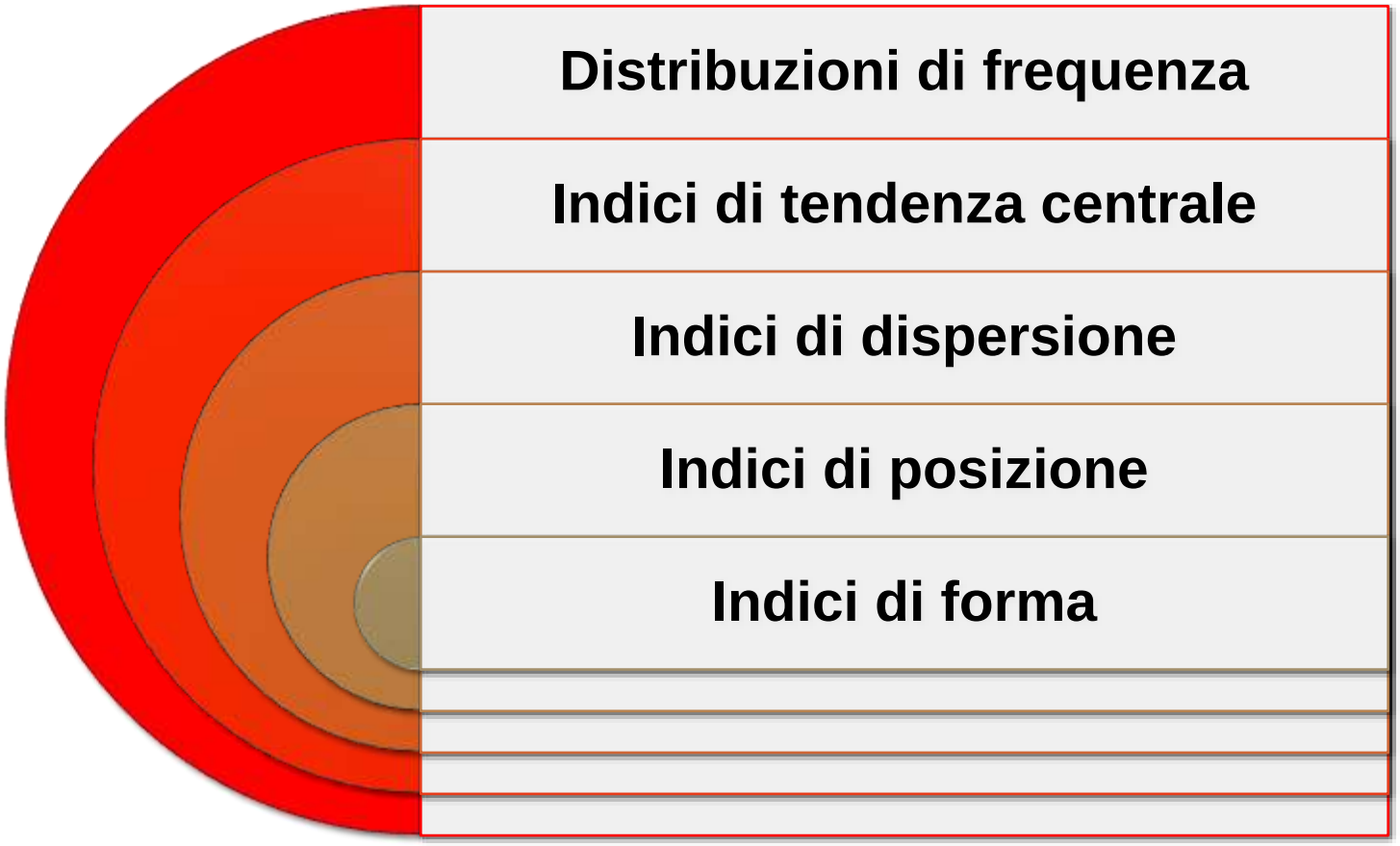
Il nostro obiettivo sarà quello di **individuare i valori dei parametri della funzione** – modello che portano alla **migliore goodness of fit** del modello ai dati.

Il **miglior modello** è quello che lascia la **minima quantità di variabilità di Y non spiegata dalle variabili esplicative**…

… usando il minor numero possibile di variabili esplicative (**principio di parsimonia**)

Modelli univariati

Iniziamo con i **modelli statistici più semplici**, che **rappresentano sinteticamente** caratteristiche di **una sola distribuzione**, differenziandoli per scala di misura e scopo:



Gli esempi e gli esercizi proposti usano il dataframe gatti:

scaricatelo da Elly, insieme alla descrizione delle variabili che lo compongono: leggete la descrizione e aprite il dataframe in R, prima di procedere oltre.

Modelli univariati per descrivere distribuzioni nominali

Distribuzione di frequenza

A livello **nominale**, i descrittori riguardano la **distribuzione di frequenza**, cioè lo schema con cui si associa a ciascuna modalità della variabile la **frequenza** pertinente, ossia il **numero di volte in cui si verifica un determinato "evento"** in un gruppo di altri eventi: **frequenze assolute**.

Le **frequenze relative** o **proporzioni** sono date dal **rapporto tra le frequenze assolute e il totale delle unità**; moltiplicate per 100, diventano **percentuali**.

$$f_i = \frac{n_i}{N}$$

Le **frequenze cumulate assolute** sono la somma di tutte le frequenze assolute che si susseguono dalla prima all'ultima modalità; dividendole per il totale delle osservazioni si avranno le frequenze cumulate **relative**, e, moltiplicando queste ultime per 100, le frequenze cumulate **percentuali**.

La funzione `table`

Le distribuzioni di frequenza sono sintetizzate da R con la funzione **`table(variabile)`**, che mostra le **frequenze assolute** in una **tabella di contingenza**. La modalità che compare con la maggiore frequenza nella variabile è la **moda**, cioè **l'indice di tendenza centrale** per le variabili a livello nominale.

Per sapere quanti maschi e quante femmine hanno partecipato alla ricerca sui gatti, usiamo `$genere`

```
> table(gatti$genere)
 F  M
48 31
```

Per sapere se hanno un gatto:

```
> table(gatti$vive_con_gatto)
no si
38 41
```

Ecco com'è distribuito il loro stato civile.

I due divorziati possono dare "fastidio" per analisi che confrontano i gruppi:

```
table(gatti$stato_civile)
 coniugato divorziato     single
        16          2         61
```

ri-categorizziamoli come "single", dicotomizzando la variabile:

Ri-assegniamo l'etichetta della categoria "single" alla categoria "divorziato":

```
gatti$stato_civile[gatti$stato_civile=="divorziato"]<-"single"
```

```
table(gatti$stato_civile)
coniugato divorziato     single
       16          0         63
```

Eliminiamo il livello "divorziato" del factor, che **ha frequenza = 0**, con `droplevels(factor)`, creando la nuova variabile `$stato_civile2`:

```
gatti$stato_civile2<-droplevels(gatti$stato_civile)
```

```
> table(gatti$stato_civile2)
coniugato     single
       16         63
```

Per aggiungere il marginale:

`addmargins(table)`

```
> addmargins(table(gatti$genere))
  F   M Sum
 48  31  79
```

Quando ci sono NA, in `table` aggiungiamo l'argomento **`exclude=NULL,`** che istruisce R a **non omettere** in output la frequenza dei dati mancanti.

*Usiamo per un esempio il dataframe **`attaccamento`**: che riporta i dati di caregiver di persone con demenza; venti pazienti sono ricoverati in RSA, altri venti sono a casa. **Solo per questi ultimi**, si è chiesto al caregiver **quali aiuti** avessero. La variabile ha quindi **NA**, corrispondenti alle risposte dei venti caregiver il cui assistito è ricoverato*

```
table(attaccamento$con_aiuto_di)
assistenza domiciliare   badante      nessuno
                     1        10            9
```

```
table(attaccamento$con_aiuto_di, exclude=NULL)
assistenza domiciliare   badante      nessuno          <NA>
                     1        10            9            20
```

La funzione `prop.table`

Per calcolare le **proporzioni** si usa `prop.table(table(variabile))`: oggetto di `prop.table` è quindi un oggetto `table`:

```
> prop.table(table(gatti$cresciuto_animali_domestici))
       no        si
0.4177215 0.5822785
```

Possiamo abbinare `round(oggetto, decimali)` per eliminare inutili decimali:

```
> round(prop.table(table(gatti$cresciuto_animali_domestici)),2)
  no   si
0.42 0.58
```

Se preferite le **percentuali**, m**oltiplicate le proporzioni** per 100:

```
> round(prop.table(table(gatti$cresciuto_animali_domestici)),3)*100
  no   si
41.8 58.2
```

Attenzione ai NA: se in `table` **non** indicate `exclude=FALSE`, il **totale** di riferimento sarà composto dai **soli dati non mancanti**:

```
> round(prop.table(table(attaccamento$con_aiuto_di)),2)
assistenza domiciliare                badante                nessuno
                  0.05                   0.50                   0.45
```

