

Einführung in R

<http://www.r-project.org>

Sprache und Arbeitsumgebung für statistische Berechnungen und Graphiken
Veröffentlicht unter GNU-Lizenz, angelehnt an S/S-Plus
Verfügbar für Linux, Mac OS X, Windows

Diverse GUIs: <http://www.r-project.org/GUI>

Diverse zusätzliche Pakete und Dokumentationen (Referenz, FAQ, Tutorien)

1 Grundlagen

Starten / Beenden

./R
q()

Hilfe

help.start	Html-Hilfe
help.search()	Suche in Hilfe-Einträgen
? <i>Befehl</i>	Direkte Hilfe
example(<i>Befehl</i>)	Beispiel
demo(<i>Demoname</i>)	Demo für einzelne Pakete

Zusätzliche Bibliothek (package)

library(*Paketname*)

Zuweisung

<- (lokal), <<- (global)

Datentypen (mode)

numeric (integer, double), complex, logical, character

Mathematische/ logische Operatoren

+, -, /, *, ^, ==, >, <, >=, <=, !=, &, |

Mathematische Funktionen

abs, exp, log, sin, cos, sqrt, max, min, sort, simplex, solve

Datensätze allgemein

<i>objekt</i> [<i>i,j</i>]	Zugriff auf Objekteintrag
<i>objekt</i> \$ <i>spaltenbeschriftung</i>	
dim(<i>x</i>) <- c(<i>i,j</i>)	Dimension zuweisen
length(<i>objekt</i>)	Länge

Vektoren

vector(mode=" <i>datentyp</i> ",length= <i>vektorlänge</i>), oder: <i>datentyp</i> (length= <i>vektorlänge</i>)	
c(<i>werte</i>)	Kombiniert Werte zu einem Vektor
seq(<i>von, bis, schrittweite</i>)	Erzeugt die angegebene Sequenz
append(<i>vektorname, werte</i>)	hängt Werte an den Vektor an
<i>startwert:endwert</i>	Erzeugt eine Sequenz mit Schrittweite 1

Felder

`array(Elemente, Dimension)`
`nrow(array), ncol(array)` Zeilenzahl, Spaltenzahl
`rownames(array) <- namensvektor` benennt die Zeilen (colnames für Spalten)

Matrizen

`matrix(Datenvektor, nrow=Zeilen, ncol=Spalten)`
`diag(x)` Erzeugt eine Einheitsmatrix
`eigen(matrix)` Eigenwerte, Eigenvektor
`t(matrix)` Transponierte Matrix

Ein- und Auslesen von Daten

`variable <- read.table("dateipfad", fill=TRUE)`
`write(variable, file="name", ncolumns=spalten)`

Kontrollfluss

`if(Bedingung) {Anweisungen} else {Anweisungen}`
`while(Bedingung) {Anweisungen}`
`for(var in seq) {Anweisungen}` Bsp.: `for(i in 1:10)`

Eigene Funktionen

`name <- function(Parameter) {Anweisungen return(Wert)}`
`name(Parameter)` Aufruf der Funktion
`source("/Pfad/zu/Datei/mult.r")` Funktion mult in Datei mult.r; wird vor Aufruf geladen

2 Graphik

Fenster

`x11(breite, höhe, punktgröße)` öffnet neues Graphikfenster
`par(mfrow=c(zeilenzahl, spaltenzahl))` Teilt Graphikfenster auf

High-Level-Zeichenbefehle (erzeugen neue Graphik)

`plot(variable, argumente)`
`hist(variable, argumente)` Histogramm: Anzahl von Werten in Datum
`pairs(x)` Matrix von Abb. für multivariate Daten
`pers(x, y, ...)` Daten mit versch. Farben darstellen
`contour(x, y, ...)`

Low-Level-Zeichenbefehle (ergänzen Graphik)

`points(x, y)`
`lines(x, y)`
`polygon(x, y, z, ...)`
`legend(locator(1), "legende", fill="farbe")` Legende einfügen

Parameter von High-Level-Graphiken

`log="y"` Logarithmische Skala auf der y-Achse
`type="z"` Darstellungstyp (Punkte, Linien, etc.);
`z=p|l|b|o|h|s|n`
`col="farbe"` Farbe
`xlim=wert, ylim=wert` maximale Werte
`expression(...)` Tex-formatierter Befehl, Syntax: s. plotmath
`add="TRUE"` Graphik verhält sich als Low-Level-Graphik

Speichern von Graphiken in verschiedenen Dateiformaten

`pdf|postscript("dateinamen", argumente)`
`jpeg|bmp|png(filename="dateinamen", argumente)`

3 Statistik

Kenngößen von Stichproben

<code>mean(vektor)</code>	Mittelwert
<code>median(vektor)</code>	Median
<code>var(vektor)</code>	Varianz
<code>sd(vektor)</code>	Standardabweichung
<code>skewness(vektor)</code>	Schiefe

Zufallszahlen

`sample(vektor, züge, replace=TRUE, prob=NULL)`

Verteilungen

<code>dverteilung(argumente)</code>	Dichte bzw. Wahrscheinlichkeitsfunktion
<code>pverteilung(argumente)</code>	Verteilungsfunktion
<code>qverteilung(argumente)</code>	Quantil (zu p wird x gegeben)
<code>rverteilung(argumente)</code>	Zufallszahlen

verteilung steht z. B. für:
`binom`(Binomialverteilung), `hyper`(Hypergeometrische Verteilung), `geom` (Geometrische Verteilung), `pois`(Poisson-Verteilung), `norm` (Normalverteilung), `unif` (Gleichverteilung)