R Cheatsheet

Copyright 2014, Bruce Beauchamp

Version 1.0 13 Aug 2014

Contents

[R Cheatsheet 1](#_Toc396189346)

[Control statements 3](#_Toc396189347)

[Iterating through a vector 3](#_Toc396189348)

[Factors 3](#_Toc396189349)

[Convert items to factors 3](#_Toc396189350)

[data.frames 4](#_Toc396189351)

[Add a column to a data.frame 4](#_Toc396189352)

[Append a row to a data.frame 4](#_Toc396189353)

[Column Sums 4](#_Toc396189354)

[Create a data.frame from scratch 4](#_Toc396189355)

[Get a cell of data from a data.frame 5](#_Toc396189356)

[Get a column of data from a data.frame 5](#_Toc396189357)

[Get the column names of a data.frame 5](#_Toc396189358)

[Get the number of rows in a data.frame 5](#_Toc396189359)

[Get a Row of Data from a data.frame 5](#_Toc396189360)

[Get the rows containing a string or part of a string 5](#_Toc396189361)

[Group by rows and apply a function to the groups 6](#_Toc396189362)

[Missing data 6](#_Toc396189363)

[Order a data.frame based on a column 6](#_Toc396189364)

[Descending order 7](#_Toc396189365)

[Select Rows which meet criteria 7](#_Toc396189366)

[Select Columns from a data.frame 7](#_Toc396189367)

[Data Types 8](#_Toc396189368)

[Get the type of an object 8](#_Toc396189369)

[Environment 9](#_Toc396189370)

[Change the Working Directory 9](#_Toc396189371)

[Get the Working Directory 9](#_Toc396189372)

[List the Files in the Working Directory 9](#_Toc396189373)

[Load a Function into the Environment 9](#_Toc396189374)

[Load a library 9](#_Toc396189375)

[List the Functions Available in the Environment 9](#_Toc396189376)

[Matrices 9](#_Toc396189377)

[Create a matrix 9](#_Toc396189378)

[Misc 10](#_Toc396189379)

[Printing a variable to the console 10](#_Toc396189380)

[Reading Data 11](#_Toc396189381)

[Read a CSV file into a data.frame 11](#_Toc396189382)

[Strings 11](#_Toc396189383)

[Concatenate two strings 11](#_Toc396189384)

[Length of a string 11](#_Toc396189385)

[Padding a string with zeros 11](#_Toc396189386)

[Replace all the characters in a string 11](#_Toc396189387)

[Replace characters in a string 11](#_Toc396189388)

[Trim a string 12](#_Toc396189389)

[Vectors 13](#_Toc396189390)

[Creating a vector 13](#_Toc396189391)

[Iterating over a vector 13](#_Toc396189392)

[Length of a vector 13](#_Toc396189393)

[Return valid elements of a vector 13](#_Toc396189394)

# Control statements

## Iterating through a vector

id = 1:3

for (i in id) {

cat(i, "\n")

}

|  |
| --- |
| 1  2  3 |
|  |
| |  | | --- | |  | |

# data.frames

## Add a column to a data.frame

> x

var1 var2 var3

1 2 8 11

2 5 9 15

3 3 6 13

4 1 10 12

5 4 7 14

> x$var4 <- rnorm(5)

> x

var1 var2 var3 var4

1 2 8 11 1.4959815

2 5 9 15 1.1187864

3 3 6 13 0.5330004

4 1 10 12 -0.3492245

5 4 7 14 0.1875488

## Append a row to a data.frame

> df

id nobs

1 1 2

> df <- rbind(df, data.frame(id=17, nobs=34))

> df

id nobs

1 1 2

2 17 34

## Column Sums

|  |
| --- |
| > x  var1 var2 var3 var4  1 2 8 11 1.4959815  2 5 NA 15 1.1187864  3 3 6 13 0.5330004  4 1 NA 12 -0.3492245  5 4 7 14 0.1875488  > colSums(x)  var1 var2 var3 var4  15.000000 NA 65.000000 2.986093 |
|  |
| |  | | --- | |  | |

## Create a data.frame from scratch

> df <- data.frame(id=integer(), nobs=integer())

