

R Grundlagen

Schummelzettel

Hilfoptionen

Zugriff auf Hilfsdateien

?mean

Hilfe erhalten für eine bestimmte Funktion

help.search('weighted mean')

Hilfsdateien nach einem Wort oder Ausdruck durchsuchen

help(package = 'dplyr')

Hilfe für ein bestimmtes Package bekommen.

Objektinformationen

str(iris)

Zusammenfassung der Objektstruktur.

class(iris)

Auskunft über die Klasse des Objekts.

Verwendung von Packages

install.packages('dplyr')

Ein Package von CRAN herunterladen und installieren.

library(dplyr)

Lädt das Package in die Sitzung und macht die Benutzung seiner Funktionen möglich.

dplyr::select

Verwendet eine bestimmte Funktion aus einem Package.

data(iris)

Lädt einen eingebauten Datensatz.

Arbeitsverzeichnis

getwd()

Zeigt das aktuelle Arbeitsverzeichnis an.

setwd('C://ordner/pfad')

Verändert das aktuelle Arbeitsverzeichnis.

Verwenden Sie Projekte in RStudio, um das Arbeitsverzeichnis in den Ordner zu setzen, in dem Sie arbeiten möchten.

Vektoren

Vektoren erstellen

c(2, 4, 6)	2 4 6	Verbindet Elemente zu einem Vektor
2:6	2 3 4 5 6	Eine Zahlenfolge
seq(2, 3, by=0.5)	2.0 2.5 3.0	Eine komplexe Folge
rep(1:2, times=3)	1 2 1 2 1 2	Wiederholt einen Vektor
rep(1:2, each=3)	1 1 1 2 2 2	Wiederholt Elemente in einem Vektor

Funktionen für Vektoren

sort(x)

Gibt x sortiert zurück.

table(x)

Zeigt Anzahl der Elemente im Vektor.

rev(x)

Gibt x umgekehrt zurück.

unique(x)

Zeigt welche Werte vorkommen.

Vektorelemente auswählen

Positionsabhängig

x[4]	Das vierte Element.
x[-4]	Alle Elemente außer das vierte.
x[2:4]	Alle Elemente von Position 2 bis 4.
x[-(2:4)]	Alle Elemente außer von Position 2 bis 4.
x[c(1,5)]	Elemente an Position 1 und 5.

Werteabhängig

x[x == 10]	Elemente, die gleich 10 sind.
x[x < 0]	Alle Elemente, die kleiner als 0 sind.
x[x %in% c(1,2,5)]	Alle Elemente, die im Vektor genannt werden.

Benannte Vektoren

x['Apfel']	Elemente mit dem Namen 'Apfel'.
------------	---------------------------------

Programmieren

For-Schleife

```
for (Variable in Zahlensequenz){
  ...etwas machen...
}
```

Beispiel

```
for (i in 1:4){
  j <- i + 10
  print(j)
}
```

While-Schleife

```
while (Bedingung){
  ...etwas machen...
}
```

Beispiel

```
while (i < 5){
  print(i)
  i <- i + 1
}
```

If und Else

```
if (Bedingung){
  ...etwas machen...
} else {
  ...etwas anderes machen...
}
```

Beispiel

```
if (i > 3){
  print('i ist groesser als 3')
} else {
  print('i ist kleiner oder gleich 3')
}
```

Funktionen

```
Funktionsname <- function(variable){
  ...etwas machen...
  return(neue_variable)
}
```

Beispiel

```
quadrat <- function(x){
  ergebnis <- x*x
  return(ergebnis)
}
```

Daten einlesen und schreiben

Siehe **readr** package

Eingabe	Ausgabe	Beschreibung
df <- read.table('Datei.txt')	write.table(df, 'Datei.txt')	Liest und schreibt in eine Textdatei.
df <- read.csv('Datei.csv')	write.csv(df, 'Datei.csv')	Liest und schreibt eine kommagetrennte Datei. Dies ist ein Sonderfall von read.table/write.table.
load('Datei.RData')	save(df, file = 'Datei.Rdata')	Liest und schreibt eine R-Datei, ein Dateityp den R verwendet.

Bedingungen

a == b	Sind gleich	a > b	Größer als	a >= b	Größer oder gleich	is.na(a)	Fehlt a
a != b	Nicht gleich	a < b	Kleiner als	a <= b	Kleiner oder gleich	is.null(a)	Ist a NULL

Datentypen

Konvertierung zwischen üblichen Datentypen in R. Man kann immer von einem höheren Wert in der Tabelle zu einem niedrigeren Wert transformieren.

as.logical	TRUE, FALSE, TRUE	Logische Datentypen (WAHR oder FALSCH)
as.numeric	1, 0, 1	Ganze Zahlen oder Gleitkommazahlen
as.character	'1', '0', '1'	Zeichenketten. Häufig einfacher zu handhaben als factors.
as.factor	'1', '0', '1' Levels: 0, 1	Zeichenketten mit definierten Ebenen. Werden für manche statistischen Modelle genutzt.

