

Didattica integrativa

Esercitazioni pratiche con il software R

dott. Luca Menghini Ph.D.

Assegnista di ricerca, Dipartimento di Psicologia,
Università degli Studi di Bologna

luca.menghini3@unibo.it

Analisi dei dati in ambito di comunità

Corso di laurea magistrale in Psicologia di comunità, della promozione
del benessere e del cambiamento sociale

Università degli Studi di Padova
Anno Accademico 2021 - 2022




Mi presento

- **2014: Triennale in Scienze Psicologiche Sociali e del Lavoro @uniPD**

“Biofeedback training per la gestione dello stress nei contesti organizzativi”

- **2016: Magistrale in Psicologia Sociale, del Lavoro e della Com. @uniPD**

“Un Protocollo di Assessment Psicofisiologico per la Valutazione del Rischio Stress lavoro-correlato”

- **2017: Tirocinio post-laurem presso Inside Performance** (stress management & biofeedback in contesti organizzativi e sportivi) + **Laboratorio di psicofisiologia @uniPD** (processamento dati e accuratezza sensori actigrafici) ← 

- **2017-2021: Dottorato in Scienze Psicologiche @uniPD**

“Workplace stress in real time: Towards the psychophysiological assessment of stressors and strain under ecological conditions”

- **2020: Esperienza di ricerca all'estero @SRI International (CA, USA)**

Accuratezza e uso Sleep Consumer Technology, Relazioni giornaliere tra sonno e stress

- **2021: Assegno di ricerca @uniBO**

“State workaholism as a predictor of daily fluctuations in blood pressure, emotional exhaustion, and sleep quality”

Obiettivi delle esercitazioni

- Acquisire competenze di base nell'uso del software R
- Consolidare le conoscenze apprese nel corso
- Implementare le tecniche analitiche apprese durante il corso utilizzando il software R su dataset reali
- Svolgere insieme gli esercizi propedeutici all'esame

Outline

- **Giorno 1:** Installare R e RStudio, acquisire confidenza con l'interfaccia del software, e i comandi di base per caricare un dataset e calcolare le statistiche descrittive
- **Giorno 2:** ..
- ...

Giorno 1

Il linguaggio R



- R è un linguaggio e un ambiente di programmazione per il **calcolo statistico** e la **visualizzazione grafica** dei dati
- basato sul 'linguaggio S' (Becker & Chambers, 1984), usato per creare il software S-Plus e poi R, creato da Ross Ihaka e Robert Gentleman, nel 1996
- oggi sviluppato da un gruppo di ricerca internazionale (*R Core Team*), che aggiorna periodicamente (ogni anno) il programma di base (*Base R*)
- progressiva ed esponenziale aggiunta di nuovi pacchetti (*packages*) che ne estendono le funzionalità

Il linguaggio R



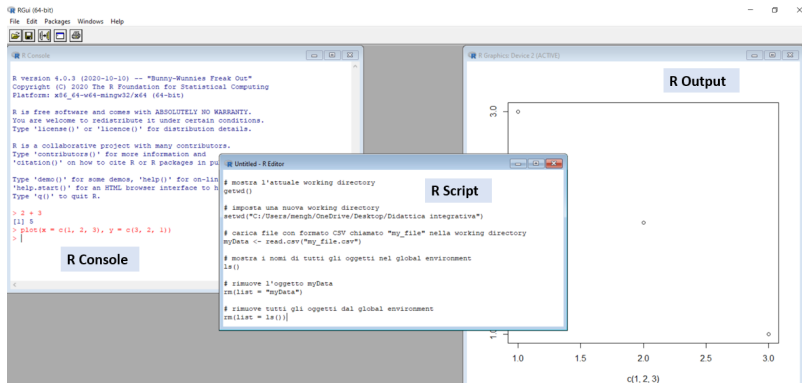
- ampia varietà di tecniche statistiche (es. modelli lineari e non lineari) e grafiche (es. pacchetto **ggplot2** - **link a lezione dedicata**)
- pensato per essere **semplice** ma al contempo in grado di generare **output di alta qualità** (grafici, tabelle e report con equazioni e simboli matematici, ecc.); funzioni di default ottimizzate + possibilità di avere il **pieno controllo**
- **software gratuito** (GNU General Public License), **open source** (ogni funzione è documentata e visibile in dettaglio) che funziona su tutti i **principali sistemi operativi**: Windows, MacOS, e UNIX (es. Linux)
- enorme comunità di utenti (per qualsiasi problema, basta googlare ;-))

