

Testing Psicologico

Lezione 2

Filippo Gambarota

@Università di Padova

Case Study

Un (fake) dataset

Il dataset `psych.csv` contiene un dataframe di dati simulati riguardo a **diagnosi psicologiche, questionari e variabili sociodemografiche**. Le variabili sono:

- `eta`: corrisponde all'età dei pazienti (età minima dei pazienti 10 anni)
- `diagnosi`: rappresenta la diagnosi psicologica
- `ses`: rappresenta lo status socio-economico
- `dep_score`: rappresenta il punteggio ad un questionario di depressione dove 0 è il punteggio minimo (assenza di aspetti depressivi) e 100 è il punteggio massimo di depressione.
- `ans_score`: rappresenta il punteggio ad un questionario di ansia dove 0 è il punteggio minimo (assenza di aspetti ansiosi) e 100 è il punteggio massimo di ansia
- `self_esteem_score`: rappresenta il punteggio ad item di autostima dove 1 è bassa autostima e 6 è alta autostima

Before starting...

- Importiamo il dataset (attenzione all'*estensione*, al *separator* e agli *argomenti* della funzione che scegliete)
- Controllare struttura e tipo di variabili del dataset
- Controllare **anomalie** nel dataset (valori strani o mancanti) e sistemarli. Gli errori anche in raccolta dati sono sempre dietro l'angolo
 - nel caso di valori mancanti, rimuovere quelle righe (vedi la funzione `complete.cases()`)
 - nel caso di valori anomali, creare un nuovo dataset con le righe anomale e rimuoverle dal dataset principale
- Aggiungere una colonna che indica il numero del paziente da 1 a quanti sono i pazienti
- Aggiungere una colonna `eta_bin` che prende i valori di "maggiorenne" e "minorenne" in base all'età
- Rimuovere dal dataset le righe associate a pazienti con diagnosi `altro`

Before starting...

```
dat <- read.csv("../..data/psych.csv", header = TRUE, sep = ";") # importare
```

```
str(dat) # struttura e tipo di variabili
```

```
## 'data.frame':    185 obs. of  6 variables:
## $ eta           : int  0 17 10 20 38 28 11 33 57 26 ...
## $ diagnosi      : chr  "ocd" "ocd" "ansiasociale" "ocd" ...
## $ ses           : chr  "medio" "alto" "alto" "alto" ...
## $ dep_score     : int  0 5 37 0 65 100 45 100 76 90 ...
## $ ans_score     : int  65 67 95 84 16 46 53 52 31 46 ...
## $ self_esteem_score: int  2 4 4 3 6 5 1 2 1 2 ...
```

```
# ?complete.cases # a cosa serve?
complete.cases(dat)
```

```
## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [16] TRUE TRUE TRUE TRUE TRUE
## [ reached getOption("max.print") -- omitted 165 entries ]
```

```
dat <- dat[complete.cases(dat), ] # selezioniamo solo quelli senza valori mancanti
```

Before starting...

Abbiamo detto che l'età dovrebbe essere maggiore/uguale a 10 anni. Vediamo se abbiamo dei casi anomali:

```
dat[dat$eta < 10, ] # errori di codifica o raccolta dati
```

```
##      eta      diagnosi    ses dep_score ans_score self_esteem_score
## 1      0              ocd medio          0          65                2
## 18     7 depressione alto          90          43                3
## 30     0 depressione medio         57          50                4
## [ reached 'max' / getOption("max.print") -- omitted 17 rows ]
```

```
dat_errori <- dat[dat$eta < 10, ] # dataset con errori
dat <- dat[dat$eta >= 10, ] # dataset senza errori
```

Continuiamo con il pre-processing...

```
dat$id <- 1:nrow(dat) # colonna che identifica il numero del paziente
```

```
dat$eta_bin <- ifelse(dat$eta >= 18, "maggiorenne", "minorenne")
```

```
dat <- dat[dat$diagnosi != "altro", ] # togliamo le diagnosi altro
```