Mathematische Funktionen

log(x)	Natürlicher Logarithmus	sum(x)	Summe
exp(x)	Exponentialfunktion	mean(x)	Mittelwert
max(x)	Größtes Element	median(x)	Median
min(x)	Kleinstes Element	quantile(x)	Quantile
round(x, n)	Runde auf n Dezimalstellen	rank(x)	Rang der Elemente
signif(x, n)	Runde auf n signifikante Stellen	var(x)	Varianz
cor(x, y)	Korrelation	sd(x)	Standardabweichung

Variablenzuweisung

```
> a <- 'apfel'
> a
[1] 'apfel'
```

Entwicklungsumgebung

ls()	Listet alle Variablen der Umgebung.
rm(x)	Entfernt x aus der Umgebung.
rm(list = ls())	Entfernt alle Variablen aus der Umgebung.

Sie können in der Registerkarte "Environment" in RStudio ihre Variablen einsehen

Matrizen

```
m <- matrix(x, nrow = 3, ncol = 3)
Erstellt eine Matrix aus x.
```



m[2,] Wählt eine Zeile aus
m[, 1] Wählt eine Spalte aus
m[2,3] Wählt ein Element aus

t(m) Transponiert Matrix m.
m %*% n Matrixmultiplikation
solve(m, n) Findet x sodass: $m * x = n$

Listen

```
l <- list(x = 1:5, y = c('a', 'b'))
```

Eine Liste ist eine Sammlung von Elementen. Dabei kann die Liste aus verschiedenen Datentypen bestehen.

l[[1]]	l[2]	l\$x	l['y']
Inhalt des ersten Elements von Liste l	Zweites Element von Liste l.	Inhalt des Elements mit Namen x von Liste l.	Element mit Namen y von Liste l.

Datensätze

Siehe **dplyr** Package

```
df <- data.frame(x = 1:3, y = c('a', 'b', 'c'))
```

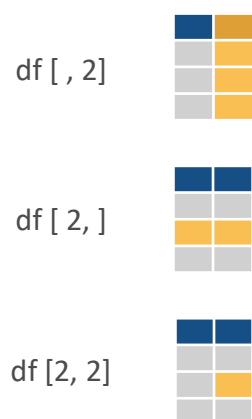
Ein besonderer Fall der Liste, in der alle Elemente die selbe Länge haben.

x	y
1	a
2	b
3	c

Listen Untergruppen



Matrix Untergruppen

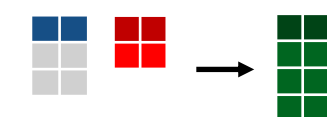


nrow(df) Anzahl der Zeilen
ncol(df) Anzahl der Spalten
dim(df) Anzahl der Spalten und Zeilen

cbind - Verbindet Spalten.



rbind - Verbinde Zeilen.



Datensatz anzeigen

View(df) Zeigt den ganzen Datensatz.
head(df) Zeigt die ersten 6 Zeilen.

Zeichenketten

Siehe **stringr** Package

paste(x, y, sep = ' ')	Verbindet mehrere Vektoren miteinander.
paste(x, collapse = ' ')	Verbindet Elemente eines Vektors miteinander.
grep(pattern, x)	Findet den regulären Ausdruck in x.
gsub(pattern, replace, x)	Ersetzt Übereinstimmungen in x mit einer Zeichenkette.
toupper(x)	Konvertiert zur Großschreibung.
tolower(x)	Konvertiert zur Kleinschreibung.
nchar(x)	Anzahl an Zeichen in der Zeichenkette.

Faktoren

factor(x)	Verwandelt einen Vektor in einen Faktor. Kann die Ebenen des Faktors und die Reihenfolge bestimmen.
cut(x, breaks = 4)	Verwandelt einen numerischen Vektor in einen Faktor, indem er in Bereiche „geschnitten“ wird.

Statistik

lm(y ~ x, data=df) Lineares Modell	t.test(x, y) Führt einen t-Test für den Unterschied zwischen Mittelwerten durch.	prop.test Testet nach einem Unterschied zwischen Verhältnissen.
glm(y ~ x, data=df) Generalisiertes lineares Modell	pairwise.t.test Führe einen t-Test für paarweise Datensätze durch.	aov Führt eine Varianzanalyse durch.
summary Erhalte detailliertere Informationen über das Modell		

Verteilungen

	Zufallsvariable	Dichtefunktion	Kumulative Verteilung	Quantil
Normal-	rnorm	dnorm	pnorm	qnorm
Poisson-	rpois	dpois	ppois	qpois
Binomial-	rbinom	dbinom	pbinom	qbinom
Gleich-	runif	dunif	punif	qunif

Grafik

Siehe **ggplot2** Package

plot(x) Werte von x der Reihe nach. 	plot(x, y) Werte von x im Vergleich zu y. 	hist(x) Histogramm von x.
---	---	---

Datum / Daten

Siehe **lubridate** Package