> df <- rbind(df, data.frame(id=1, nobs=2))

> df

id nobs

1 1 2

|  |
| --- |
| > x <- data.frame("var1"=sample(1:5), "var2"=sample(6:10), "var3"=sample(11:15))  > x  var1 var2 var3  1 2 8 11  2 5 9 15  3 3 6 13  4 1 10 12  5 4 7 14 |
|  |
| |  | | --- | |  | |

## Get a cell of data from a data.frame

> pollutantData[1,]

Date sulfate nitrate ID

1 2003-01-01 NA NA 1

> pollutantData[1,"ID"]

[1] 1

## Get a column of data from a data.frame

> x

var1 var2 var3

1 2 8 11

2 5 9 15

3 3 6 13

4 1 10 12

5 4 7 14

> x[,1]

[1] 2 5 3 1 4

> x[,"var2"]

[1] 8 9 6 10 7

## Get the column names of a data.frame

> colnames(outcomeData)

## Get the number of rows in a data.frame

|  |
| --- |
| > nrow(data1)  [1] 153 |
|  |
| |  | | --- | |  | |

## Get a Row of Data from a data.frame

|  |
| --- |
| data1[1,]  Ozone Solar.R Wind Temp Month Day  1 41 190 7.4 67 5 1 |

## Get the rows containing a string or part of a string

> df

city

1 Sacramento

2 Auburn

3 Sacramento

> grep("Sacramento", df$city)

[1] 1 3

> grep("Sacrament", df$city)

[1] 1 3

> grep("Sacrament", df$city, value=TRUE)

[1] "Sacramento" "Sacramento"

## Group by rows and apply a function to the groups

> head(InsectSprays)

count spray

1 10 A

2 7 A

3 20 A

4 14 A

5 14 A

6 12 A

> tapply(InsectSprays$count, InsectSprays$spray, sum)

A B C D E F

174 184 25 59 42 200

## Identify rows meeting some criteria

> which(with(housing\_data, ACR == 3 & AGS == 6))

[1] 125 238 262 470 555 568 608 643 787 808 824 849 952 955 1033 1265 1275

[18] 1315 1388 1607 1629 1651 1856 1919 2101 2194 2403 2443 2539 2580 2655 2680 2740 2838

[35] 2965 3131 3133 3163 3291 3370 3402 3585 3652 3852 3862 3912 4023 4045 4107 4113 4117

[52] 4185 4198 4310 4343 4354 4448 4453 4461 4718 4817 4835 4910 5140 5199 5236 5326 5417

[69] 5531 5574 5894 6033 6044 6089 6275 6376 6420

## Merge two data.frames

merged <- merge(gdp\_data, edstats\_data, by.x="Country", by.y="CountryCode", all=TRUE)

## Missing data

> x

var1 var2 var3 var4

1 2 8 11 1.4959815

2 5 NA 15 1.1187864

3 3 6 13 0.5330004

4 1 NA 12 -0.3492245

5 4 7 14 0.1875488

> sum(is.na(x$var2))

[1] 2

## Order a data.frame based on a column

> x

var1 var2 var3

1 2 8 11

2 5 9 15

3 3 6 13

4 1 10 12

5 4 7 14

> x[order(x$var1, x$var3),]

var1 var2 var3

4 1 10 12

1 2 8 11

3 3 6 13

5 4 7 14

2 5 9 15

> library(plyr)

> arrange(x, var2)

var1 var2 var3

1 3 6 13

2 4 7 14

3 2 8 11

4 5 9 15

5 1 10 12

### Descending order

> arrange(x, desc(var2))

var1 var2 var3

1 1 10 12

2 5 9 15

3 2 8 11

4 4 7 14

5 3 6 13

|  |  |
| --- | --- |
| |  | | --- | |  | |

## Select Rows which meet criteria

> test <- iris[iris$Species=="virginica", ]

> test

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

101 6.3 3.3 6.0 2.5 virginica

102 5.8 2.7 5.1 1.9 virginica

103 7.1 3.0 5.9 2.1 virginica

## Select Columns from a data.frame

outcomeData <- outcomeData[, c(2, 7, column)]