Scaricare e installare R



1. Scaricare R dal sito <https://www.r-project.org/>
CRAN (*Comprehensive R Archive Network*): rete di server che offrono le versioni aggiornate e la relativa documentazione
2. Cliccare sulla voce **CRAN** nel menu *Download* a sinistra, selezionare un mirror (es. [il primo](#), oppure [quello dell'Università di Padova](#)), quindi il proprio sistema operativo (Linux, MacOS, o Windows)
3. Installare R aprendo il file .exe (Windows) o .pkg (MacOS) appena scaricato, oppure seguire i comandi in base alla propria versione di Linux

L'interfaccia di Base R



- **R Console:** per scrivere (>) ed eseguire (tasto Enter) velocemente dei comandi
- **R Script** (menu File > New R Script): per scrivere, modificare e salvare sequenze di comandi (salvati con formato .R)
- **Outputs** (es. plot): finestre che si aprono lanciando il relativo comando

Alcuni comandi elementari

Commenti (#)

```
# questo è un commento
```

Semplici operazioni matematiche

```
2 + 2 # addizione
```

```
[1] 4
```

```
2 * 2 # moltiplicazione
```

```
[1] 4
```

```
log(3) # algoritmo naturale
```

```
[1] 1.098612
```

```
exp(1) # funzione esponenziale
```

```
[1] 2.718282
```

Espressioni più lunghe (con **parentesi tonde**)

```
sqrt(5) * ( (4 - 1/2)^2 - pi/2^(1/3) )
```

```
[1] 21.81623
```

Assegnare valori a degli **oggetti** (<-)

```
x <- 3 # creo oggetto 'x' associato al valore 3
```

```
x # stampo il valore di x
```

```
[1] 3
```

I nomi degli oggetti possono includere lettere, numeri, trattini bassi e punti (es. pippo, pippo32, pippo.32, pippo_32)

```
pippo_32 <- x / 3
```

```
pippo_32 # stampo il valore di pippo_32
```

```
[1] 1
```

R è sensibile alle minuscole!

Mentre non è sensibile agli spazi

```
3+2
```

```
[1] 5
```

```
3 + 2
```

```
[1] 5
```

Hands on: Operazioni aritmetiche con R

Calcola il risultato delle seguenti operazioni utilizzando R ([soluzioni](#)):

Source: <https://psicostat.github.io/Introduction2R/first-comands.html#esercizi>

$$1. \frac{(45+21)^3 + \frac{3}{4}}{\sqrt{32 - \frac{12}{17}}}$$

$$2. \frac{\sqrt{7-\pi}}{3(45-34)}$$

$$3. \sqrt[3]{12 - e^2} + \ln(10\pi)$$

$$4. \frac{\sin(\frac{3}{4}\pi)^2 + \cos(\frac{3}{2}\pi)}{\log_7 e^{\frac{3}{2}}}$$

$$5. \frac{\sum_{n=1}^{10} n}{10}$$

Extra: Assegna il risultato dell'operazione 4 all'oggetto **x**, il risultato della 5 all'oggetto **y**, e calcola la somma **x + y**

RStudio



- RStudio è un ambiente di sviluppo integrato per R, che lo integra con un'**interfaccia grafica ottimizzata** per facilitarne l'utilizzo (es. accesso a file e oggetti, grafici, dataset, ecc.) presentando tutto in un'unica finestra
- fondato da J J Allaire nel 2009 (scritto con linguaggio Java e C++), gestito e sviluppato da gruppo di ricerca internazionale (gli stessi di **tidyverse** e **shiny**)
- **gratuito e open source** (GNU General Public License) + versioni a pagamento

Scaricare e installare RStudio



NB: soltanto **dopo** aver installato R

1. Scaricare RStudio dal sito <https://rstudio.com>
2. Cliccare sulla voce **Download** nel menu in alto, selezionare la versione gratuita (FREE) di **RStudio Desktop**, quindi il proprio sistema operativo
3. Installare RStudio aprendo il file appena scaricato

L'interfaccia di RStudio

The screenshot displays the RStudio interface with the following components:

- Source Pane:** Contains R code for creating a dataset and plotting a boxplot.

```
1 sleep # stampo dataset sleep
2
3
4 plot(extra-group,data=sleep) # boxplot
5
6
7
```
- Console Pane:** Shows the output of the executed code, displaying a table of data and the command prompt.

```
8 0.8 1 8
9 0.0 1 9
10 2.0 1 10
11 1.9 2 1
12 0.8 2 2
13 1.1 2 3
14 0.1 2 4
15 -0.1 2 5
16 4.4 2 6
17 5.5 2 7
18 1.6 2 8
19 4.6 2 9
20 3.4 2 10
> plot(extra-group,data=sleep) # grafico
> |
```
- Environment Pane:** Lists the objects in the Global Environment.

