

| Functions | Description | Example | Screenshot / Result | Where did I see it? |
|-----------|--|--|--|---------------------|
| sqrt() | Math functions, which is useful to find the square root in R for an individual number or an expression. | sqrt(16) + (10 + 3) | $\sqrt{16} + (10 + 3)$ | Arithmetics |
| | | 3 + (27^(1/3) - 8) | $3 + (\sqrt[3]{27} - 8)$ | |
| c() | "c" stands for combine It combines numeric values into vectors. | c(2, 10, 13) | 2 10 13 | |
| | | c(10:15) | 10 11 12 13 14 15 | Vectors |
| c() | Accessing elements of a vector can be done via indexing, which uses squared brackets. | v.age <- c(15, 17, 34, 6, 101) v.age[2] | 17 | Vectors |
| seq() | To make bigger or smaller steps in a sequence. | seq(from = 1, to = 10, length.out = 5), ## same as seq(from = 1, to = 10, leng = 5) | | Vectors |
| c() | if length of vectors does'nt match (recycling) | c(10,20,30) + 1:6 = 11 22 33 14 25 36 | | Vectors |
| methods() | It lists all known methods for the generic function of interest. | methods(plot) | plot.acf* plot.density* plot.hclust* plot.medpolish* plot.princomp* plot.stl* | Random |
| rep() | It allows the user to create vector in which a value or some values are replicates. | rep(x = 1:3, times = 3) | 1 2 3 1 2 3 1 2 3 | Vectors |
| matrix() | Create a matrix. | matrix(1:10, ncol = 5) | [,1] [,2] [,3] [,4] [,5] [1,] 1 3 5 7 9 [2,] 2 4 6 8 10 | Matrices |
| rbind() | Take a sequence of vector, matrix or data-frame arguments and combine by columns or rows, respectively. Also used to add columns to an existing matrix. | rbind(1:5, 6:10) | [,1] [,2] [,3] [,4] [,5] [1,] 1 2 3 4 5 [2,] 6 7 8 9 10 | Matrices |
| cbind() | Take a sequence of vector, matrix or data-frame arguments and combine by columns or rows, respectively. Also used to add columns to an existing matrix. | cbind(1:5, 6:10) | [,1] [,2] [1,] 1 6 [2,] 2 7 [3,] 3 8 [4,] 4 9 [5,] 5 10 | Matrices |
| diag() | Extract or replace the diagonal of a matrix, or construct a diagonal matrix. | diag(nrow = 4) | [,1] [,2] [,3] [,4] [1,] 1 0 0 0 [2,] 0 1 0 0 [3,] 0 0 1 0 [4,] 0 0 0 1 | Matrices |

letters;
Letters Output are the 26 lower/ upper-case letters of the Roman alphabet.

data.frame() Function creates a data frame.

class() Output type.

colnames()
rownames() Change column names.
Change row names.

make.names() Make syntactically valid names out of character vectors.
Also To «check» if name is illegal and what is the output.
identical() The safe and reliable way to test two objects for being *exactly* equal.

all.equal() Compares R objects x and y testing 'near equality'.

list() Creates a list.

is.list() Checks is it is a list.

rm() Removes object.

ls() Lists all the variables created in own enviroment.

rnorm() Generates multivariate normal random variates.

args() Displays the argument names and corresponding default values of a function or primitive.

letters

d <- data.frame(x = 1, y = 1:4)

class(d)

colnames(df)<-
c("EmployeeID", "Salary")

make.names(c("a and b", "a_and_b"))

```
[1] "a" "b" "c" "d" "e" "f"  
[2] "g" "h" "i" "j" "k" "l" "m"  
[3] "n" "o" "p" "q" "r" "s" "t"  
[4] "u" "v" "w" "x" "y"  
[5] "z"
```

```
  x y  
1 1 1  
2 1 2  
3 1 3  
4 1 4
```

```
"data.frame"
```

```
> df <- data.frame(ID=1:2,Salary=c(10000,30000))  
> df  
  ID Salary  
1  1 10000  
2  2 30000  
>  
> colnames(df)<-c("EmployeeID","Salary")  
> df  
  EmployeeID Salary  
1           1 10000  
2           2 30000  
>  
> rownames(df)<-c("001","025")  
> df  
  EmployeeID Salary  
001         1 10000  
025         2 30000
```

