R Cheatsheet

Copyright 2014, Bruce Beauchamp

Version 1.0 13 Aug 2014

Contents

[R Cheatsheet 1](#_Toc395716763)

[Control statements 3](#_Toc395716764)

[Iterating through a vector 3](#_Toc395716765)

[data.frames 4](#_Toc395716766)

[Append a row to a data.frame 4](#_Toc395716767)

[Create a data.frame from scratch 4](#_Toc395716768)

[Get a cell of data from a data.frame 4](#_Toc395716769)

[Get the column names of a data.frame 4](#_Toc395716770)

[Get the number of rows in a data.frame 4](#_Toc395716771)

[Get a Row of Data from a data.frame 4](#_Toc395716772)

[Select Rows which meet criteria 5](#_Toc395716773)

[Select Columns from a data.frame 5](#_Toc395716774)

[Data Types 6](#_Toc395716775)

[Get the type of an object 6](#_Toc395716776)

[Environment 7](#_Toc395716777)

[Change the Working Directory 7](#_Toc395716778)

[Get the Working Directory 7](#_Toc395716779)

[List the Files in the Working Directory 7](#_Toc395716780)

[Load a Function into the Environment 7](#_Toc395716781)

[List the Functions Available in the Environment 7](#_Toc395716782)

[Matrices 7](#_Toc395716783)

[Create a matrix 7](#_Toc395716784)

[Misc 7](#_Toc395716785)

[Printing a variable to the console 7](#_Toc395716786)

[Reading Data 8](#_Toc395716787)

[Read a CSV file into a data.frame 8](#_Toc395716788)

[Strings 8](#_Toc395716789)

[Concatenate two strings 8](#_Toc395716790)

[Padding a string with zeros 8](#_Toc395716791)

[Vectors 9](#_Toc395716792)

[Creating a vector 9](#_Toc395716793)

[Iterating over a vector 9](#_Toc395716794)

[Length of a vector 9](#_Toc395716795)

[Return valid elements of a vector 9](#_Toc395716796)

# Control statements

## Iterating through a vector

id = 1:3

for (i in id) {

cat(i, "\n")

}

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

# 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 |
|  |
|  |

# 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 |
| 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 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" |
|  |
|  |

# 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”)

## List the Functions Available in the Environment

ls()

# 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

# Strings

## Concatenate two strings

|  |
| --- |
| > paste("a","b", sep = "")  [1] "ab" |
|  |
| |  | | --- | | > |   > paste0("1","2")  [1] "12" |

## Padding a string with zeros

> sprintf("%03d", 17)

[1] "017"

# 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 |
|  |
| |  | | --- | |  | |