Exploratory Data Analysis

EDA

- Calcoliamo delle statistiche descrittive appropriate per ogni tipologia di variabile
- Calcoliamo le frequenze relative delle diagnosi
- Calcoliamo le frequenze relative delle diagnosi condizionate al ses
- Facciamo un istogramma dei punteggi di ansia e depressione
- Facciamo un barplot dei punteggi di autostima
- Facciamo un barplot dei punteggi di autostima condizionati al tipo di diagnosi
- Calcoliamo media, mediana e deviazione standard e coefficiente di variazione dei punteggi di depressione, ansia e autostima per ogni diagnosi. Cosa notiamo?

EDA

```
summary(dat) # funzione utile ma molto generica
```

```
##      eta      diagnosi      ses      dep_score
## Min.   :10.00  Length:132    Length:132    Min.    : 0.00
## 1st Qu.:18.00  Class :character  Class :character  1st Qu.: 30.50
##  ans_score  self_esteem_score      id      eta_bin
## Min.    : 0.00  Min.    :1.000    Min.    : 1.00  Length:132
## 1st Qu.:33.75  1st Qu.:1.000    1st Qu.: 37.75  Class :character
## [ reached getOption("max.print") -- omitted 4 rows ]
```

```
table(dat$ses)
```

```
##
##  alto basso medio
##   61   34   37
```

```
table(dat$diagnosi)
```

```
##
## ansiasociale    bipolare  depressione      ocd
##           27           8           77           20
```

EDA

```
f_diagnosi <- table(dat$diagnosi)/length(dat$diagnosi)
f_diagnosi
```

```
##
## ansiasociale    bipolare depressione      ocd
## 0.20454545 0.06060606 0.58333333 0.15151515
```

```
sum(f_diagnosi)
```

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

```
# provate ad usare ?prop.table()
```

```
ds_f <- table(dat$diagnosi, dat$ses) # absolute, come le relativizziamo? vedi prop.table()
ds_f
```

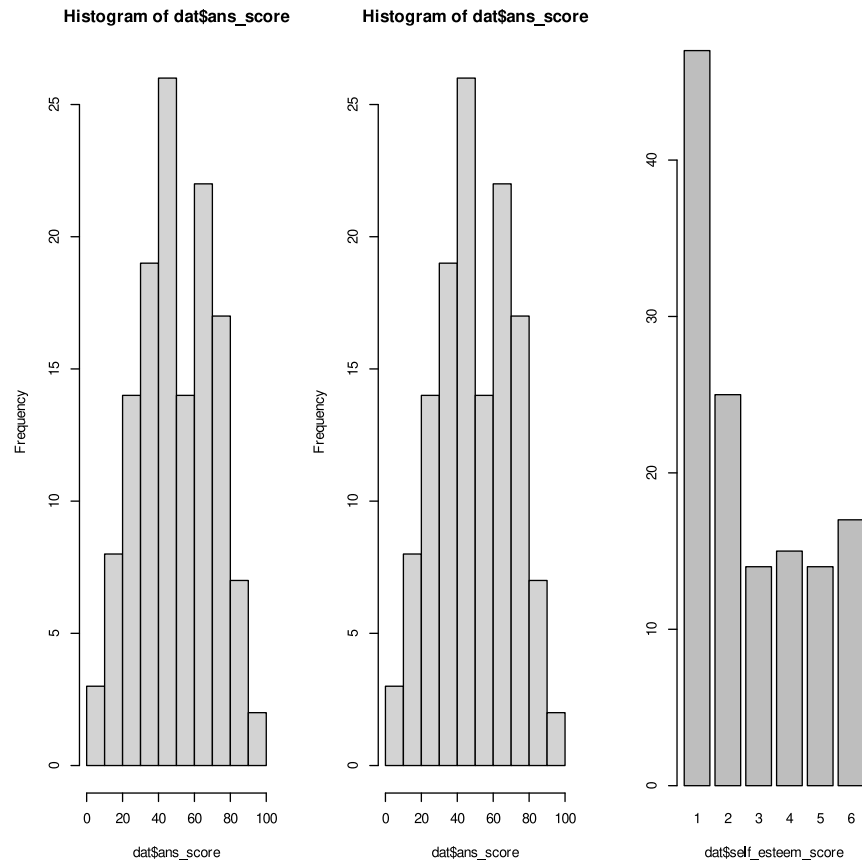
```
##
##          alto basso medio
## ansiasociale  12    8    7
## bipolare      2    1    5
## depressione  42   18   17
## ocd           5    7    8
```

```
prop.table(ds_f, margin = 1) # a cosa serve margin?
```

```
##
##          alto      basso      medio
## ansiasociale 0.4444444 0.2962963 0.2592593
## bipolare     0.2500000 0.1250000 0.6250000
## depressione  0.5454545 0.2337662 0.2207792
## ocd          0.2500000 0.3500000 0.4000000
```