Proporzioni calcolate come rapporto tra frequenza di ogni modalità e ***totale dei dati non mancanti*** *N= 20*

Se invece indicate `exclude=FALSE`, le proporzioni saranno calcolate sul totale di **tutte le osservazioni**, **dati mancanti compresi**.

```
> round(prop.table(table(attaccamento$con_aiuto_di, exclude=NULL)),2)
assistenza domiciliare                badante                nessuno
                  0.02                   0.25                   0.22
```

Proporzioni calcolate come rapporto tra frequenza di ogni modalità e totale di ***tutte*** *le osservazioni N= 40*

`Freq(distribuzione)` del package **DescTools** dà frequenze assolute, percentuali e cumulate:

```
> Freq(gatti$cresciuto_animali_domestici)
    level  freq   perc  cumfreq  cumperc
1      no    33  41.8%       33    41.8%
2      si    46  58.2%       79   100.0%
```

Per gestire i dati mancanti si usa l'argomento `useNA=`: di default è "no", e si può cambiare impostando `useNA="always"`

```
> Freq(attaccamento$con_aiuto_di)
                    level  freq   perc  cumfreq  cumperc
1  assistenza domiciliare     1   5.0%        1     5.0%
2                badante    10  50.0%       11    55.0%
3                nessuno     9  45.0%       20   100.0%
```

```
> Freq(attaccamento$con_aiuto_di, useNA = "always")
                    level  freq   perc  cumfreq  cumperc
1  assistenza domiciliare     1   2.5%        1     2.5%
2                badante    10  25.0%       11    27.5%
3                nessuno     9  22.5%       20    50.0%
4                   <NA>    20  50.0%       40   100.0%
```

Per cambiare **l'ordine** delle categorie (di default, per ordine di livello), si usa `ord="asc"` per crescente e `ord="desc"` per decrescente

Raggruppamento in classi

Quando abbiamo caratteri continui o caratteri discreti con un numero elevato di modalità, si ricorre al **raggruppamento dei dati in classi**. Le classi possono essere di uguale o diversa **ampiezza**, data dalla differenza tra il **limite inferiore** e il **limite superiore-**

Vediamo come si presenta la distribuzione di frequenza di `gatti$eta`

```
> table(gatti$eta)
18 19 20 21 22 23 24 25 26 27 28 29 30 31 33 37 40 41 46 49 51 52 53 54 55 56 58 61 65 66 67 68
 1  2  1  1  4 13 10  6  2  3  1  2  8  1  2  1  1  1  2  1  1  1  1  1  2  1  2  1  1  2  1  2
```

La moda è 23 (anni), e **l'ampiezza delle classi è = 1**: ogni classe corrisponde a una modalità della variabile. Le **frequenze cumulate assolute** si ottengono con `cumsum(table)`:

```
> cumsum(table(gatti$eta))
18 19 20 21 22 23 24 25 26 27 28 29 30 31 33 37 40 41 46 49 51 52 53 54 55 56 58 61 65 66 67 68
 1  3  4  5  9 22 32 38 40 43 44 46 54 55 57 58 59 60 62 63 64 65 66 67 69 70 72 73 74 76 77 79
```

Da questa variabile continua **creiamo una variabile discreta** (`$eta_categorie`) dividendola in **quattro classi** di diversa ampiezza: 18 -25 anni, 26-45 anni, 46-60 anni, 61-68 anni

Ci sono **vari modi per farlo**: cominciamo a vederne uno.

```
gatti$eta_cat2[gatti$eta<=25]<-"ragazzi"
gatti$eta_cat2[gatti$eta>25 & gatti$eta<=45]<-"giovani"
gatti$eta_cat2[gatti$eta>45 & gatti$eta<=60]<-"maturi"
gatti$eta_cat2[gatti$eta>60]<-"anziani"
```

```
table(gatti$eta_cat2)
anziani giovani  maturi ragazzi
      7      22      12      38
```

La variabile dovrebbe essere di classe **factor**, o meglio **ordered factor**. Però:

```
class(gatti$eta_cat2)
[1] "character"
```

Cambiamo classe con **as.factor(variabile)** e vediamo i suoi livelli con **levels(factor)**

```
gatti$eta_cat2<-as.factor(gatti$eta_cat2)
```

```
levels(gatti$eta_cat2)
[1] "anziani" "giovani" "maturi"  "ragazzi"
```

```
class(gatti$eta_cat2)
[1] "factor"
```

Riordiniamo il fattore con **ordered(factor, livelli=)**, in cui scriveremo l'ordine corretto dei livelli:

```
gatti$eta_cat2<-ordered(gatti$eta_cat2,levels=c("ragazzi","giovani","maturi","anziani"))
```

```
table(gatti$eta_cat2)
ragazzi giovani  maturi anziani
     38      22      12       7
```

```
class(gatti$eta_cat2)
[1] "ordered" "factor"
```

Vediamo un'altra funzione per ri-categorizzare variabili: `ifelse(test= se trovi questa condizione, yes= fai questo, no= altrimenti fai quest'altro).`

`test=` condizione da soddisfare, `yes=` azione da eseguire se la condizione è soddisfatta, `no=` azione da eseguire se la condizione non è soddisfatta. Cominciamo con variabili **dicotomiche**: dividiamo i soggetti in **giovani** (**fino a 25 anni**) e **adulti** (**sopra i 25 anni**):

```
gatti$eta_due<-ifelse(test= gatti$eta<=25, yes= "ragazzi", no= "adulti")
```

```
table(gatti$eta_due)
 adulti ragazzi
     41      38
```

Possiamo **nidificarla** per creare variabili politomiche, inserendo più condizioni da soddisfare:

```
gatti$eta_tre<-ifelse(test= gatti$eta<=25, yes= "ragazzi",
 no=ifelse(test=gatti$eta>25 & gatti$eta<=45, yes="giovani",no="maturi"))
```

```
table(gatti$eta_tre)
giovani  maturi ragazzi
     22      19      38
```

```
gatti$eta_quattro<-ifelse(test= gatti$eta<= 25, yes= "ragazzi",
 no= ifelse(test= gatti$eta> 25 & gatti$eta<= 45, yes= "giovani",
 no= ifelse(test= gatti$eta >45 & gatti$eta <=60,  yes= "maturi", no= "anziani")))
```

```
table(gatti$eta_quattro)
anziani giovani  maturi ragazzi
      7      22      12      38
```

Ulteriormente nidificata, è subito `factor`:

```
gatti$eta_quattro<-as.factor(ifelse(test= gatti$eta<= 25, yes= "ragazzi",
 no= ifelse(test= gatti$eta> 25 & gatti$eta<= 45, yes= "giovani",
 no= ifelse(test= gatti$eta >45 & gatti$eta <=60,  yes= "maturi", no= "anziani"))))
```

La densità di frequenza

Le **ampiezze** delle classi 18-25, 26-45, 46-60, 61-68 sono differenti; si calcolano usando i **limiti reali** della classe, che considerano **l'approssimazione** (25.6 e 45.2 cadono entrambe nella categoria 26-45): sottraiamo 0.5 al limite inferiore e sommiamo 0.5 al limite superiore.

```
25.5-17.5   45.5-25.5   60.5-45.5   68.5-60.5
[1] 8       [1] 20      [1] 15      [1] 8
```

Il **rapporto tra frequenza e ampiezza della classe** è la **densità di frequenza**: **numero medio di casi per unità di ampiezza della classe**.

$$h_i = \frac{n_i}{c_i - c_{i-1}}$$

```
(numeratore<-table(gatti2$eta_discreta))
ragazzi giovani  maturi anziani
     38      22      12       7
```

```
38/22
[1] 1.727273
```

```
(denominatore<-c(25.5-17.5, 45.5-25.5, 60.5-45.5, 68.5-60.5))
[1]  8 20 15  8
```

```
densita_frequenza<-numeratore/denominatore
ragazzi giovani  maturi anziani
  4.750   1.100   0.800   0.875
```

*La densità di frequenza dei ragazzi è di **oltre quattro volte maggiore** di quella dei giovani; la loro frequenza assoluta N= 38, invece, è di **meno di due volte maggiore** della frequenza assoluta dei giovani N=22.*

Concludiamo osservano l'output di `Freq(distribuzione)` con variabili numeriche.

