Base R Cheat Sheet

Getting Help

Accessing the help files

?mean

Get help of a particular function.

help.search("weighted mean")

Search the help files for a word or phrase.

help(package = "dplyr")

Find help for a package.

More about an object

class(iris)

Find the class an object belongs to.

Using Libraries

install.packages("haven")

Download and install a package from CRAN.

library(haven)

Load the package into the session, making all its functions available to use.

haven::read_dta

Use a particular function from a package.

data(iris)

Load a build-in dataset into the environment.

Working Directory

getwd()

Find the current working directory (where inputs are found and outputs are sent).

setwd("path")

Change the current working directory.

Use projects in RStudio to set the working directory to the folder you are working in.

Vectors

Creating Vectors

c(2, 4, 6)	2 4 6	Join elements into a vector
2:6	2 3 4 5 6	An integer sequence
seq(2, 3, by = 0.5)	2.0 2.5 3.0	A complex sequence
rep(1:2, times = 3)	1 2 1 2 1 2	Repeat a vector
rep(1:2, each = 3)	1 1 1 2 2 2	Repeat elements of a vector

Vector Functions

sort(x)

Return x sorted

rev(x)
Return x reversed.

unique(x)

See unique values.

Selecting Vector Elements

By Position

x[4] The fourth element.

x[-4]

All but the fourth.

x[2:4]

Elements two to four.

x[-(2:4)]

All elements except two to four.

x[c(1, 5)]

Elements one and five.

By Value

x[x == 10]

Elements which are equal to 10.

x[x < 0]

All elements less than zero.

x[x %in% c(1, 2, 5)] Elements in the set 1, 2, 5.

Named Vectors

x["apple"]

Element with name "apple".

Programing

For Loop

```
for (variable in sequence) {
   Do something
}
```

Example

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

While Loop

```
while (condition) {
   Do something
}
```

Example

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

If Statements

```
if (condition) {
   Do something
} else {
   Do something different
}
```

Example

```
if (i > 3) {
    print("Yes")
} else {
    print("No")
}
```

Functions

```
function_name <- function(arg) {
   Do something
}</pre>
```

Example

```
square <- function(x) {
   x * x
}</pre>
```

Conditions

a == b a != b Are equal a > b

Not equal a < b

b Greater than
b Less than

a >= b

e b or Les

or equal to

Less than or equal to

is.na(

is.null

is.na(a) Is missing
is.null(a) Is null

Types

Converting between common data types in R. Can always go from a higher value in the table to a lower value.

as.logical	TRUE, FALSE, TRUE	Boolean values (TRUE or FALSE).	
as.integer	1L, 0L, 1L	Integers.	
as.double	1, 0, 1	Floating point numbers.	
as.character	"1", "0", "1"	Character strings. Generally preferred to factors.	
as.factor	"1", "0", "1", Levels: "0", "1"	Character strings with set levels.	

Math Functions

log(x)	Natural log.	sum(x)	Sum.
exp(x)	Exponential.	mean(x)	Mean.
max(x)	Largest element.	median(x)	Median.
min(x)	Smallest element.	quantile(x)	Percentage quantiles.
round(x, n)	Round to n decimal places.	rank(x)	Rank of elements
		var(x)	The variance.

Variable Assignment

Correlation.

sd(x)

The standard deviation.

cor(x, y)

> a <- "apple" > a [1] "apple"

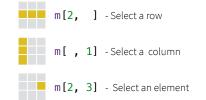
The Environment

ls() List all variables in the environment. rm(x)Remove x from the environment.

You can use the environment panel in Studio to browse variables in your environment.

Matrixes

 $m \leftarrow matrix(x, nrow = 3, ncol = 3)$ Create a matrix from x.



Lists

 $l \leftarrow list(x = 1:5, y = c("a", "b"))$

A list is collection of elements which can be of different types.

1[[2]] Second element of l.

New list with only the first element.

1[1]

Element named

1\$x

New list with only element named y.

l["y"]

t(m)

Transpose

Also see the **dplyr** library.

Data Frames

 $df \leftarrow data.frame(x = 1:3, y = c("a", "b", "c"))$ A special case of a list where all elements are the same length.

х	у
1	а
2	b
3	С



head(df)

df[[2]] df\$x Understanding a data frame

List subsetting

See the full data View(df) frame. See the first 6

rows.

Matrix subsetting

df[, 2] nrow(df) Number of rows. ncol(df) df[2,] Number of columns.

df[2, 2]

dim(df) Number of columns and rows.

Strings

See the **stringr** library.

Factors

factor(x)

cut(x, breaks = 4)

Turn a vector into a factor. Can set the levels of the factor and the order.

Turn a numeric vector into a factor by "cutting" into sections.

Statistics

 $lm(x \sim y, data = df)$ Linear model.

 $glm(x \sim y, data = df)$ Generalized linear model.

summary(model) Get more detailed information out a model.

Distributions

	Random Variates	Density Function	Cumulative Distribution	Quantile
Normal	rnorm	dnorm	pnorm	qnorm
Poison	rpois	dpois	ppois	qpois
Binomial	rbinom	dbinom	pbinom	qbinom
Uniform	runif	dunif	punif	qunif

Plotting

See the **ggplot2** library.

Dates

See the **lubridate** library.