EDA

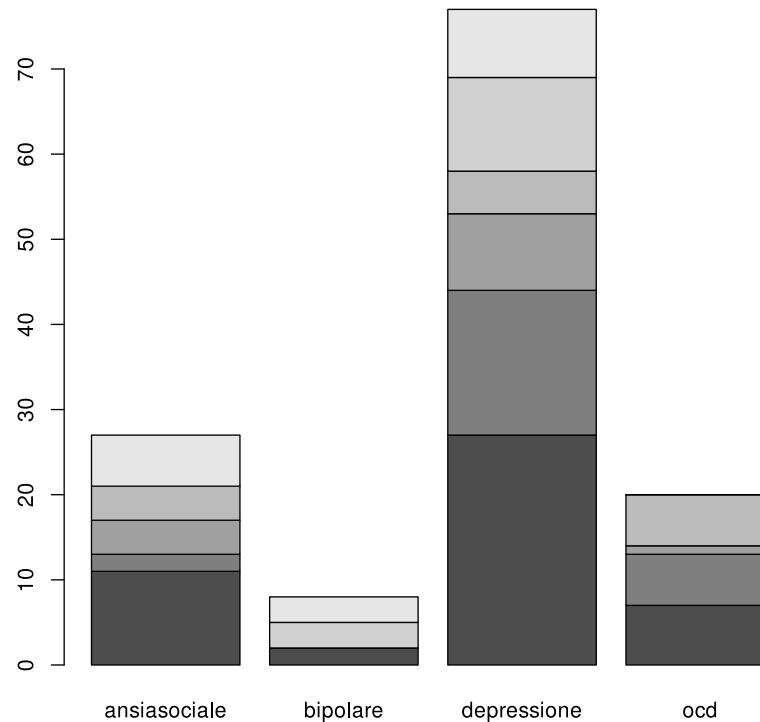
```
par(mfrow = c(1,3)) # per fare 2 grafici  
hist(dat$ans_score)  
hist(dat$ans_score)  
barplot(table(dat$self_esteem_score), xlab = "dat$self_esteem_score")
```



EDA

Il barplot condizionato è molto utile per esplorare visivamente la relazione tra due variabili categoriali:

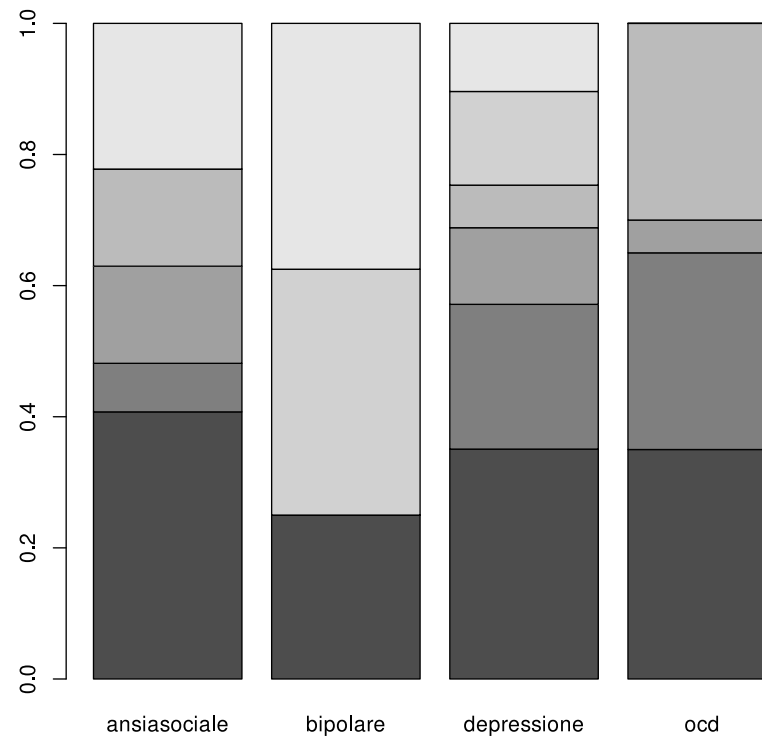
```
barplot(table(dat$self_esteem_score, dat$diagnosi)) # quale è il problema qui?
```



EDA

Dobbiamo fare le frequenze relative per fare in modo che sia più efficace:

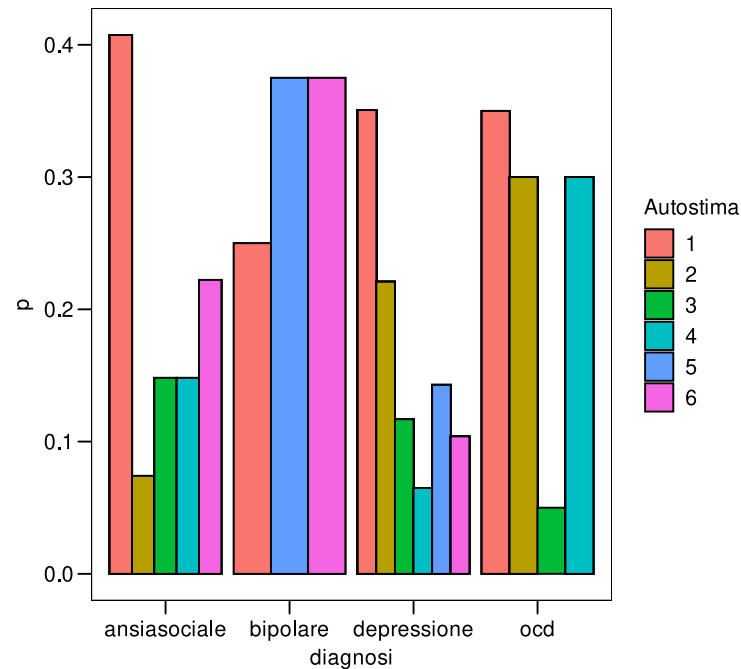
```
freq_self_diag <- prop.table(table(dat$self_esteem_score, dat$diagnosi), 2)  
barplot(freq_self_diag)
```



EDA

Facciamolo più carino usando `ggplot2` (Advanced)