`Freq` decide in autonomia una propria suddivisione in classi di ampiezza costante, di cui mostra frequenze e frequenze cumulate:

```
Freq(gatti$eta)
      level  freq   perc cumfreq cumperc
1   [15,20]     4   5.1%       4    5.1%
2   (20,25]    34  43.0%      38   48.1%
3   (25,30]    16  20.3%      54   68.4%
4   (30,35]     3   3.8%      57   72.2%
5   (35,40]     2   2.5%      59   74.7%
6   (40,45]     1   1.3%      60   75.9%
7   (45,50]     3   3.8%      63   79.7%
8   (50,55]     6   7.6%      69   87.3%
9   (55,60]     3   3.8%      72   91.1%
10  (60,65]     2   2.5%      74   93.7%
11  (65,70]     5   6.3%      79  100.0%
```

Il numero di intervalli può essere definito da `breaks=`: ad esempio, per suddividere la distribuzione in 4 parti, scriveremo:

```
Freq(gatti$eta, breaks = 4)
           level  freq   perc cumfreq cumperc
1  [17.9,30.5]    54  68.4%      54   68.4%
2    (30.5,43]     6   7.6%      60   75.9%
3    (43,55.5]     9  11.4%      69   87.3%
4    (55.5,68]    10  12.7%      79  100.0%
```

Prima di proseguire:

1. *Calcolate le frequenze assolute, le proporzioni e le percentuali delle tre variabili che descrivono la capacità dei soggetti di discriminare l'intenzione comunicativa dei gatti nei tre contesti: quali commenti potremmo fare?*

---

2. *Anche la distribuzione del livello di istruzione non è ottimale: unite i soggetti con specializzazione post lauream ai laureati, creando la variabile $istruzione2; che tipo di variabile avete creato?*
3. *Selezionate solo i soggetti che vivono con un gatto e fate le stesse operazioni del punto 1: l'interpretazione del dato cambia?*
4. *Considerate per tutto il campione la variabile $empatia_gatti, che esprime l'autovalutazione sull'empatia specifica per i gatti:*
   a. *descrivete la distribuzione di frequenza della variabile;*
   b. *considerate i punteggi fino a 8 come indicatori di bassa empatia, da 9 a 18 come indicatori di media empatia e da 19 fino al più grande come indicatori di travolgente empatia: dividete la distribuzione della variabile in base a queste tre classi e calcolatene la densità di frequenza.*

Modelli univariati per descrivere distribuzioni ordinali

I ranghi

Nelle distribuzioni su scala ordinale, i numeri rappresentano la posizione dell'osservazione all'interno della distribuzione **ordinata**: abbiamo quindi una distribuzione di **ranghi**.

Dato che il discorso è **statisticamente molto semplice**, esercitiamoci un po' con R. **Selezioniamo** solo soggetti "esperti" nel campo felino: devono **vivere con un gatto**, essere **cresciuti con animali domestici** e **avere più di 25 anni**.

```
una_vita_con_gatto<-subset(gatti,cresciuto_animali_domestici=="si" & vive_con_gatto=="si"
 & eta>25)
```

```
length(una_vita_con_gatto$sogg)
[1] 11
```

Creiamo il dataframe **esperti** con il **codice** degli 11 esperti e il loro **punteggio di empatia** per i **gatti**:

```
esperti <- data.frame(soggetto= una_vita_con_gatto$sogg,
 empatia_gatti= una_vita_con_gatto$empatia_gatti)
```

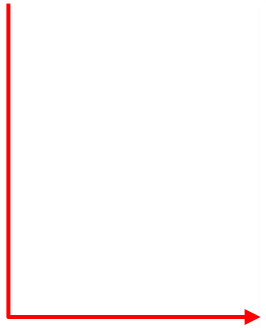
```
esperti
   soggetto empatia_gatti
1       s16             6
2       s17             8
3       s19            25
4       s20            22
5       s21             8
6       s28            27
7       s33            19
8       s34            22
9       s48            16
10      s73            22
11      s74            11
```

```
table(esperti$empatia_gatti)
 6  8 11 16 19 22 25 27
 1  2  1  1  1  3  1  1
```

Trasformiamo i valori di empatia in ranghi: 6 avrà rango “1”, essendo il più basso, mentre 27 avrà il rango più alto, cioè 11.

Usiamo **rank(variabile)** per creare **$ranghi**

```
esperti$ranghi<-rank(esperti$empatia_gatti)
```



Ecco il nuovo dataframe:

	soggetto	empatia_gatti	ranghi
1	s16	6	1.0
2	s17	8	2.5
3	s19	25	10.0
4	s20	22	8.0
5	s21	8	2.5
6	s28	27	11.0
7	s33	19	6.0
8	s34	22	8.0
9	s48	16	5.0
10	s73	22	8.0
11	s74	11	4.0

Facilitiamo la lettura **ordinando il dataframe** per rango con **`order(variabile)`**. L'ordine è **crescente**: **`decreasing= FALSE`** (defaulti) o decrescente: **`decreasing= TRUE`**

Il dataframe esperti è creato *dal dataframe `esperti`* *ordinato in base a* *i valori delle righe della variabile `$ranghi`, in senso crescente*

```
esperti<- esperti[order(esperti$empatia_gatti),]
```

mantenendo tutte le colonne presenti

```
esperti
   soggetto empatia_gatti ranghi
1       S16             6    1.0
2       S17             8    2.5
5       S21             8    2.5
11      S74            11    4.0
9       S48            16    5.0
7       S33            19    6.0
4       S20            22    8.0
8       S34            22    8.0
10      S73            22    8.0
3       S19            25   10.0
6       S28            27   11.0
```

Ai valori con stesso punteggio (**ties**) si assegna il loro **rango medio**, cioè ovvero la **media dei ranghi** che avrebbero occupato.

L'assegnazione del rango medio è l'impostazione di default per i ties di `rank(variabile, ties.method="average")`. Tra le altre impostazioni possibili ,"`min`" e "`max`" assegnano ai ties il rango più piccolo e il rango più alto che toccherebbe loro (come nelle gare di atletica):

```
esperti$ranghi<-rank(esperti$empatia_gatti, ties.method = "min")
esperti$ranghi
 [1]  1  2  2  4  5  6  7  7  7 10 11
esperti$ranghi<-rank(esperti$empatia_gatti, ties.method = "max")
esperti$ranghi
 [1]  1  3  3  4  5  6  9  9  9 10 11
```

L'argomento **opzionale** `na.last=` gestisce i **ranghi nel caso di dati mancanti**: di **default** è `na.last=TRUE` → i casi con NA sono posti in fondo alla distribuzione. Se `= FALSE` sono messi nei primi posti; se `=NA`, i dati mancanti sono omessi; se `= "keep"`, sono mantenuti nella distribuzione, con rango NA.

La mediana

La mediana è un **indicatore di tendenza centrale**: è la **modalità dell'osservazione che divide la distribuzione ordinata in due parti uguali** → valore al di sopra o al di sotto del quale cade un ugual numero di osservazioni.

Se N è dispari → è il valore che occupa il posto centrale $\frac{N+1}{2}$ della distribuzione **ordinata**;

se N è pari → è la media aritmetica dei termini che occupano le due posizioni centrali della graduatoria, ossia le posizioni $\frac{N}{2}$ e $\frac{N}{2}+1$

Noi potremmo contare così:

```
table(esperti$empatia_gatti)

 6  8 11 16 19 22 25 27
 1  2  1  1  1  3  1  1
       5           5
```

Ma chiederemo molto più rapidamente :

```
median(esperti$empatia_gatti)
[1] 19
```

I quantili

La mediana è **anche** un indice di posizione: gli **indici di posizione** indicano i **valori corrispondenti a specifiche posizioni** nella **distribuzione ordinata**.

- ✓ **Quartili**: **3 valori** q_1, q_2 e q_3 che dividono la distribuzione ordinata in **4 parti uguali**: sotto q_1 cade il 25% della distribuzione ordinata, sotto q_2 il 50% (**q_2 = mediana**), sotto q_3 il 75%. I valori che cadono tra q_1 e q_3 costituiscono il **range interquartilico** (**IR**) o **differenza interquartilica**: identificano i valori che occupano le posizioni centrali della distribuzione.
- ✓ **Decili**: **9 valori** che dividono la distribuzione ordinata in **10 parti uguali**.
- ✓ **Percentili**: **99 valori** che dividono la distribuzione ordinata in **100 parti uguali**.

Tutti rientrano nella categoria generale dei cosiddetti **quantili: valori corrispondenti a una generica posizione entro una distribuzione ordinata**.

I quartili sono indicati in `summary(distribuzione)`, che conosciamo:

```
summary(esperti$empatia_gatti)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  6.00    9.50   19.00   16.91   22.00   27.00
```

`fivenum(distribuzione)` riporta i cinque descrittori consigliati da **Tukey**, che ritroveremo parlando dei **boxplots**: sono gli stessi di `summary`, **tranne la media**.

*Vediamo la più ampia distribuzione `$empatia_gatti` **per tutti i 79 soggetti**:*

```
fivenum(gatti$empatia_gatti)
[1]  3.0  8.0 15.0 18.5 27.0
```

minimo — *1° quartile* — *mediana* — *3° quartile* — *massimo*

Come potremmo commentare i due output di esperti e campione complessivo?

