

Introduzione a R

- linguaggio a oggetti per analisi statistica
- più moderno di SAS
- Rstudio
- interattivo
- case sensitive

- open source (base)
- Finestre:
 1. Editor
 2. Console (terminale)
 3. History
 4. Files, Plots, Packages, Help

$2+3$ somma

$2-3$ sottrazione

$2*3$ moltiplicazione

$2/3$ divisione

$4 + 5 * 10$ espressione

$2 ^ 3$ potenza

Tabella, come dataset in SAS.

- Dalla barra degli strumenti superiore:

1. Session
2. Set Working Directory
3. Choose Directory
4. *working-directory*
5. tasto 'Seleziona Cartella'

Nuovi oggetti

- assegnazione: <-, ->

```
x1 <- 2+3
```

- Visualizzare: scrivere il nome e eseguire
- ; è un separatore, non terminatore

```
x2 <- 2-3; x2
```

Espressioni su oggetti

```
x3 <- x1 + 2 * x2
```

```
x3
```


Oggetti nel workspace

- `ls()`
- `objects`
- finestra *Environment*

Cancellare un oggetto

- `rm(x1)`
- `remove(x2)`

Per cancellare tutti gli oggetti:

- `rm(list=ls())`

- `?rm`
- Vale per le funzioni

- .RData: sessione di lavoro
- .R: programma in R

- `str(x1)`: mostra la struttura di un oggetto
- `dim(x1)`: mostra la dimensione di un oggetto
- `length(x1)`: mostra la lunghezza di un oggetto

- scalare: numero o stringa
- liste: sequenza ordinata di valori **eterogenei**
- vettori: sequenza ordinata di valori **omogenei**
- matrici: vettore bidimensionale
- data frame: come dataset in SAS

```
lista <- c(1/2, 1/2+2, 1/(2+2), 1/2*4, 1/(2*5)); lista
```

```
lista2 <- c(1, 4, "parola"); lista2
```

```
lista3 <- c(1/x, 1/x+2, lista2); lista3
```

- una lista è **senza dimensione**

```
dim(lista2)
```

- progressioni aritmetiche di ragione 1

```
a <- 1:6; a  
b <- 0.5:10.5; b
```

1. Gli estremi sono *inclusi*
2. concatenabili in `c()`
3. ordine decrescente possibile

```
c(0.5:3, 22:20, 2:1)
```


Costruzione sequenze

```
seq(from=3,to=12)
```

```
seq(from=3,to=12, by=3)
```

```
seq(1,10, length=3) # ragione 4.5
```

```
rep(1,5)  
rep(1, times=5)
```

```
rep(1:6, times=2)
```

```
rep(1:6, each=2)
```

- Liste di valori omogenei
- cbind: costruzione per colonna
- rbind: costruzione per riga

```
e <- cbind(1,2,99); e
```

```
str(e)  
dim(e)  
length(e)
```

Questa volta la dimensione esiste

```
f <- rbind(1,2,99); f
```

```
str(f)
```

```
dim(f)
```

```
length(f)
```

- insieme di colonne

```
m1 <- cbind(c(1,0.3,4), c(2,3,5), c(10, 0.8, 0.987), c(1, 2, 5)); m1  
str(m1)  
dim(m1)  
length(m1)
```

- insieme di righe

```
m2 <- rbind(c(1,2, 10,1), c(0.3, 3, 0.8, 2), c(4, 5, 0.987, 5)); m2  
str(m2)  
dim(m2)  
length(m2)
```

1. coerenza nel numero di elementi
2. le colonne sono omogenee

Matrici 4

```
m3 <- matrix(0,3,2); m3  
str(m3)  
dim(m3)
```

```
m4 <- matrix(c(1,2,3,4,5,6), 2, 3); m4
```


Matrice diagonale

- `diag(8, 3, 3)`
- `diag(c(1,2,3), 3, 3)`
- `diag(m2)`
- `diag(10)`
- `diag(4,10,6)`

- `cos()`: coseno
- `sin()`: seno
- `exp()`: esponenziale
- `log()`: logaritmo
- `sqrt()`: radice quadrata
- `abs()`: valore assoluto

Esempio

```
10*cos(5)+ 1/80 - exp(-3) + sqrt(20)*log(3)
```

- `choose()`: coefficiente binomiale
- `choose(5,2)`

Operazioni su vettori

```
x <- rbind(3, 7, 1.4, 11); x  
y <- rbind(2, 0, 7, 1); y
```

Somma

$x + y$

Moltiplicazione scalare per vettore:

$4 * x$

Prodotto tra due vettori:

$x \%*\% t(y)$

Operazioni su matrici

- somma e differenza
- prodotto di uno scalare per una matrice
- trasposizione
- prodotto matriciale
- matrice identità `diag(4)`
- traccia = somma degli elementi sulla diagonale principale: `sum(diag(m1))`
- determinante: `det()`
- risoluzione equazioni lineari $Ax = B$: `solve(A, B)`

- nomi oggetti: `names()`
- nomi colonne: `colnames()`
- nomi righe: `rownames()`
- `colnames(m1) = c("c1", "c2", "c3", "c4")`
- `m1[, "c2"]`

- matrice
- intestazione di colonne
- data row

```
mtcars
```

Accedere a singole parti

```
mtcars["Datsun 710",]
```

```
mtcars[, "mpg"]
```

```
mtcars[, c("mpg", "gear")]
```