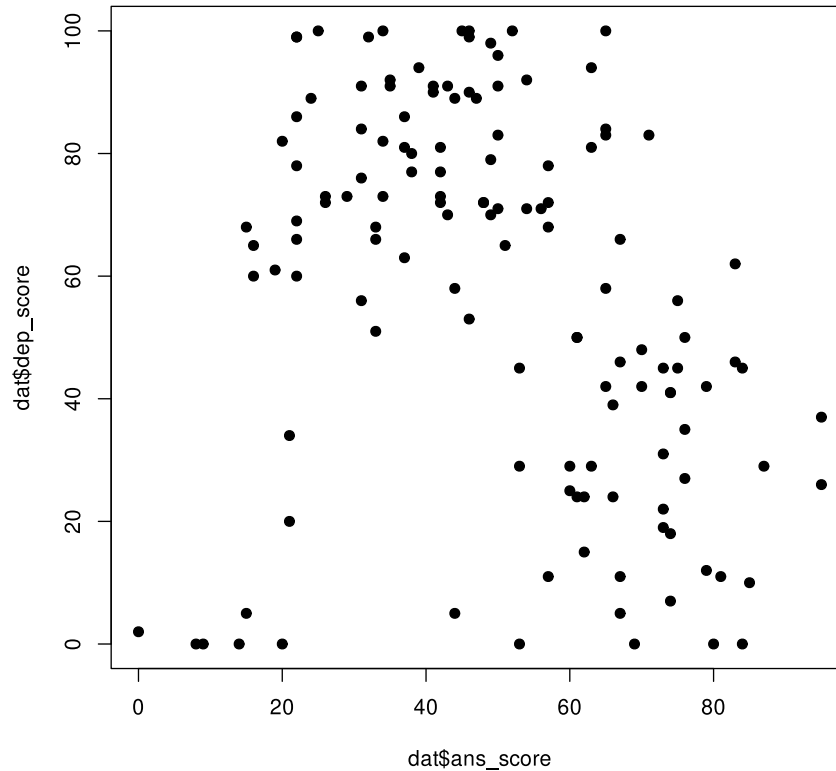
```
ggplot(datf) +  
  geom_col(aes(x = diagnosi, y = p, fill = factor(self_esteem_score)),  
           color = "black", position = position_dodge()) +  
  labs(fill = "Autostima") +  
  ggthemes::theme_par()
```



EDA

Facciamo uno scatterplot di ansia e depressione:

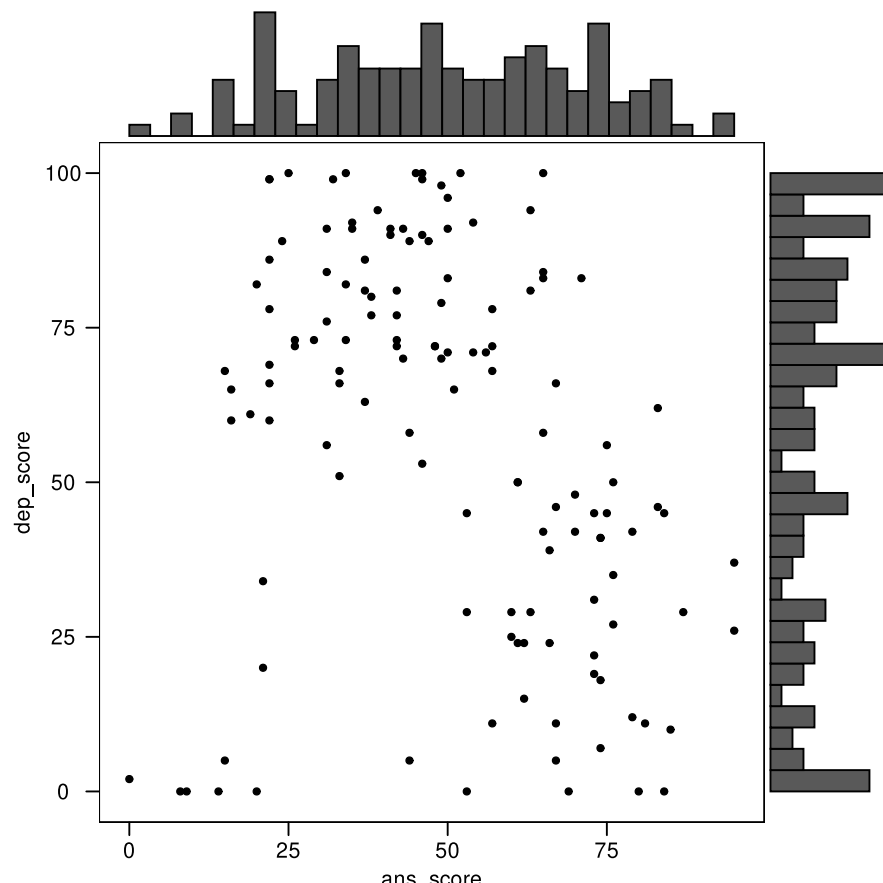
```
plot(dat$ans_score, dat$dep_score, pch = 19)
```



