

# Introduzione a R



## Giornata 1

Filippo Gambarota

**In cosa consiste questo corso?**

# Gli obiettivi principali


- Impareremo tutti gli **elementi principali** che compongono il linguaggio R
- Impareremo i **costrutti della programmazione** e la loro applicazione in R
- Impareremo a **impostare un progetto** in R

# Altri obiettivi importanti

- Imparare la **logica generale della programmazione** spendibile con qualsiasi linguaggio
- Imparare gli elementi di base per poi stare al passo con la **velocissima evoluzione di R**
- Migliorare il modo di affrontare i problemi e **rendere più efficiente il lavoro quotidiano**

**Organizzazione pratica**

# Il materiale principale

- Libro **Introduction2R** disponibile online su Github 
- Slide del corso (principalmente per supportare le lezioni)
- Altra documentazione, link e materiale collegato

# Organizzazione delle lezioni

Il miglior modo per imparare un linguaggio è sicuramente la pratica con dati o piccoli problemi da risolvere. E' anche importante però avere una comprensione teorico-logica dei vari argomenti per poter essere più autonomi possibile.

- Presentazione degli argomenti principali con gli aspetti teorici più importanti
- Applicazione direttamente scrivendo codice
- Comprensione dei risultati del codice, dei messaggi di errore e dei casi particolari

# Materiale di supporto

- Il materiale del corso (principalmente scripts, slides e link) sarà disponibile alla pagina Github: [filippogambarota/corsoR](https://github.com/filippogambarota/corsoR)
- Il materiale e i vari link sono facilmente consultabili nel [sito associato](#).



# Un cenno al libro

Il libro è disponibile online ed è interamente scritto in R (si si possono scrivere anche libri 😊). Il libro è in costante *work in progress* in base a errori, feedback e aggiornamenti stessi di R.

Il libro è nato da un'idea del mio amico e collega Claudio Zandonella Callegher ed io ho contribuito in alcune sezioni.

Potete attivamente contribuire allo sviluppo del libro fornendoci:

- suggerimenti
- **segnalare errori**
- segnalare parti non chiare

# Segnalare errori libro

In particolare, errori di scrittura o imprecisioni sono le cose più importanti. Per questo ho pensato ad un modo efficiente per gestire il tutto. Trovate istruzioni dettagliate [qui](#), in breve:

- trovate un errore o imprecisione
- aprire il [documento condiviso](#)
- riportare tutti i dettagli
- sentirsi soddisfatti 😊

**Che cos'è R**



**R** è un linguaggio di programmazione fortemente votato alla statistica, gestione di dati e visualizzazione.

E' nato nel 1993 da **Ross Ihaka** e **Robert Gentleman** come successore di un'altro linguaggio chiamato **S**.


















E' un software completamente **open-source** e **gratuito** in continua evoluzione e cambiamento.

**Perchè R**

# Qualche statistica

Worldwide, Aug 2021 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	29.93 %	-2.2 %
2		Java	17.78 %	+1.2 %
3		JavaScript	8.79 %	+0.6 %
4		C#	6.73 %	+0.2 %
5	↑	C/C++	6.45 %	+0.7 %
6	↓	PHP	5.76 %	-0.0 %
7		R	3.92 %	-0.1 %
8		Objective-C	2.26 %	-0.3 %
9	↑	TypeScript	2.11 %	+0.2 %
10	↓	Swift	1.96 %	-0.3 %
11	↑	Kotlin	1.81 %	+0.3 %
12	↓	Matlab	1.48 %	-0.4 %
13		Go	1.29 %	-0.2 %
14	↑↑	Rust	1.21 %	+0.2 %
15	↓	VBA	1.16 %	-0.1 %
16	↓	Ruby	1.02 %	-0.1 %
17		Scala	0.79 %	-0.1 %
18	↑	Ada	0.77 %	+0.2 %

Jul 2021	Jul 2020	Change	Programming Language	Ratings	Change
1	1		 C	11.62%	-4.83%
2	2		 Java	11.17%	-3.93%
3	3		 Python	10.95%	+1.86%
4	4		 C++	8.01%	+1.80%
5	5		 C#	4.83%	-0.42%
6	6		 Visual Basic	4.50%	-0.73%
7	7		 JavaScript	2.71%	+0.23%
8	9	↗	 PHP	2.58%	+0.68%
9	13	↗	 Assembly language	2.40%	+1.46%
10	11	↗	 SQL	1.53%	+0.13%
11	20	↗	 Classic Visual Basic	1.39%	+0.73%
12	8	↘	 R	1.32%	-1.08%
13	12	↘	 Go	1.17%	-0.04%
14	50	↗	 Fortran	1.12%	+0.90%
15	24	↗	 Groovy	1.09%	+0.51%
16	10	↘	 Swift	1.07%	-0.37%
17	16	↘	 Ruby	0.95%	+0.14%

PYPL and TIOBE index

# Qualche statistica

## Contributed Packages

### Available Packages

Currently, the CRAN package repository features 18578 available packages.