R Objects	
Data	
ab	36 obs. of 3 variables
abilities	36 obs. of 3 variables
att	210 obs. of 2 variables
- Plots Pane:** Displays a boxplot titled 'extra' on the y-axis and 'group' on the x-axis. The x-axis has two categories: 1 and 2. The y-axis ranges from -1 to 5.

- **Source:** R Scripts (.R), documenti e presentazioni (.Rmd), applicazioni (.app), ecc. Per lanciare un o più comandi, selezionali e premi **Ctrl + Enter** oppure clicca sul tasto Run in alto a destra
- **Environment** (oggetti presenti nel workspace) & **History** (storico comandi eseguiti)

Hands on: Operatori relazionali e logici

Operatori relazionali

```
3 == 3 # uguale
[1] TRUE

3 != 3 # diverso
[1] FALSE

x >= 3 # maggiore o uguale
[1] TRUE

5 %in% c(3, 5, 8) # inclusione
[1] TRUE
```

Operatori logici

```
x <- TRUE
y <- !x # negazione
y
[1] FALSE

x & (5 < 2) # congiunzione
[1] FALSE

x | (5 < 2) # disgiunzione inclusiva
[1] TRUE
```

Esercizi sugli operatori relazionali e logici:

Source: <https://psicostat.github.io/Introductions2R/first-comands.html#esercizi>

1. Definisci una proposizione per valutare la seguente condizione: “x è un numero compreso tra -4 e -2 oppure è un numero compreso tra 2 e 4”
2. Definisci due relazioni false e due vere che ti permettano di valutare i risultati di tutti i possibili incroci che puoi ottenere con gli operatori logici & e |
3. Esegui le seguenti operazioni $4 \wedge 3 \in c(2,3,4)$ e $4 * 3 \in c(2,3,4)$. Cosa osservi nell’ordine di esecuzione degli operatori?

Oggetti e funzioni

- **Oggetti:** identificano dei valori salvati nel workspace (**Environment**); i valori vengono assegnati agli oggetti con il simbolo `<-` (minore e meno); per richiamare un oggetto è sufficiente scrivere il suo nome

```
pippo_32 <- 2 # assegno valore a oggetto
pippo_32 # stampo oggetto
[1] 2
pippo_32 <- pippo_32 + 1 # aggiorno oggetto
pippo_32
[1] 3
```

- **Funzioni:** etichette associate a sequenze di comandi programmati per restituire uno specifico output (chiamato **valore**) sulla base di uno o più input (chiamati **argomenti**); il nome della funzione è sempre seguito dalle parentesi tonde, entro le quali si impostano gli argomenti (spesso ci sono dei valori di default)

```
sqrt(x = 9) # radice quadrata dell'argomento x
[1] 3
seq(from = 1, to = 5) # sequenza numerica dal valore 'from' al valore 'to'
[1] 1 2 3 4 5
```


Tipi (classi) di oggetti

Logical (logico)

```
x <- TRUE
x <- T
class(x)
[1] "logical"
```

Numeric (numeri)

```
x <- 1.4
class(x)
[1] "numeric"
```

Integer (numeri interi)

```
as.integer(x)
[1] 1
```

Character (stringa di testo)

```
x <- "Mi   piace R"
x
[1] "Mi   piace R"
```

Vector (vettore): serie di valori con la stessa classe (es. numeric) combinati con la funzione `c()` (combine)

```
x <- c(1, 10.5, 3, 2)
x + 1
[1] 2.0 11.5 4.0 3.0
sqrt(x)
[1] 1.000000 3.240370 1.732051 1.414214
y <- c("mi","piace", "R")
```

Matrix (matrice): tabella `nrow * ncol`

```
x <- matrix(1:12, nrow = 3, ncol = 4)
rownames(x) <- y # nomi di riga
x
```

	[,1]	[,2]	[,3]	[,4]
mi	1	4	7	10
piace	2	5	8	11
R	3	6	9	12

Funzioni e pacchetti

Molte cose in R si fanno usando delle funzioni, composte dai seguenti elementi: **nome**, **parentesi tonde**, **argomenti** (nomeArgomento = valoreArgomento oppure senza nome, in base alla posizione di default), valore restituito (*value*)

```
sqrt(x = c(1,2,3))  
[1] 1.000000 1.414214 1.732051  
  
sqrt(c(1,2,3))  
[1] 1.000000 1.414214 1.732051
```