In realtà, il primo e il terzo valore di `fivenum` non sono esattamente il 1° e il 3° quartile, ma la mediana della prima metà e la mediana della seconda metà della distribuzione ordinata... ma a fini pratici questa differenza è sostanzialmente irrilevante.

Con NA, si aggiunge **`na.rm=TRUE`**; la distribuzione non va ordinata prima del calcolo.

Per conoscere un qualsiasi percentile, si può usare `quantile(distribuzione, probs= posizione)`: `probs=(da 0 a 1)` indica la posizione del percentile.

Di default, `probs= c(0, 0.25, 0.5, 0.75, 1)`, cioè esattamente gli stessi indici di posizione di `summary` e `fivenum`, espressi però come percentili

```
quantile(gatti$empatia_gatti)
  0%  25%  50%  75% 100%
 3.0  8.0 15.0 18.5 27.0
```

Ma si può ottenere qualsiasi percentile o serie di percentili:

```
quantile(gatti$empatia_gatti, probs= c(.05, .33, .98))
   5%    33%    98%
 3.00 10.00 23.88
```

```
quantile(gatti$empatia_gatti, .25)
25%
  8
```

```
quantile(gatti$empatia_gatti, .75)
 75%
18.5
```

Si applicano a scale ordinali, a intervalli o rapporti equivalenti, e, diversamente dalla media e dalle misure di dispersione basate sulla media (varianza, deviazione standard), non sono influenzate dalla presenza di casi anomali → **statistiche robuste**.

Prima di proseguire:

*1. Considerate **tutti** i soggetti nel dataframe gatti: costruite la distribuzione la distribuzione delle frequenze percentuali assolute della variabile `$autovalutazione_relazione_gatto` e commentatela: è stata una buona idea? Perché?*

2. Calcolate il primo e il terzo quartile della variabile `$autovalutazione_relazione_gatto` e interpretatene l'output.

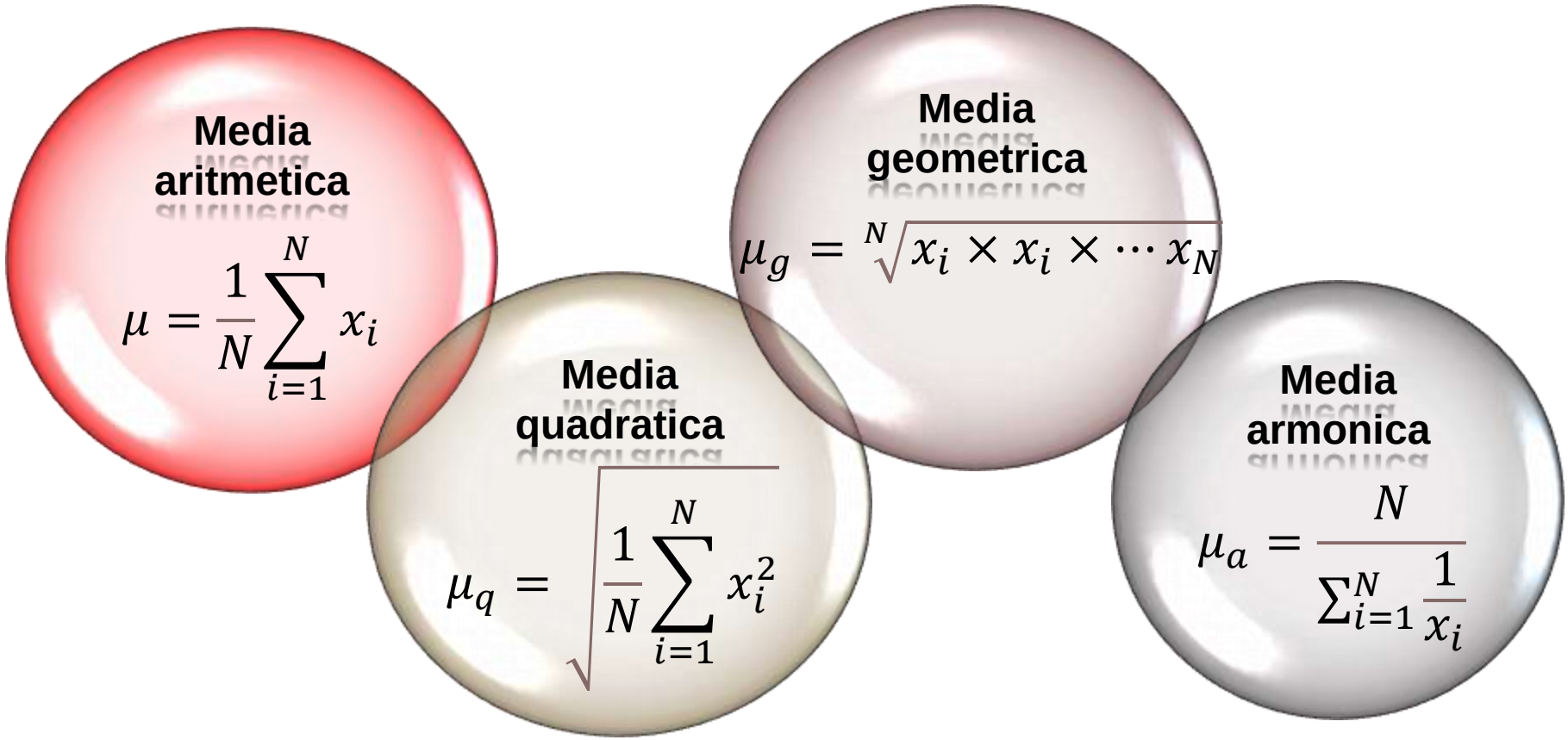
3. Usate i quantili così individuati per creare la variabile di raggruppamento `$amiconi`, in cui i soggetti che faticano a entrare in relazione con un gatto sono individuati dal livello "scarsa relazione", quelli così così dal livello "media relazione" e quelli che pensano come un gatto dal livello "buona relazione"

4. calcolate la densità di frequenza della variabile `$amiconi`

Modelli univariati per descrivere distribuzioni a intervalli e rapporti equivalenti

Media e varianza: goodness of fit ed errori del modello

Possiamo usare **tutte le proprietà dei numeri** per dati misurati a livello intervallare e a rapporti (**scale metriche**): useremo la **media** come indice di tendenza centrale, anche se sarebbe opportuno specificare media **aritmetica**, dato che:



Lavoreremo solo con la media aritmetica. Se ne avrete bisogno, ci sono `Gmean(variabile)` e `Hmean(variabile)` di `DescTools` per la media geometrica e armonica,

La **media** è uno dei **modelli** più semplici : è un **valore ipotetico** (**non** necessariamente **presente** nel dataframe), che **sintetizza** nella **il più fedelmente possibile** l'intera distribuzione.

Serviamocene per richiamare la **goodness of fit**: confrontiamo il modello-media con i dati e, tanto più **piccola** sarà la **differenza** tra dati e modello, tanto migliore sarà quest'ultimo.

Usiamo gli `esperti`; `$ranghi` non serve, **togliamola** anteponendo **– al numero di colonna**:

- Per **eliminare una o più righe**, anteponiamo **–** al numero di riga/righe:

  `dataframe[-riga,]`

- Per **eliminare una cella**, le assegniamo NA:

  `dataframe[riga, colonna]<-NA`

```
str(esperti)
'data.frame':11 obs. of  3 variables:
```

```
esperti<-esperti[,-3]
```

```
str(esperti)
'data.frame':11 obs. of  2 variables:
$ soggetto        : Factor w/ 79 levels "S1","S10"
$ empatia_gatti   : int  6 8 8 11 16 19 22 22 22 2
```

Abbiamo già visto la media in `summary`; richiediamola con **`mean(variabile)`**, aggiungendo **`na.rm=TRUE`** se ci sono NA:

```
summary(esperti$empatia_gatti)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   6.00    9.50   19.00   16.91   22.00   27.00 
```

```
mean(esperti$empatia_gatti)
[1] 16.90909
```

*Il **punteggio massimo ottenibile è 30**: il punteggio medio di questi soggetti sembra descrivere, quindi, una **moderata empatia***

Questo **modello-media si adatta bene a tutti i soggetti**? Se, non conoscendo nulla di uno dei soggetti tranne la **media** del suo gruppo di appartenenze, **facessimo previsioni** sulla sua empatia, queste **sarebbero affidabili**?