Data frames
Data frames
/ Vector

```
"a. and. b" "a_and_b"
```

```
> identical(1, 1.)  
[1] TRUE
```

```
> all.equal(gamma(2:14), cumprod(1:13))  
[1] TRUE
```

```
> x <- list(1)  
> x  
[[1]]  
[1] 1  
> is.list(x)  
[1] TRUE
```

```
> tmp <- 1:4  
> tmp  
[1] 1 2 3 4  
> rm(tmp)  
> tmp  
Fehler: objekt 'tmp' nicht gefunden  
> ls()  
[1] "a"  
[5] "AirPassengers"  
[9] "Arabs"
```

Matrices

Random

Random

Fitting Models

Lists

Lists

Vector / Function

Function

```
> rnorm(2)  
[1] 0.2436874 1.2324759
```

```
> args(plot)  
function (x, y, ...)
```

Random

Random

`apropos()` Returns a character vector giving the names of objects in the search list matching (as a regular expression).

`getwd()` Get working directory.

`setwd()` Change/ set working directory.

`view()` Shows data set in extra window.

`table()` Table function in R -`table()`, performs categorical tabulation of data with the variable and its frequency. `Table()` function is also helpful in creating Frequency tables with condition and cross tabulations.

`unique()` Returns a vector, data frame or array but with duplicate elements/rows removed.

`all.is.numeric()` Tests, without issuing warnings, whether all elements of a character vector are legal numeric values, or optionally converts the vector to a numeric vector.

`as.numeric()` Converts numbers into a numeric value.

`assert()` Emits a message in case of errors, which can be a helpful hint for diagnosing the errors.

`dim()` Retrieve or set the dimension of an object.

`nrow()`
`ncol()` Return the number of rows or columns present in x.

`nlevels()` Return the number of levels which its argument has.

`levels()` Provides access to the levels attribute of a variable.

`pairs()` A matrix of scatterplots is produced.

```
> apropos("plot")
[1] ".acfPlot"
[4] ".contourPlot"
[7] ".distFitPlot"
[10] ".thplot"
```

Packages
Import data
Import data
Import data

```
> table(iris[, "Species"])
      setosa versicolor  virginica 
        50         50         50
```

Inspect data

```
> unique(iris)
      Sepal.Length Sepal.Length Sepal.Length
1              5.1              5.1              5.1
2              4.9              4.9              4.9
3              4.7              4.7              4.7
```

Inspect data

```
> rv <- c("-0.1", " 2.7 ", "3")
> as.numeric(rv)
[1] -0.1  2.7  3.0
```

Inspect data

Random

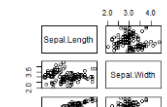
```
> x <- 1:6 ; dim(x) <- c(2,3)
> x
      [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

Inspect data

Inspect data

Inspect data

Inspect data



Graph

palette() View or manipulate the color palette.

```
> palette()
[1] "black" "#DF536B" "#6
```

Graph

colours() Returns the built-in color names which R knows about.

```
colours()
[1] "white"
[7] "antiquewhite4"
[13] "azure"
[19] "bisque"
```

Graph

abline() This function adds one or more straight lines through the current plot.

Graph

par() Par can be used to set or query graphical parameters. Parameters can be set by specifying them as arguments to par in tag = value form, or by passing them as a list of tagged values.

Graph

aes() Aesthetic mappings describe how variables in the data are mapped to visual properties (aesthetics) of geoms

Graph

jitter() Add a small amount of noise to a numeric vector.