R Help system: Per conoscere i dettagli (argomenti) di qualsiasi funzione, basta inserire il simbolo ? seguito dal nome della funzione

```
?sqrt
```

R packages: Per ottenere funzioni aggiuntive rispetto a quelle dei pacchetti di base, è necessario installare e aprire il relativo pacchetto (**package**)

```
install.packages("nome_pacchetto") # installare un pacchetto  
  
library(nome_pacchetto) # aprire un pacchetto  
  
nome_pacchetto::nome_funzione() # usare funzione senza aprire il pacchetto
```

Il primo passo di ogni analisi: La working directory

Per aprire un file che si trova in una specifica cartella, è necessario prima impostare la **working directory**, ovvero la cartella dalla quale vengono importati i file di input e nella quale vengono esportati i file di output.

```
getwd() # funzione per stampare la WD attuale
[1] "C:/Users/mengh/OneDrive/Desktop/PHD/Didattica/Didattica integrativa 2022 Analisi dei D
dir()[1:3] # primi 3 file nella WD
[1] "appunti Pastore.didattica integrativa.txt"
[2] "data"
[3] "Didattica integrativa 2022 Analisi dei Dati.Rproj"

setwd("C:/Users/mengh/OneDrive/Desktop") # impostare una nuova directory
```

Trucchetto con RStudio: ogni volta che iniziamo un nuovo progetto (es. analisi tesi, report progetto), come primo passo creiamo un nuovo **R project** (.Rproj) dal menu **File > New R Project**, selezionando una directory esistente o creandone una nuova. Così quella sarà già impostata come la WD di tutti gli script associati al progetto.

Caricare ed esportare un dataset

Per caricare un dataset che si trova nella working directory, è necessario usare una specifica funzione in base al formato del file.

```
# file .RData
load(file = "data/questionarioDidatticaIntegrativa.RData") # import
save(di, file = "data/questionarioDidatticaIntegrativa.RData") # export
```

R può importare ed esportare dati salvati con molti formati diversi, alcuni dei quali richiedono l'installazione di pacchetti aggiuntivi.

```
# file .CSV (comma separated values)
di <- read.csv("data/questionarioDidatticaIntegrativa.csv") # import
write.csv(di, "data/questionarioDidatticaIntegrativa.csv", row.names=FALSE) # export
```

Un assaggio di analisi: Statistiche descrittive

R ha delle funzioni di base per facilitare il calcolo delle statistiche descrittive di base.

```
summary(di$numVar) # descrittive variable numVar
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-1.9706	-0.5748	0.1251	0.1337	0.7109	2.0657

Oltre a questi, c'è una serie di funzioni per calcolare media, deviazione standard, ecc.

```
mean(di$numVar) # media
```

```
[1] 0.1336529
```

```
sd(di$numVar) # deviazione standard
```

```
[1] 0.9473183
```

Hands on: Dataset & descrittive

1. Scaricare il file `questionarioDidatticaIntegrativa.CSV` da Github:
<https://github.com/Luca-Menghini/eseRcitazioni> > `data`
2. Salvare il file in una cartella e impostare quella cartella come working directory
3. Importare il file su R
4. Calcolare la media, la mediana, il secondo quartile, e la deviazione standard della variabile `numVar`
5. Usare la funzione `histogram` per visualizzare la distribuzione della variabile `numVar`

Risorse & contatti

Risorse: Primi passi con R

In italiano:

- Callagher, C. Z., & Gambarota, F. (2021). *Introduzione a R*. Corso introduttivo online: <https://psicostat.github.io/Introduction2R>
- Pastore, M. (2015). *Analisi dei dati in Psicologia (Con applicazioni in R)*. Bologna: Il Mulino.

In inglese:

- R Core Team. *The R Manuals*. Manuali in formato pdf: <https://cran.r-project.org/manuals.html> (in particolare *An Introduction to R (with many examples, R Data Import/Export)*)
- Dalgaard, P. (2008). *Introductory statistics with R*. New York: Springer.

Contatti



dott. Luca Menghini, Ph.D.

Assegnista di ricerca, Dipartimento di Psicologia,
Università degli Studi di Bologna

luca.menghini3@unibo.it

[Linkedin](#) | [ResearchGate](#) | [GitHub](#) | [Twitter](#)