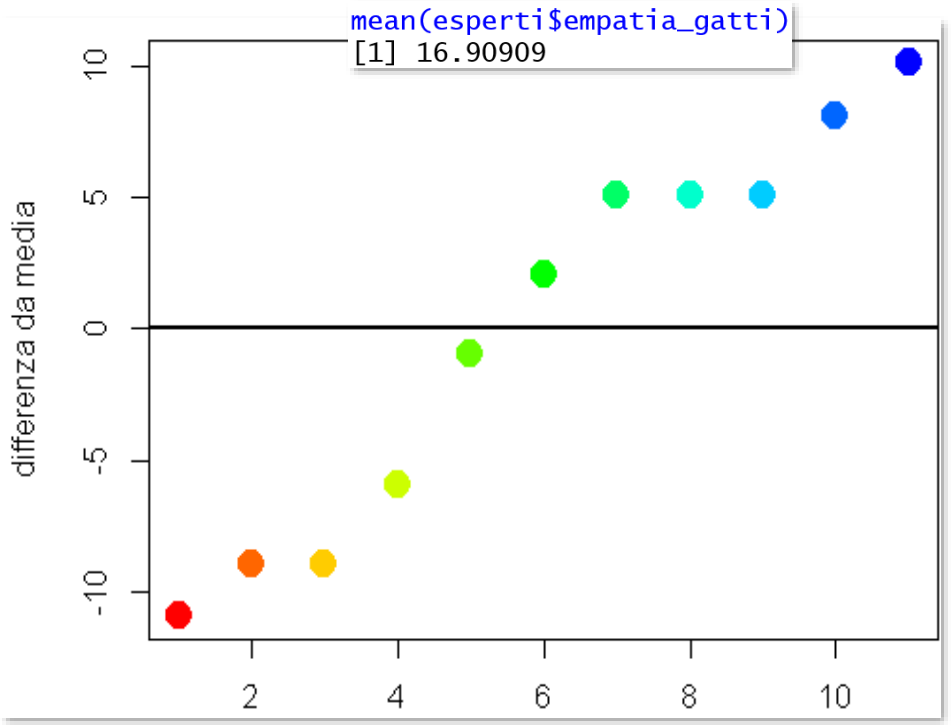
Verifichiamolo, **confrontando l'empatia di ogni soggetto con la media**: creiamo una variabile **composta dalle differenze** / **scarti** / **deviazioni dalla media** di ogni punteggio:

```
(esperti$differenza_media<-(esperti$empatia_gatti-mean(esperti$empatia_gatti))
[1] -10.9090909  -8.9090909  -8.9090909  -5.9090909  -0.9090909   2.0909091
[7]   5.0909091   5.0909091   5.0909091   8.0909091  10.0909091
```

```
esperti
sogg empatia_gatti ranghi differenza_media
 s16            6    1.0         -10.9090909
 s17            8    2.5          -8.9090909
 s21            8    2.5          -8.9090909
 s74           11    4.0          -5.9090909
 s48           16    5.0          -0.9090909
 s33           19    6.0           2.0909091
 s20           22    8.0           5.0909091
 s34           22    8.0           5.0909091
 s73           22    8.0           5.0909091
 s19           25   10.0           8.0909091
 s28           27   11.0          10.0909091
```

Per S48 la media è un ottimo modello, di altri (S16, S17, S21) la media sovrastima gravemente l'empatia, come gravemente sbaglia per sottostima con S19 e S28.

```
 s20           22    8.0           5.0909091
 s34           22    8.0           5.0909091
 s73           22    8.0           5.0909091
 s19           25   10.0           8.0909091
 s28           27   11.0          10.0909091
```



Nel complesso, questi **errori** nella stima / **devianze dalla medie** si **annullano**, dato che **la somma algebrica degli scarti è uguale a zero**:

```
round(sum(esperti$differenza_media),3)
[1] 0
```

```
plot(esperti$differenza_media, col=rainbow(15), pch= 19, cex=2,
ylab="differenza da media"); abline(h = 0, lwd=2)
```

Il modello sembrerebbe **perfetto**, avendo un **errore complessivo = 0**, ma **non è così**.

Per avere una stima **realistica** della **goodness of fit** del modello – media, eliminiamo i segni degli scarti **elevando al quadrato gli errori**, cioè gli scarti

```
esperti$scarti_quadrato<- esperti$differenza_media^2
```

La **somma** degli **errori al quadrato** (**Sum of Squared errors: SS**), cioè la somma degli scarti dalla media al quadrato, è la **devianza**.

```
esperti[,c(1,4)]
   soggetto scarti_quadrato
1       s16      119.0082645
2       s17       79.3719008
3       s19       65.4628099
4       s20       25.9173554
5       s21       79.3719008
6       s28      101.8264463
7       s33        4.3719008
8       s34       25.9173554
9       s48        0.8264463
10      s73       25.9173554
11      s74       34.9173554
```

La devianza, **indice di dispersione** della distribuzione attorno al valore centrale media, è **un indice di goodness of fit del modello**.

```
(devianza_esperti<-sum(esperti$scarti_quadrato))
[1] 562.9091
```

La devianza non è utile quando si tratta di **confrontare il fit di modelli diversi**: essendo una somma di errori, distribuzioni composte da pochi casi ottengono devianze più piccole di quelle rilevabili in distribuzioni più numerose, anche se il loro modello avesse un peggior fit.

La soluzione è **ponderare la somma degli errori per la numerosità della distribuzione**: **dividiamo la devianza** per N, o, meglio **per i gradi di libertà = N-1**, facendone la media. Otteniamo così l'indice di **goodness of fit ponderato** del modello (**Means of Squared errors: MS**), cioè la **varianza**

```
(varianza_esperti<-devianza_esperti/(11-1))
[1] 56.29091
```

Più semplicemente: `var(variabile)`:

```
var(esperti$empatia_gatti)
[1] 56.29091
```

Nelle statistiche di base non c'è una funzione per richiedere la devianza, ma basta **moltiplicare** `var` **per i gradi di libertà** $(N-1)$:

```
var(esperti$empatia_gatti)*(11-1)
[1] 562.9091
```

Ora, se tutti ci ricordiamo cosa sono i gradi di libertà, il discorso è chiuso, altrimenti ripassiamoli.

Gradi di libertà – degree of freedom - df

I gradi di libertà di una distribuzione corrispondono al **numero di valori indipendenti della distribuzione**, cioè **quelli il cui valore non dipende da alcun altro dato**.


```
quattro<-c(150,350,250, 50)
sum(quattro-250)
[1] -200
```

L'ultimo gettone estratto deve essere di 200 euro superiore alla media per far tornare i conti

```
cinque<-c(150,350,250, 50, 450)
sum(cinque-250)
[1] 0
```

Perciò, nella nostra distribuzione di $N = 5$, **4 gettoni hanno un valore indipendente** gli uni dagli altri, **ma 1 no**

Generalizzando

Un numero N di osservazioni che costituiscono un **campione** estratto da una **popolazione** può assumere **qualsiasi valore previsto nella popolazione**. Però, se vogliamo usare questo campione per calcolare la **varianza** della **popolazione** (e di solito siamo più interessati a stimare l'errore del modello nella popolazione, piuttosto che nel campione) dobbiamo usare la **media** del **campione** come **stima** della **media** della **popolazione**.

Dobbiamo quindi **tenere un parametro costante**. Se la **media del campione** è (ipoteticamente) $\bar{x} = 10$, assumiamo che la media della popolazione sia $\mu = 10$ e teniamo costante questo valore. Con questo parametro fisso, i valori delle N osservazioni non possono variare a piacere, perché **per mantenere costante la media, solo $N - 1$ di essi possono assumere qualsiasi valore previsto in popolazione**

Di conseguenza, **se teniamo un parametro costante, allora i gradi di libertà devono essere uno in meno rispetto al numero totale**

Ecco perché, quando usiamo un campione per stimare la varianza di una popolazione, dobbiamo dividere la devianza per $N - 1$, invece che per N.

Formalizzando:

Possiamo dare per assodato che la media **è** un semplice modello statistico che si adatta ai dati meglio per alcuni casi e peggio per altri? Allora formalizziamo questo concetto così

$$\boldsymbol{dato\ reale_i = (modello) + errore_i}$$

Il **dato osservato** è dato da / è uguale a / è **predetto dal modello** (**qui**, la media) più una **quota di errore intrinseca al modello** (**qui**, lo scarto dalla media). È un'equazione onnipresente; la ritroveremo nella regressione semplice.

Possiamo anche formalizzare devianza e varianza come indici di goodness of fit:

$$\boldsymbol{devianza = \sum (dato\ reale - modello)^2}$$

La qualità di un modello è analizzata **valutando le deviazioni dal modello dei dati reali**.

Anche questa equazione è ubiqua.

Confrontiamo la media degli esperti con la media dei soggetti non esperti: questi ultimi sono coloro che O non hanno un gatto, O non sono cresciuti con un animale domestico, O hanno ≤25 anni, O hanno una combinazione di queste caratteristiche, ma non tutte e tre.