```
> jitter(rep(0, 3))
[1] 0.019864689 -0.010607213 0.004506879
```

Random

dev.off() Closes the specified plot (by default the current device) and if it is an imguR device, uploads the plots for web hosting. Also get back to default values (e.g. after par()).

Graph

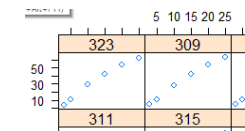
demo() Runs some demonstration R scripts.

demo(graphics)

Random

xyplot Creates an XY plot from the first two columns of a dataframe/matrix, or from two separate vectors of numeric values.

xyplot(height ~ age | Seed, data = Loblolly)



Graph (lattice)

locator() Reads the position of the graphics cursor when the (first) mouse button is pressed. Supported on screen devices such as X11, windows and quart.

Random

sessionInfo() Print version information about R, the OS and attached or loaded packages.

Packages

lmer() Fit a linear mixed-effects model (LMM) to data, via REML or maximum likelihood.

Packages

| | | | |
|------------------|---|---|-----------------|
| .libPaths() | Gets/sets the library trees within which packages are looked for. | | Packages |
| packageVersion() | Will report the version number of a requested installed package | | Packages |
| anyNA() | Test whether there are any missing values. | <pre>> anyNA(airquality) [1] TRUE</pre> | Missing value |
| is.na() | Check if an object is NA. Always return TRUE or FALSE, a logical vector of length one. | <pre>is.na(airquality\$ozone) [1] FALSE FALSE FALSE F [21] FALSE FALSE FALSE F</pre> | Missing value |
| na.omit() | The na.omit R function removes all incomplete cases of a data object (typically of a data frame, matrix or vector) | | Missing value |
| apply() | Returns a vector or array or list of values obtained by applying a function to margins of an array or matrix. | <pre>> apply(airquality, MARGIN = 2, FUN = class) Ozone Solar.R Wind Temp Wt "numeric" "numeric" "numeric" "numeric" "nume</pre> | Missing value |
| ggplotly() | Create plotly graphs using ggplot2 syntax. | | Graph |
| with() | Evaluate an R expression in an environment constructed from data, possibly modifying (a copy of) the original data. | <pre>with(mtcars, mpg[cyl == 8 & disp > 350]) [1] 18.7 14.3 10.4 10.4 14.7 19.2 15.8</pre> | Data Enviroment |
| find() | Finds indices of nonzero elements. | <pre>find(-3:3 >= 0) Fehler in find(-3:3 >= 0) : is.character(what) is not TRUE</pre> | Data Enviroment |
| find("") | Shows in which package the function is in. | <pre>> find("plot") [1] "package:graphics" "package:base"</pre> | Packages |
| filter() | Selects rows based on their values | <pre>starwars %>% filter(species == "Droid")</pre> | Dplyr |
| mutate() | Adds new variables that are functions of existing variables | <pre>starwars %>% mutate(name, bmi = mass / ((height / 100) ^ 2)) %>% select(name:mass, bmi)</pre> | Dplyr |
| select() | Picks variables based on their name | <pre>starwars %>% select(name, ends_with("color"))</pre> | Dplyr |
| summarise() | Calculates summary statistics | <pre>mtcars %>% summarise(mean = mean(displ), n = n())</pre> | Dplyr |
| arrange() | Sorts the rows | <pre>starwars %>% arrange(desc(mass))</pre> | Dplyr |
| replace() | Replace missing values | <pre>d.testNAs_2 %>% replace_na(list(Age = 100, Salary = 0))</pre> | Dplyr |

Compares are shows the difference.

setdiff()

as.date()

Converts between character representations and objects of class "Date" representing calendar dates.

```
> find("setdiff")
[1] "package:base"
> setdiff(1:4, 2)
[1] 1 3 4
> setdiff(1:4, 2:3)
[1] 1 4
> setdiff(1:4, 1:4)
integer(0)
```

Random

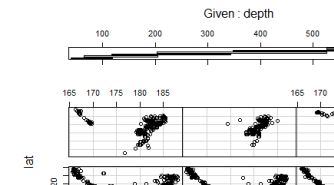
Random

Produces two variants of the conditioning plots.

coplot()
left_join()
right_join()

Join two datasets.

coplot(lat ~ long | depth, data = quakes)



Graph

Joining data sets

Shows methods and classes for R objects, plus other programming tools, as described in the reference.

method()

```
methods(plot)
[1] plot.ANY-method
[6] plot.boot*
[11] plot.decomposed.ts*
[16] plot.factor*
```

Generic and methods functions

nchar()

Number of characters

nchar(v.char)

```
nchar(v.char)
.] 4 5 14 8
```

Regular expressions

substr()

Get first three characters

substr(v.char, start = 1, stop = 3)

```
> substr(v.char, start = 1, stop = 3)
[1] "Ann" "Ann" "Joh" "12 "
```

Regular expressions

grepl()

Contains, starts, ends... with certain "object".

```
grepl(v.char, pattern = "c")
gsub(v.char, pattern = "Anna",
replacement = "Annamaria")
[1] "Annamaria" "Annamaria "
"Johnny Cashout" "12 Luc @"
```

```
grepl(v.char, pattern = "c")
[1] FALSE FALSE FALSE TRUE
```

Regular expressions

gsub()

Substitute some text

Regular expressions