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

str(iris)

Get a summary of an object's structure.

class(iris)

Find the class an object belongs to.

Using Packages

install.packages('dplyr')

Download and install a package from CRAN.

library(dplyr)

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

dplvr::select

Use a particular function from a package.

data(iris)

Working Directory

getwd()

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

setwd('C://file/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)
                                               into a vector
                                               An integer
2:6
                                               sequence
                                               A complex
seq(2, 3, by=0.5)
                                               sequence
rep(1:2, times=3)
                                             Repeat a vector
                                             Repeat elements
rep(1:2, each=3)
                                               of a vector
```

Vectors Functions

sort(x) rev(x)

Return x sorted. Return x reversed.

table(x)unique(x)

See counts of values. See unique values.

Selecting Vector Elements

Bv Position

The fourth element. x[4]

All but the fourth. $\times [-4]$

Elements two to x[2:4]four.

All elements except $\times [-(2:4)]$ two to four.

Elements one and x[c(1, 5)]five.

By Value

Element which are x[x == 10]equal to 10.

Element which are x[which(x==10)]equal to 10.

> All elements less x[x < 0]than zero.

x[x %in% c(1,2,Elements in the set $\{1, 2, 5\}.$ 5)1

Named Vectors

Element with name x['apple'] 'apple'.

Programming

For Loop

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

```
for (i in 1:4) {
    print(j)
```

Functions

While Loop

Example

while (condition) {

Do something

while (i < 5) {

print(i)

```
funct_name <- function(var) {</pre>
    Do something
    return (new_variable)
```

Example

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

If Statement

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

Example

```
print('Yes')
print('No')
```

Reading and Writing Data Also see the **readr** package.

Input	Output	Description
<pre>df <-read.table('file .txt')</pre>	write.table(df, 'file .txt')	Read and write a delimited text file.
df <-read.csv('file.	write.csv(df, 'file.	Read and write a comma separated value file. This is a special case of
		read.table/write.table.
load('file.RData')	<pre>save(df, file = 'file .RData')</pre>	Read and write a n R data file, a file type special for R.

Conditions	
Conditions	l

a == b	Are equal	a > b	Greater than	a >= b	Greater than or equal to	is.na(a)	Is missing
a !=b	Not equal	a < b	Less than	a <= b	Less than or equal to	is.null(a)	Is null
c e	c or e	C &&y	c and y		- 4		

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.numeric	1, 0, 1	Integer or floating point numbers
as.integer	1, 0, 1	Integers
as.character	'1', '0', '1'	Character strings. Generally preferred to factors
as.factor	'1', '0', '1' levels: '1', '0'	Character strings with preset levels. Needed for some statistical models

Maths 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.	
signif(x, n)	Round to n sig- nificant figures.	var(x)	The variance.	
cor(x, y)	Correlation.	sd(x)	The standard de- viation.	

Variable Assignment

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

The Environment

ls ()	List all variables in the environment.
rm(x)	Remove x from the environment.
rm (list = ls ())	Remove all variables from the envi- ronment.

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

Lists

1 < -1 ist (x = 1:5, y = c('a', 'b'))

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

1[[2]] Second element of I.

1[1] New list with only the first 1\$x 1['y']

Element named x. element.

New list with only element named v.

Miscellaneous

Arithmetics

16 = 3*5 + 116%/%3 16%3

Quotient (result 5). Remainder (result 1).

Permutations

```
sample(x, size, replace =
                                  Give a sample of the specified size from
                                           the elements of x.
              F)
           sample(c(1:5), 10, replace = TRUE)
sort(x, decreasing = F)
                                         Sort a vector or factor.
             sort (c(5, 1, 7, 3)) (Result 1 3 5 7)
                                   Returns a permutation rearranging x.
order(x, decreasing = F)
            order(c(5, 1, 7, 3)) (Result 3 1 4 2)
```

Function tests

replicate(n, f(x))Execute n times the function f(x). replicate $(10, \exp(1000000))$ Return the CPU times used to compute f(x). system.time(f(x))system.time(exp(1000000))

Memo

Strings

grep(pattern, x)

Also see the stringr package.

cat(x, y, sep = '') Join and print multiple vectors together. cat(x, collapse = '') Join and print elements of a vector together.

Find regular expression matches in x.

gsub(pattern, replace, x) Replace matches in x with a string.

> toupper(x) Convert to uppercase.

> tolower(x) Convert to lowercase.

nchar(x) Number of characters in a string.

Factors

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

cut(x, breaks = 4)Turn a numeric vector into a factor by 'cutting' into sections.

Statistics

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

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

summary Or fivenum Get more detailed information out a model.

t.test(x, y) Perform a t-test for difference between means.

pairwise.t.test Perform a t-test for paired data.

prop.test Test for a difference between proportions.

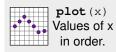
aov Analysis of variance.

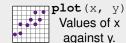
Distributions

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

Plotting

Also see the **ggplot2** package.







hist(x) Histogram of x.