Creiamo un fattore che distingua esperti (11) da non esperti, cominciando a definire i primi:

```
gatti$expertise[gatti$vive_con_gatto== "si" & gatti$cresciuto_animali_domestici== "si" &
 gatti$eta >25]<-"esperti"
```

```
table(gatti$expertise, exclude=NULL)
esperti    <NA>
     11      68
```

Gli NA **sono i soggetti non esperti**: assegniamogli l'etichetta "non esperti" con **`is.na`**:

```
gatti$expertise[is.na(gatti$expertise)]<-"non esperti"
```

```
table(gatti$expertise, exclude=NULL)
    esperti non esperti        <NA>
         11         68           0
```

Oppure, meno creativamente:

```
gatti$expertise <- ifelse(gatti$vive_con_gatto== "si" & gatti$cresciuto_animali_domestici== "si"
  & gatti$eta >25, "esperti", "non esperti")
```

```
table(gatti$expertise)
    esperti non esperti
         11         68
```

Non conosciamo la media dell'empatia dei 68 non esperti, che potrebbe essere inferiore all'empatia degli esperti: per calcolarla, **potremmo** creare il subset `non_esperti`

```
non_esperti<-subset(gatti, expertise=="non esperti")
mean(non_esperti$empatia_gatti)
[1] 13.08824
```

Ma è **più sensato** usare `tapply(X= misura, INDEX= fattore, FUN= funzione)`, che applica una funzione a **ogni gruppo di valori definito da un livello** (o da una combinazioni di livelli) di un fattore (o più fattori):

Applica *alla variabile $empatia* *per ogni livello del fattore $expertise* *la funzione "calcola media"*

```
tapply(X=gatti$empatia_gatti, INDEX=gatti$expertise, FUN=mean)
    esperti non esperti
   16.90909    13.08824
```

***Sì, è inferiore**: i non esperti hanno in media meno empatia degli esperti.*

Se ci sono dati mancanti e la funzione da applicare ha bisogno di istruzioni per gestirli, va indicato `na.rm=TRUE`

Com'è la qualità dei due modelli-media costruiti? Usiamo ancora `tapply` per richiedere la **varianza** dei due gruppi:

```
tapply(gatti$empatia_gatti, gatti$expertise, var)
    esperti non esperti
   56.29091    33.12643
```

La **MS è più bassa per il gruppo dei non esperti**: **la loro media è quindi un modello migliore per descrivere l'empatia** alla media degli esperti.

Se usassimo la SS, ignorando che il gruppo dei non esperti è oltre sei volte maggiore del gruppo di esperti, avremmo tratto conclusioni **opposte**:

Devianza degli esperti → *varianza degli esperti moltiplicata per* $N - 1$

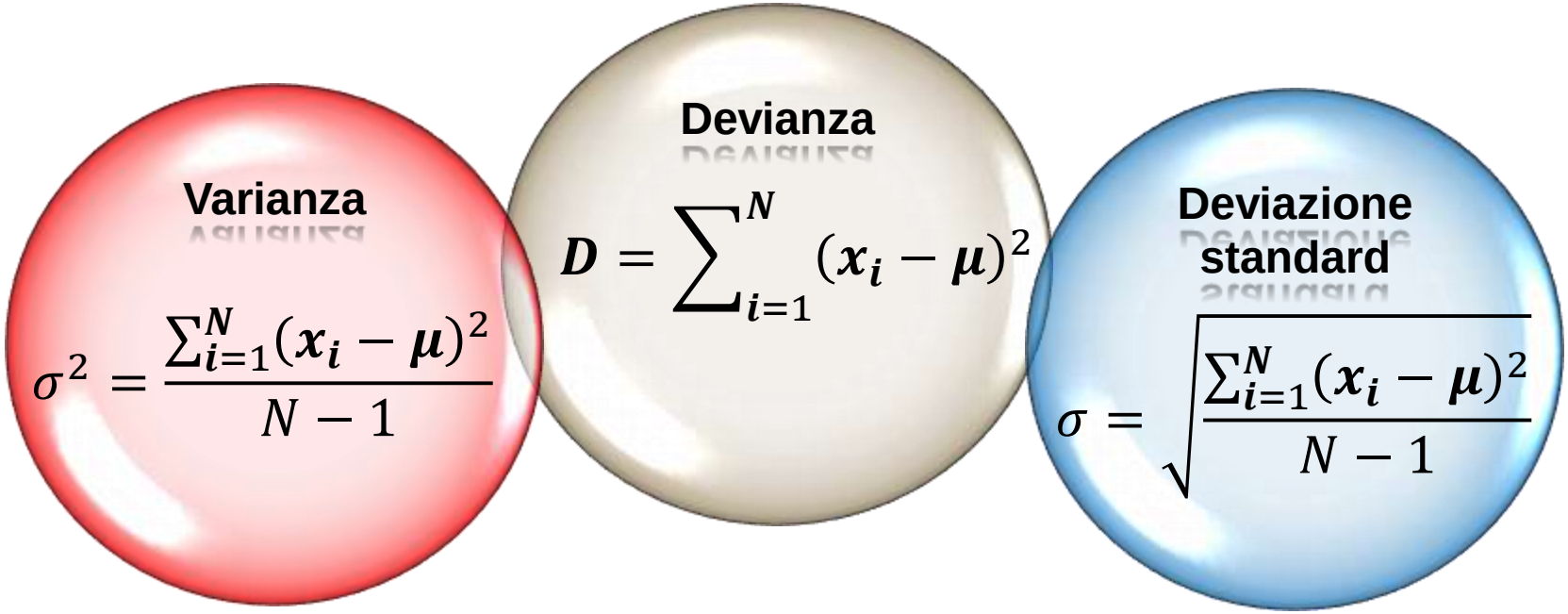
```
56.29091*10
[1] 562.9091
```

Devianza dei non esperti → *varianza dei non esperti moltiplicata per* $N - 1$

```
33.12643*67
[1] 2219.471
```

La deviazione standard o scarto quadratico medio

SS e MS sono scarti al quadrato: **mentre la media è espressa nella stessa unità di misura del carattere, la varianza è il quadrato di tale unità di misura**. Per **descrivere i dati** usando la stessa base, mettiamo la **varianza sotto radice quadrata**: **deviazione standard** (ds o sd).



Usiamo `sd(distribuzione)`:

```
tapply(gatti$empatia_gatti, gatti$expertise, sd)
    esperti non esperti
   7.502727    5.755556
```

La media trimmed

Se la distribuzione presenta **casi anomali** all'uno e/o all'altra **coda**, può essere opportuno eliminarli **prima di calcolare la media**, per non distorcerne il valore. Vedremo nella regressione come individuare con precisione i casi anomali (**outlier**) per **migliorare il fit di un modello**. Oppure, possiamo **eliminare una quota prefissata di casi alle due estremità della distribuzione**, e.g. il 2% dei casi più bassi e il 2% dei casi più alti: la media calcolata su una **distribuzione troncata** si definisce **trimmed mean**.

R usa l'argomento `trim=proporzione di casi da eliminare` nella funzione `mean` per calcolare la media troncata, con una proporzione massima $p_{max} = 0.5$ per coda.

```
mean(esperti$empatia_gatti); mean(esperti$empatia_gatti, trim=.2)
[1] 16.90909
[1] 17.14286
```

Scopriremo in TAD 2 la media e la varianza *winsorized*, anch'esse dedicate a risolvere il problema dei casi anomali.

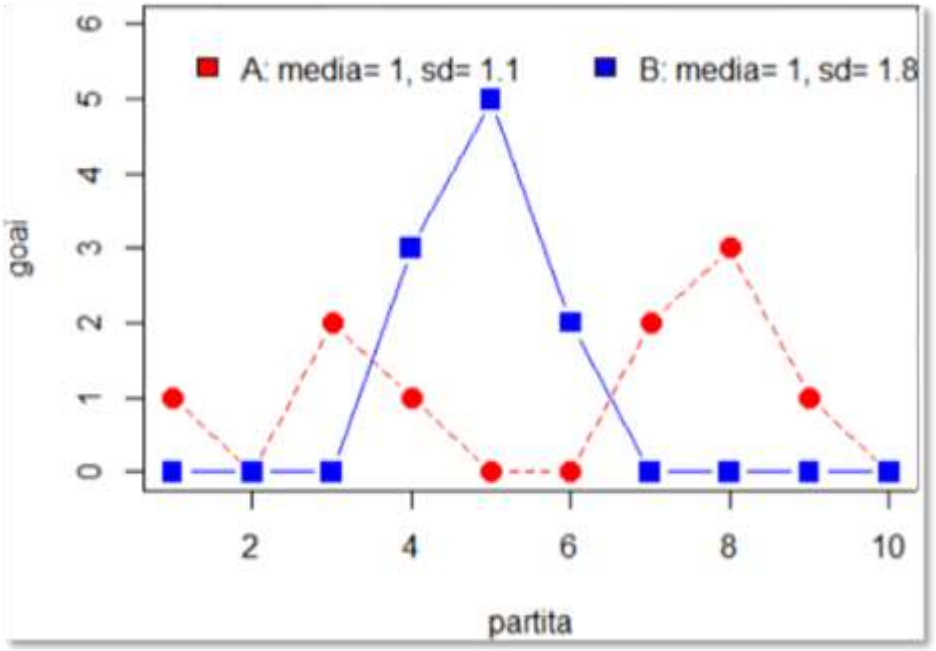
Descrivere una distribuzione con un indicatore di tendenza centrale **senza** affiancarvi l'informazione sulla dispersione è **sbagliato**.
L'interpretazione necessita di entrambe le informazioni.