[Table of available packages, sorted by date of publication](#)

[Table of available packages, sorted by name](#)

### Installation of Packages

Please type `help("INSTALL")` or `help("install.packages")` in R for information on how to install packages from this repository. The manual [R Installation and Administration](#) (also contained in the R base sources) explains the process in detail.

[CRAN Task Views](#) allow you to browse packages by topic and provide tools to automatically install all packages for special areas of interest. Currently, 41 views are available.

### Package Check Results


All packages are tested regularly on machines running [Debian GNU/Linux](#), [Fedora](#), macOS (formerly OS X) and Windows.

# E se lo dice Google 😊

## Google Data Analytics Professional Certificate

This is your path to a career in data analytics. In this program, you'll learn in-demand skills that will have you job-ready in less than 6 months. No degree or experience required.


★★★★★ 4.8 16,261 ratings

 Google Career Certificates **TOP INSTRUCTOR**

**Enroll for Free**  
Starts Aug 4

Financial aid available

277,280 already enrolled



[About](#) [How It Works](#) [Courses](#) [Instructors](#) [Enrollment Options](#) [FAQ](#)

### WHAT YOU WILL LEARN

- ✓ Gain an immersive understanding of the practices and processes used by a junior or associate data analyst in their day-to-day job
- ✓ Understand how to clean and organize data for analysis, and complete analysis and calculations using spreadsheets, SQL and R programming
- ✓ Learn key analytical skills (data cleaning, analysis, & visualization) and tools (spreadsheets, SQL, R programming, Tableau)
- ✓ Learn how to visualize and present data findings in dashboards, presentations and commonly used visualization platforms



#### Shareable Certificate

Earn a Certificate upon completion



#### 100% online courses

Start instantly and learn at your own schedule.



#### Flexible Schedule

Set and maintain flexible deadlines.



# Un cenno all'open-source

Un software si definisce open-source quando il codice sorgente è disponibile a tutti per essere modificato, aggiornato e controllato.

Open-source non significa necessariamente **gratuito** ma è sicuramente una delle modalità migliori per sviluppare e mantenere un progetto software.

R è sia open-source che gratuito e vanta una community estremamente attiva, come spesso accade con tutti i progetti open-source e in generale i linguaggi di programmazione.

# Quale è la concorrenza?

Il principale "concorrente" di R è sicuramente **Python** che offre un ambiente altrettanto potente, sviluppato e attivo

Non è facile (e forse non è possibile) capire quale sia il migliore. Sicuramente se produrre grafici, gestire dati e fare analisi statistiche è la priorità, R è la scelta più adeguata.

Una volta imparato R, Python in ogni caso è molto semplice e può essere utile per altri scopi come:

- Elaborazione immagini
- Creare esperimenti (Psychopy)

# Quale NON è la concorrenza?

Nell'ambito della statistica ci sono vari software non open-source ed a pagamento come:

- Statistica
- SPSS
- STATA
- SAS

# Quale NON è la concorrenza?

Sono degli ottimi software ma:

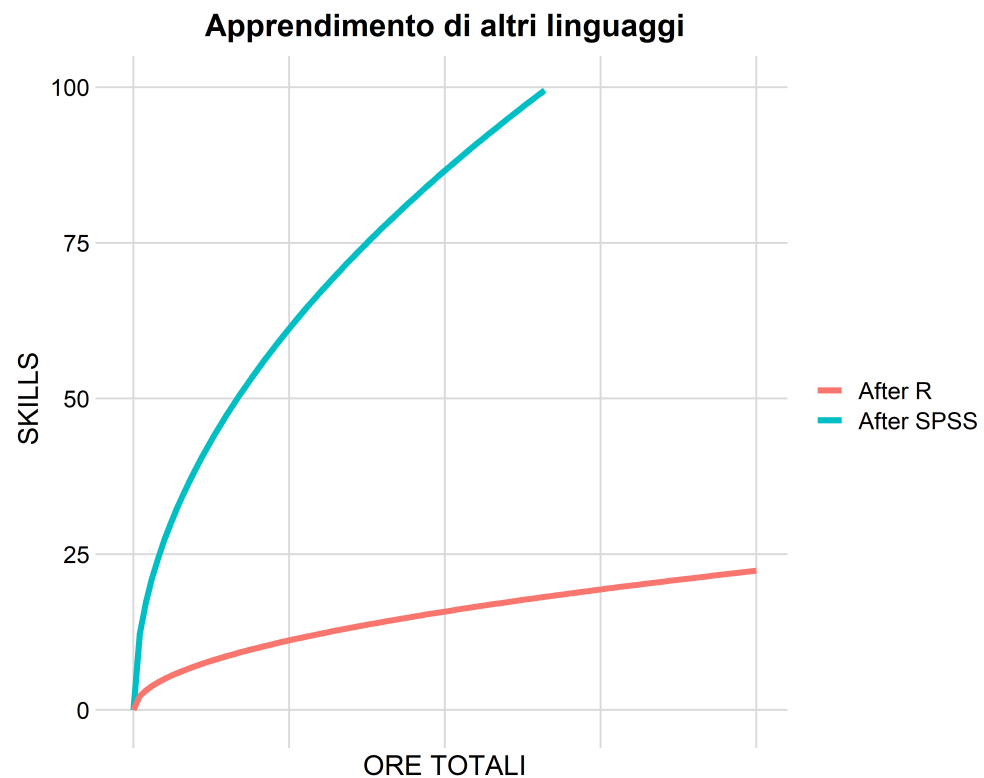
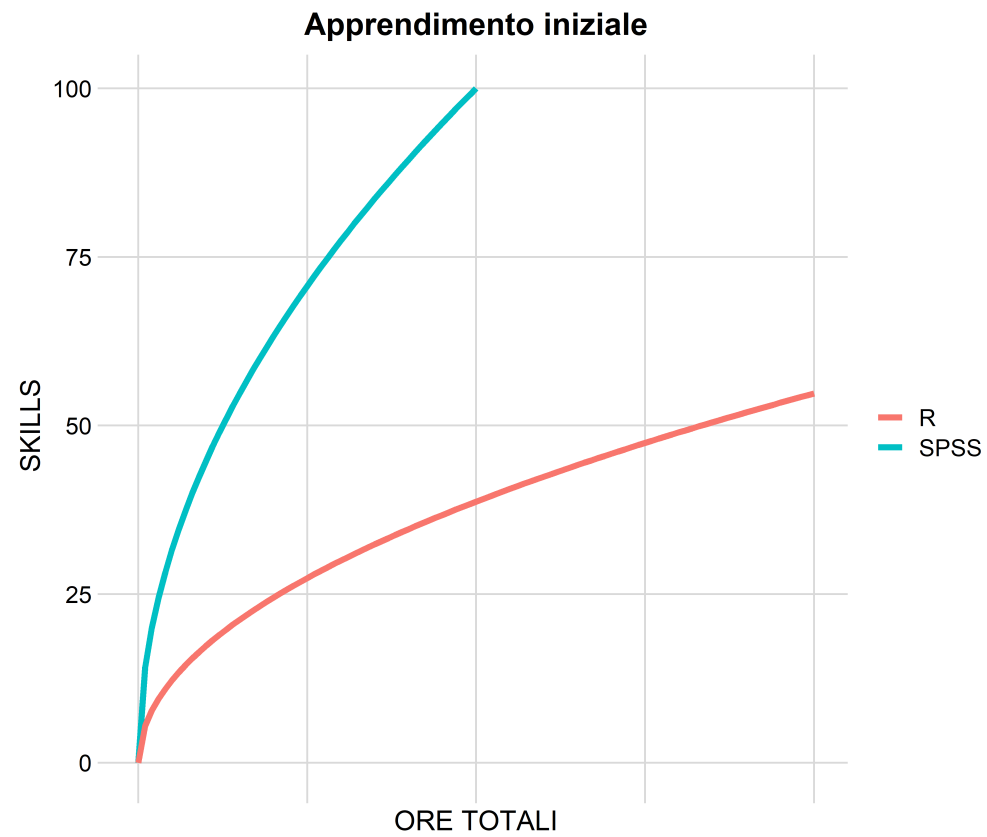
- Non forniscono conoscenze trasversali
- Siete legati ad uno specifico ambiente
- Le licenze possono costare molto
- La community non è altrettanto attiva (non open-source)

# Imparare un linguaggio come investimento

Imparare in linguaggio come R vi permette di conoscere uno strumento molto potente ma anche di imparare:

- Ragionare e risolvere problemi con il codice
- Trasferire quello che avete imparato ad altri linguaggi
- Essere sempre autonomi e non legati ad uno specifico ambiente
- Avere una skill realmente di valore

# Imparare R è difficile...MA ne vale la pena!



# R come una lingua parlata

Una lingua è composta da:

- regole grammaticali: **cosa è giusto/sbagliato**
- influenze ed evoluzioni culturali: **la lingua cambia ed evolve**
- slang ed espressioni tipiche
- convenzioni e best practice

# R come una lingua parlata

Anche in R, come in tutti i linguaggi di programmazione sono presenti questi aspetti. Questa metafora del linguaggio di programmazione come lingua parlata vorrei che stesse con noi per tutto il corso perchè molto utile per avere una visione d'insieme.