

Riepilogo sintassi elementare e Formule comuni in R

operatori relazionali:

== uguaglianza

!= diversità

<, <= minore, minore o uguale (idem per il maggiore)

& AND

| OR

! NOT

conversione verso un tipo specificato: "as." (character, integer, numeric, logical)

as.numeric("10")

verifica il tipo del dato

is.numeric(dato)

summary(dato)

#riepilogo caratteristiche dato (di qualsiasi tipo, anche struttura)

#####

vettori e matrici

#####

v <- c(2,3,4,5)

u<-seq(1,5,len=10, by= 2)

#creazione sequenza da valore iniziale, a

valore finale, di lunghezza 10 e di passo 2

z<-rep(v,2)

#duplica sequenza "v" per 2 volte

trova<-match(2,c(4,5,2,3,22,5))

#la funzione match individua uno o più

elementi all'interno di un vettore di componenti; il risultato È un vettore che indica la posizione che occupa la prima

occorrenza dell'elemento:

unique(v)

#individua elementi presenti per la prima volta

duplicated(v)

#individua elementi duplicati

which(condizione)

#seleziona elementi che verificano la condizione

W <- matrix(data = c(11,12,13,14,15,16), nrow = 2, ncol = 3, byrow = F)

#

matrice di data, organizzati per nrow, ncol indicati e ordinati per colonne/righe)

length(vet)

#lunghezza vettore

cbind(v, v1)

#costruisce matrice per colonne

rbind(v,v1)

#costruisce matrice per righe

trasforma dato in tipo Factor, indicando l'ordine dei valori ammessi

factor(dato, levels=c("med", "sup", "lau"))

```
#####
# data frame e data set
#####

record <- read.table('dati.txt', header=T)
hist(record$campo1, prob=TRUE)    #creazione istogramma. parametro prob:
grafico frequenze o densità (conteggio o area)
head (filedati)                    #contenuto primi 6 record del file dati
dim(filedati)                      #numero righe del file dati

#####
# statistica descrittiva
#####

mean(record)
min(record)
max(record)
var(record)
sd(record)

quantile(record,0.25)              #quantile del vettore ordinato (0.25 è il primo
quartile)

boxplot(record, horizontal=TRUE, main="boxplot records ",xlab="sec")
#grafico boxplot

plot(x,y)                          #grafico per punti
points(x,w, type="l", col=2)      #grafico sovrapposto, per linea
pie(dato)                          # grafico a torta
barplot(dato,xlab="province",ylab="proporzioni",main="Bar plot delle province")
#grafico a barre
hist(dato)                         #grafico istogramma
lines(x, y, col=2)                #grafico di una linea

# Il comando "table" conta il numero di elementi per ogni categoria, ovvero crea una
tabella delle frequenze assolute
FreqA <- table(dato)

# Apertura / chiusura file :

attach(dato)
detach(dato)

#####
# distribuzioni di probabilità
#####

d per la densità
```

p per la ripartizione
q per il quantile
r per la generazione di numero casuale

dnorm(x) calcola il valore della densita' in x
pnorm(x) calcola il valore della ripartizione in x.
qnorm(a) calcola l' a-quantile
rnorm(n) genera un campione di dimensione n da una normale standard .

Normal Probability Plot
qqnorm(dato)
qqplot(z,u,col="red", xlim=c(-3,3))
qqline(dato)

Statistica inferenziale
#####

t.test(x, mu = numero, conf.lev=0.99) #conf.level per default a 0.95
t.test(x, alternative="greater") #oppure alternative "less"

z.test(x, sigma) #richiede installazione pacchetto BSDA e attivazione col
comando library(BSDA)

altri test:

var.test(x,22,"less") #test varianza, esempio di test a una coda H1: varianza
minore di 22
prop.test(32,50,alternative="greater") #test sulle proporzioni, esempio a una
coda. Successi: 32 casi su 50 totali

t.test(m,f,alternative="greater") #test differenza tra medie
var.test(x,y,alternative="greater") #test differenza tra varianze

Regressione
#####

N.B.: i report di tutti i tipi di regressione li produciamo col comando
summary(regressione)

regr <- lm (costo ~ temperatura) #regressione lineare semplice

Il comando fitted() produce i punti stimati (mediante la retta)

fitted (regr)

predict(regr, newdata, interval="predict") #fornisce l'intervallo di predizione al
95%

```
resid(regr)
rstandard(reg)
```

```
#calcola i residui
#calcola i residui standardizzati
```

```
g=lm(sr~par1+par2+par3+par4, data=filedati)      #regressione lineare
multipla
```

```
lm(formula = data$peso ~ data$anni + data$specie + data$anni:data$specie)
#regressione multipla con variabili categoriche. Creare prima i subset per
specie del campo "specie" della struttura dati chiamata "data".
```

Esempio creazione di subset:

```
bianche<-subset(data, specie == "bianca", select = c("anni","peso"))
```

```
#####
### Test di Shapiro-Wilk ###
#####
```

```
# Il test di Shapiro-Wilk verifica l'ipotesi nulla che un campione
#  $y_1, \dots, y_n$  provenga da una popolazione distribuita normalmente
# La statistica W può essere interpretata come il quadrato del coefficiente
# di correlazione in un diagramma quantile-quantile.
# Il comando shapiro.test() restituisce il valore della statistica test
# di Shapiro-Wilk e il relativo p-value.
# Un p-value piccolo ( $p < 0.05$ ) indica una forte evidenza contro l'ipotesi nulla
# che i dati provengano da una distribuzione gaussiana.
```

```
shapiro.test(y)
```