Vediamo il profilo dei goal di due giocatori, A e B, nelle 10 partite di campionato che hanno entrambi giocato.

```
gioc_A<-c(1,0,2,1,0,0,2,3,1,0)
gioc_B<-c(0,0,0,3,5,2,0,0,0,0)
```

```
mean(gioc_A)
[1] 1
mean(gioc_B)
[1] 1
```

```
sd(gioc_A)
[1] 1.054093
sd(gioc_B)
[1] 1.76383
```



Entrambi hanno una $\boldsymbol{media_{goal} = 1}$*, ma il profilo del loro rendimento è chiaramente diverso: il giocatore* ***A ha una minore dispersione****, quindi un rendimento più* ***costante****, mentre il giocatore B ha una deviazione standard maggiore, indice di prestazioni decisamente imprevedibili.*

DescTools

Abbiamo accennato al package `DescTools`: approfondiamo le informazioni fornite dalla sua funzione `Desc(oggetto)`, diverse a seconda della classe dell'oggetto cui si riferisce.

Iniziamo con il caso più semplice: nominale, due categorie.

```
Desc(gatti$genere,main = "distribuzione del genere")
```

```
------------------------------
distribuzione del genere

  length      n    NAs unique
      79     79      0      2
          100.0%   0.0%

     freq   perc  lci.95  uci.95'
F      48  60.8%   49.7%   70.8%
M      31  39.2%   29.2%   50.3%
```

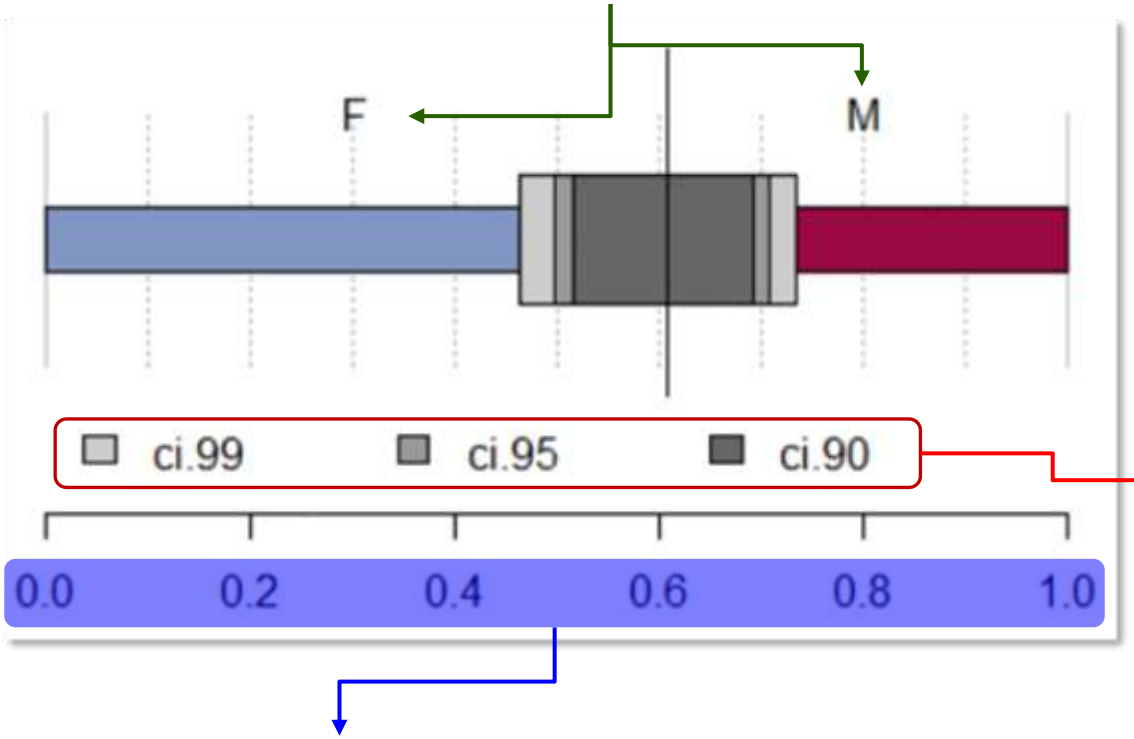
79 osservazioni (`length`), di cui 79 (100%) non missing (`n`) e nessuna (`0, %`) missing (`NAs`); compaiono due diversi valori (`unique`), **M e F**

48 F e 31 M (frequenze assolute: `freq`), corrispondenti al 60.8 e 39.2% dei casi totali (`perc`).

NON conosciamo ancora **l'intervallo di fiducia** (***CI***, confidence interval), in questo caso delle proporzioni: possiamo ignorarlo, per il momento.

Contemporaneamente a questo output, `Desc` produce anche un **grafico** (`plotit= TRUE`, di default)

proporzione cumulata di donne (F) e uomini (F).



separa la proporzione di donne (.608) da quella degli uomini

corrispondono all'ampiezza degli intervalli di fiducia, con tre diversi gradi di verosimiglianza: 90% (ci.90), 95% (ci.90) e .99% (ci.90).

Range delle proporzioni da 0 a 1

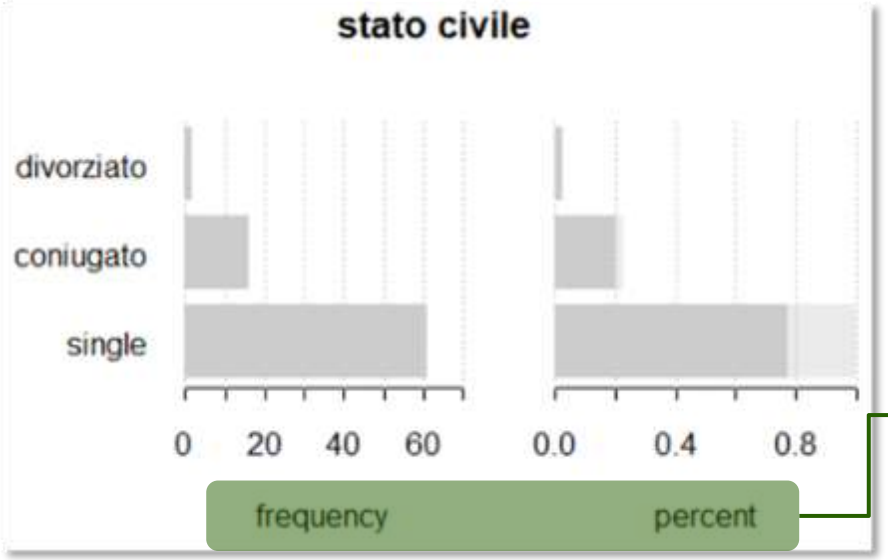
Ora vediamo una variabile nominale con più di due categorie: lo stato civile.

```
Desc(gatti$stato_civile,ord = "asc", main = "stato civile")
```

```
stato civile

   length         n       NAs    unique    levels     dupes
       79        79         0         3         3         y
           100.0%      0.0%

      level   freq    perc  cumfreq  cumperc
1 divorziato      2    2.5%        2     2.5%
2  coniugato     16   20.3%       18    22.8%
3     single     61   77.2%       79   100.0%
```



79 osservazioni (length), di cui 79 (100%) non missing (n) e nessuna missing (NAs); ci sono tre diversi valori (unique), corrispondenti a tre livelli, e nessun duplicato (dupes)

Ci sono 61 single, 16 coniugati e 2 divorziati (frequenze assolute: freq), corrispondenti al 77.2, 20.3 e 2.5% dei casi totali (perc). Sono anche riportate le frequenze cumulate (cumfreq) e le percentuali cumulate (cumperc).