EDA

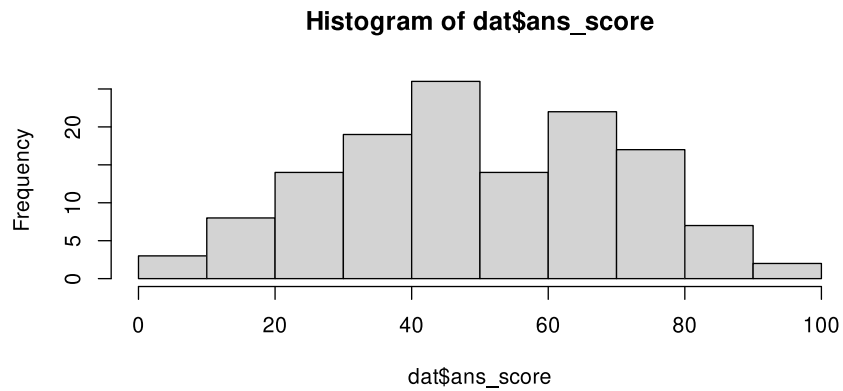
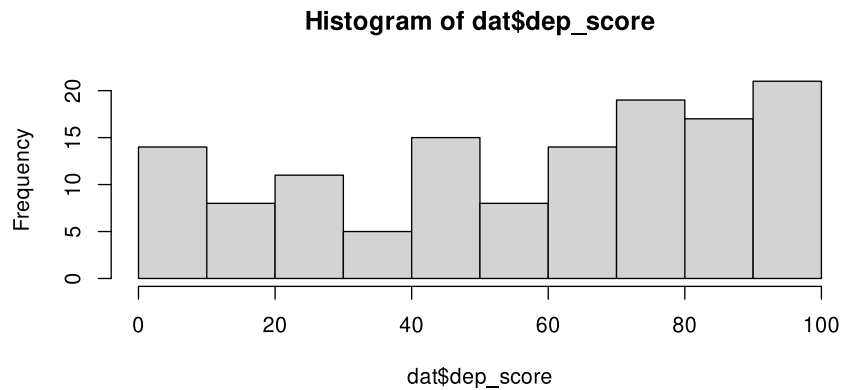
Più carino con le distribuzioni marginali:

```
plt <- ggplot(dat, aes(x = ans_score, y = dep_score)) +  
  geom_point() +  
  ggthemes::theme_par()  
  
ggExtra::ggMarginal(plt, type = "hist")
```



EDA

```
par(mfrow = c(2, 1))  
hist(dat$dep_score)  
hist(dat$ans_score)
```



EDA

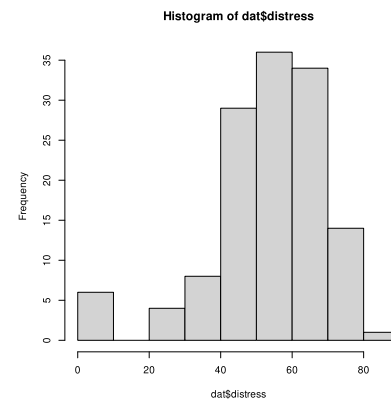
Calcoliamo un punteggio di "malessere" medio facendo la media dei punteggi di ansia e depressione per ogni soggetto:

```
distress <- apply(dat[, c("ans_score", "dep_score")], 1, mean)
dat$distress <- distress
```

```
summary(dat$distress)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	1.00	45.50	57.25	53.47	63.62	82.50

```
hist(dat$distress)
```



EDA

Calcoliamo media, mediana e deviazione standard e coefficiente di variazione dei punteggi di depressione, ansia e autostima per ogni diagnosi. Cosa notiamo?