Here, columns 2, 7, and one additional column are retained

# Data Types

## Get the type of an object

|  |
| --- |
| > x <- 4  > class(x)  [1] "numeric" |
|  |
|  |

# Dates and Times

## Converting strings to dates

> as.Date(c('4jul1776'), "%d%b%Y")

[1] "1776-07-04"

> library(lubridate)

> ymd("17760704")

[1] "1776-07-04 UTC"

> mdy("07/04/1776")

[1] "1776-07-04 UTC"

> ymd\_hms("2014-08-19 06:22:54", tz="America/Los\_Angeles")

[1] "2014-08-19 06:22:54 PDT"

## Current date

|  |
| --- |
| > Sys.Date()  [1] "2014-08-19" |
| Difference between dates (in days)  |  | | --- | | > date2  [1] "1776-07-07"  > date1  [1] "1776-07-04"  > as.numeric(date2-date1)  [1] 3 | |  | | |  | | --- | |  | | |
| |  | | --- | |  | |

## Formatting dates

> today

[1] "2014-08-19"

> format(today, "%a %b %d")

[1] "Tue Aug 19"



# Environment

## Change the Working Directory

> setwd("c:/r/Prog3")

> getwd()

[1] "c:/r/Prog3"

## Get the Working Directory

> getwd()

[1] "C:/Users/Bruce/Documents"

>

## List the Files in the Working Directory

dir()

## Load a Function into the Environment

source(“myCode.R”)

## Load a library

> library(stringr)

## List the Functions Available in the Environment

ls()

# Factors

## Convert items to factors

|  |
| --- |
| > y <- c(95647, 95602)  > y  [1] 95647 95602  > str(y)  num [1:2] 95647 95602  > y <- factor(y)  > str(y)  Factor w/ 2 levels "95602","95647": 2 1 |
|  |
|  |

# Matrices

## Create a matrix

x <- matrix(c(1,2,3,4), 2,2)

> x

[,1] [,2]

[1,] 1 3

[2,] 2 4

# Misc

## Printing a variable to the console

for (i in id) {

**cat(i, "\n")**

}

# Reading Data

## Read a CSV file into a data.frame

> data1 <- read.csv("hw1\_data.csv")

> data1

Ozone Solar.R Wind Temp Month Day

1 41 190 7.4 67 5 1

2 36 118 8.0 72 5 2

3 12 149 12.6 74 5 3

# Statistics

## Quantiles

|  |
| --- |
| > some\_data <- c(0,1,2,3,4,5,6,7,8,9, 10)  > quantile(some\_data, c(0.3, 0.8))  30% 80%  3 8 |
|  |
| |  | | --- | |  | |

## Standard Deviation

standard\_dev <- sapply(test, sd)

# Stringsm

## Concatenate two strings

|  |
| --- |
| > paste("a","b", sep = "")  [1] "ab" |
|  |
| |  | | --- | | > |   > paste0("1","2")  [1] "12" Length of a string > nchar("The quick brown fox jumps over the lazy dog")  [1] 43 |

## Padding a string with zeros

> sprintf("%03d", 17)

[1] "017"

## Replace all the characters in a string

> test

[1] "replace\_the\_underscore\_characters"

> gsub("\_", " ", test)

[1] "replace the underscore characters"

## Replace characters in a string

|  |
| --- |
| > test <- "replace\_the\_underscore\_characters"  > test  [1] "replace\_the\_underscore\_characters"  > sub("\_", " ", test)  [1] "replace the\_underscore\_characters" |
|  |
| |  | | --- | |  | |

## Trim a string

> library(stringr)

> str\_trim(" Trim this! ")

[1] "Trim this!"

# Vectors

## Creating a vector

> a = 1:2

> a

[1] 1 2

## Iterating over a vector

for (i in id) {

writeLines(paste(i))

}

|  |
| --- |
| 1  2 |
|  |
| |  | | --- | |  | |

## Length of a vector

> bad <- ozone\_col[is.na(ozone\_col)]

> length(bad)

[1] 37

## Return valid elements of a vector

|  |
| --- |
| > a  [1] 1 2 NA 4  > a[complete.cases(a)]  [1] 1 2 4 |
|  |
| |  | | --- | |  | |