Le barre del grafico indicano rispettivamente le frequenze (prima colonna) e le percentuali (seconda colonna) dei tre livelli.

```
Desc(gatti$empatia_gatti)
```

Vediamo **l'empatia per i gatti** come esempio di variabili `numeric` o `integer`:

```
length       n     NAs  unique      0s    mean  meanCI'
    79      79       0      22       0   13.62    12.25
           100.0%  0.0%            0.0%            14.99
```

```
   .05    .10    .25  median    .75    .90    .95
  3.00   5.60   8.00   15.00  18.50  21.00  22.00
```

percentili: dal 5% al 95%, compresa la mediana

```
range    sd   vcoef    mad    IQR   skew   kurt
24.00  6.12    0.45   7.41  10.50  -0.10  -1.04
```

La differenza tra minimo e massimo valore (`range`) è =24; la deviazione standard `sd` è 6.12, il range interquartilico (`IQR`) è 10.5. Non ci interessano il coefficiente di variazione (`vcoef`: media / sd) e la deviazione assoluta della mediana (`mad`), altri due indicatori di dispersione. Vedremo asimmetria (skewness, `skew`) e curtosi (`kurt`) della distribuzione. 0

```
lowest : 3 (5), 4 (3), 6 (3), 7 (5), 8 (5)
highest: 21 (6), 22 (3), 23, 25, 27
```

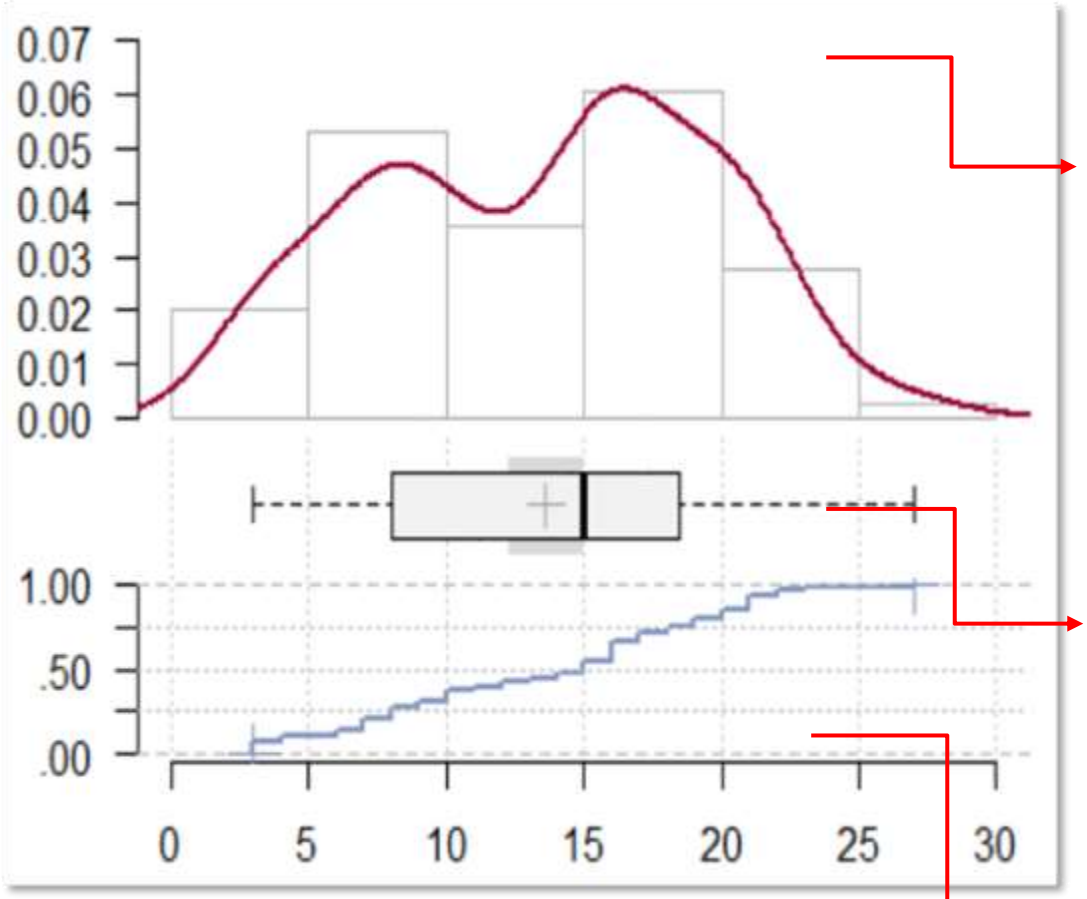
I 5 valori più alti e più bassi della distribuzione con rispettive frequenze; può essere utile conoscerli per individuare valori anomali, o decidere di calcolare una media trimmed

```
heap(?): remarkable frequency (11.4%) for the
mode(s) (= 16)
' 95%-CI (classic)
```

La moda corrisponde al punteggio 16, ottenuto dall'11.4% dei casi

Il CI della media è stato calcolato con il metodo "classico", quello che impareremo.

I grafici prodotti, nella stessa finestra, sono ben tre:



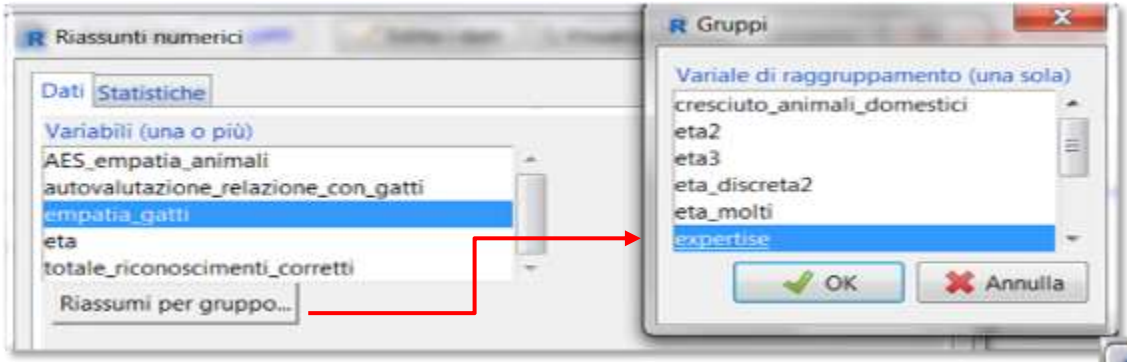
l'istogramma delle densità di frequenza cui è sovrapposta la **funzione densità di probabilità**: l'istogramma lo affrontiamo nel prossimo pacchetto di slide, la densità di probabilità in quello successivo.

boxplot della distribuzione: lo affronteremo nel prossimo pacchetto di slide

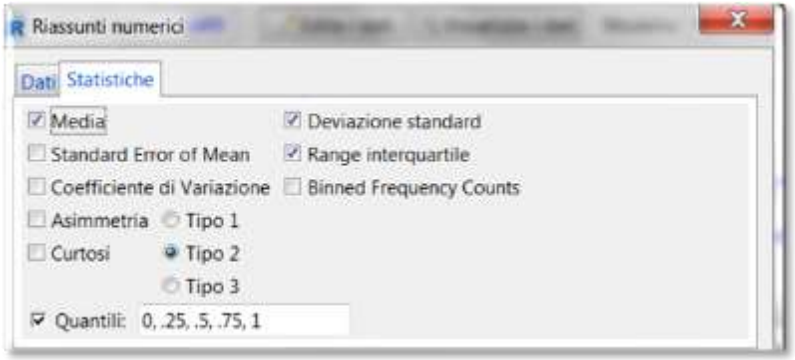
grafico delle frequenze relative cumulate

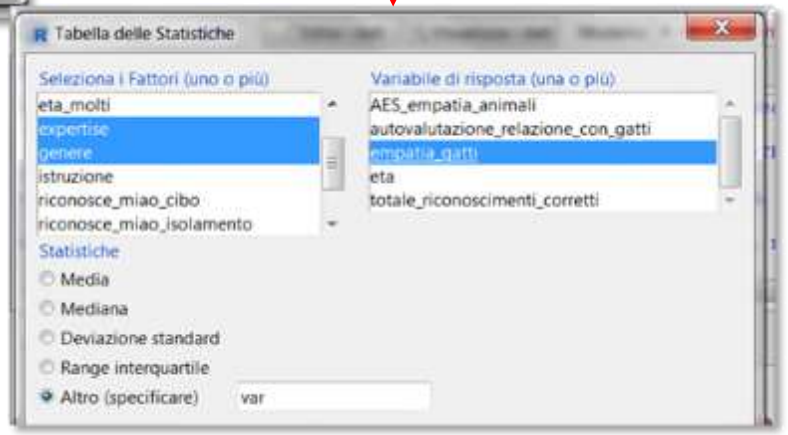
Con RCommander

Le statistiche descrittive non sono molte.



Per avere una descrizione in base alla combinazione dei livelli di più fattori:





Prima di proseguire:

*1. Considerando **tutti** i soggetti nel dataframe `gatti`, calcolate moda, mediana e media della variabile `$AES_empatia_animali` e commentate il dato, sapendo che il punteggio minimo teoricamente ottenibile è 22 e il massimo teoricamente ottenibile è 198;*

2a. Calcolate gli indici di dispersione della variabile `$AES_empatia_animali` per tutti i soggetti; e

2b. individuate i soggetti che rappresentano il 25% inferiore della distribuzione: etichettate loro come "antropocentrici" e tutti gli altri come "non antropocentrici". Verificate la correttezza della categorizzazione.

3. Calcolate il modello media della variabile `$AES_empatia_animali` per chi è cresciuto con un animale domestico e confrontatelo con quello di chi non è cresciuto con un animale domestico: quale modello si adatta meglio ai dati? Commentate i due modelli rispetto all'ipotesi che l'empatia non sia un tratto innato, ma una capacità che si può addestrare.