```
mean(dat$dep_score[dat$diagnosi == "ansiasociale"])
```

```
## [1] 40.74074
```

```
mean(dat$dep_score[dat$diagnosi == "bipolare"])
```

```
## [1] 7.625
```

```
# ...
```

Questo non è il modo più efficace e compatto... 🙄

EDA

Possiamo usare la funzione `?tapply()` per applicare una funzione a `y` splittando per `x`:

```
cv <- function(x){  
  # https://it.wikipedia.org/wiki/Coefficiente\_di\_variazione  
  sd(x, na.rm = TRUE) / abs(mean(x, na.rm = TRUE))  
}  
  
# depressione  
tapply(X = dat$dep_score, INDEX = dat$diagnosi, FUN = mean, na.rm = TRUE)
```

```
## ansiasociale    bipolare  depressione      ocd  
##      40.74074      7.62500     79.62338     12.50000
```

```
tapply(X = dat$dep_score, INDEX = dat$diagnosi, FUN = median, na.rm = TRUE)
```

```
## ansiasociale    bipolare  depressione      ocd  
##           42           1           80           11
```

```
tapply(X = dat$dep_score, INDEX = dat$diagnosi, FUN = sd, na.rm = TRUE)
```

```
## ansiasociale    bipolare  depressione      ocd  
##      9.677860    12.648405    13.298238     9.439558
```

```
tapply(X = dat$dep_score, INDEX = dat$diagnosi, FUN = cv)
```

```
## ansiasociale    bipolare  depressione      ocd  
##      0.2375475    1.6588072    0.1670142     0.7551647
```

EDA - Extra

Facciamo un modo ancora più compatto! 😊

```
my_summary <- function(x){  
  c(  
    mean = mean(x, na.rm = TRUE),  
    median = median(x, na.rm = TRUE),  
    sd = sd(x, na.rm = TRUE),  
    cv = sd(x, na.rm = TRUE) / abs(mean(x, na.rm = TRUE))  
  )  
}  
  
tapply(dat$ans_score, dat$diagnosi, my_summary)
```

```
## $ansiasociale  
##      mean      median      sd      cv  
## 70.8148148 73.0000000 9.7390453 0.1375284  
##  
## $bipolare  
##      mean      median      sd      cv  
## 13.5000000 14.5000000 7.4642003 0.5529037  
##  
## $depressione  
##      mean      median      sd      cv  
## 40.6233766 42.0000000 13.8905865 0.3419358  
##  
## $ocd  
##      mean      median      sd      cv  
## 71.3000000 73.0000000 12.6370383 0.1772376
```

```
# ... così per tutte le variabili
```

EDA - Extra Extra Extra

Facciamo un modo ancoooooora più compatto! 🤖

```
ys <- list(depressione = dat$dep_score,  
           ansia = dat$ans_score,  
           autostima = dat$self_esteem_score)  
  
lapply(ys, function(x) tapply(x, dat$diagnosi, my_summary))
```

```
## $depressione  
## $depressione$ansiasociale  
##      mean      median      sd      cv  
## 40.7407407 42.0000000  9.6778597  0.2375475  
##  
## $depressione$bipolare  
##      mean      median      sd      cv  
##  7.625000  1.000000 12.648405  1.658807  
##  
## $depressione$depressione  
##      mean      median      sd      cv  
## 79.6233766 80.0000000 13.2982379  0.1670142  
##  
## $depressione$ocd  
##      mean      median      sd      cv  
## 12.5000000 11.0000000  9.4395584  0.7551647  
##  
##  
## $ansia  
## $ansia$ansiasociale  
##      mean      median      sd      cv  
## 70.8148148 73.0000000  9.7390453  0.1375284  
##  
## $ansia$bipolare  
##      mean      median      sd      cv  
## 13.5000000 14.5000000  7.4642003  0.5529037  
##  
## $ansia$depressione  
##      mean      median      sd      cv  
## 40.6233766 42.0000000 13.8905865  0.3419358  
##  
## $ansia$ocd  
##      mean      median      sd      cv  
## 71.3000000 73.0000000 12.6370383  0.1772376  
##  
##
```