

Part 2

Introduction to R

N. Levshina © 2015

Part of the course taught at the University of Mainz, Germany
26-28 May 2015

Outline

1. What is R?

2. R syntax

3. R objects

What is R?

- statistical computing environment (from *t*-test to generalized linear models, and more...)
 - core distribution “base”
 - add-on packages by professionals and amateurs (6685 packages on 25.05.2015)
- programming language
- tools for creation of publication-quality plots

Where to get R?

- Distribution and packages: CRAN
(Comprehensive R Archive Network)
<http://cran.r-project.org/>
- Information: <http://www.r-project.org/>

RStudio

- Highly recommended (easy to manage projects, packages, data, graphs, etc.)!
- Available from <http://www.rstudio.com/products/RStudio/>

Rling

- My package with data sets and some functions for this course
- Save the .tar.gz file to a local directory
- Install in R by typing in

```
> install.packages("Your/Path/Rling_1.0.tar.gz",  
repos = NULL, type = "source")
```

Outline

1. What is R?

2. R syntax

3. R objects

Input and output

```
> 2 + 2
```

```
[1] 4
```

```
> month.name
```

```
[1] "January" "February" "March" "April" "May"
```

```
[6] "June" "July" "August" "September" "October"
```

```
[11] "November" "December"
```

```
> 2+2;4+4
```

```
[1] 4
```

```
[1] 8
```


Creation of objects

```
> a <- 3
```

```
> a
```

```
[1] 3
```

```
> a + 5
```

```
[1] 8
```

Beware: = and ==

> a = 3 # creates an object a with the value 3, an alternative to "a <- 3"

> a == 3 # tests if a equals 3

[1] TRUE

> a == 10 # tests if a equals 10

[1] FALSE

R is case-sensitive!

```
> b <- 7
```

```
> a + b
```

```
[1] 10
```

```
> a + B
```

```
Error: object 'B' not found
```

Managing your objects

```
> ls() #returns a list of objects
```

```
[1] "a"    "b"
```

```
> rm(b) #removes an object
```

```
> ls()
```

```
[1] "a"
```

Saving your workspace

1. Click on the cross or type

```
> q()
```

Select the action (to save or not to save).

```
> getwd() #to find out where your workspace will be saved
```

```
[1] "C:/Users/Your/Directory"
```

```
> setwd("C:/Users/Your/Directory") #to change it, if you like
```

2. Next session: restart R or, if you have many different workspaces, click on the R from the directory; alternatively:

```
> load("yourDirectory/yourFile.RData")
```

Getting help

> `?cor` #to open a help file with information about function 'cor'

> `??correlation` #returns a list of functions that contain this expression

Errors

```
> x <- 1:10 # creates a numeric vector with numbers from 1  
to 10
```

```
> x
```

```
[1] 1 2 3 4 5 6 7 8 9 10
```

```
> meann(x) # we want to compute the mean value of x: a  
typo
```

```
Error: could not find function "meann"
```

```
> mean(x) # correct
```

```
[1] 5.5
```

Warning messages

```
> mytable <- rbind(c(1, 2), c(3, 4)) # create a 2-by-2 table
```

```
> mytable
```

```
  [,1] [,2]
```

```
[1,]  1  2
```

```
[2,]  3  4
```

```
> chisq.test(mytable)
```

Pearson's Chi-squared test with Yates' continuity correction

data: mytable

X-squared = 0, df = 1, p-value = 1

Warning message:

In chisq.test(mytable) : Chi-squared approximation may be incorrect

Outline

1. What is R?

2. R syntax

3. R objects

Main data types in R

- Numeric vectors
- Character vectors
- Factors
- Matrices
- Data frames

Numeric vectors

```
> vnum <- 1:5 # a vector of integers from 1 to 5
```

```
> vnum
```

```
[1] 1 2 3 4 5
```

If it's not a sequence:

```
> fibonacci10 <- c(1, 1, 2, 3, 5, 8, 13, 21, 34, 55)
```

```
> fibonacci10
```

```
[1] 1 1 2 3 5 8 13 21 34 55
```

Character vectors

```
> stein <- c("a", "rose", "is", "a", "rose", "is", "a", "rose")
```

```
> stein
```

```
[1] "a" "rose" "is" "a" "rose" "is" "a" "rose"
```

Factors

```
> stein.fac <- factor(stein)
```

```
> stein.fac
```

```
[1] a  rose is a  rose is a  rose
```

```
Levels: a is rose
```

Matrices

```
> m <- cbind(1:5, 10:6)
```

```
> m
```

```
  [,1] [,2]  
[1,]  1 10  
[2,]  2  9  
[3,]  3  8  
[4,]  4  7  
[5,]  5  6
```

Data frames

```
> sex <- c("f", "m", "m", "f")
```

```
> sex
```

```
[1] "f" "m" "m" "f"
```

```
> rt <- c(455, 773, 512, 667)
```

```
> rt
```

```
[1] 455 773 512 667
```

```
> df <- data.frame(sex, rt)
```

```
> df
```

	sex	rt
1	f	455
2	m	773
3	m	512
4	f	667

Exercise

Create a character vector with the names of your fellow students. Create a vector with their heights (in cm). Combine the vectors in one data frame.

Importing your data to R

1. Create a similar table in Excel (or OpenOffice Calc). Don't forget to create a header. In case of missing values, put NA. No empty cells!
2. Save the file as a tab delimited text file (.txt).
3. Read the file in R:

```
> mydata <- read.table("C:/Your/Directory/mydata.txt", header = TRUE)
```

NB: use either forward slashes or double backward slashes, as in C:\\Your\\Directory\\mydata.txt !

Exercise

Create the following table in Excel (or OpenOffice Calc) and import it in R as a data frame under the name *Linguists*.

Last name	First name	Framework	Born	Died
de Saussure	Ferdinand	Structuralism	1857	1913
Chomsky	Noam	Generative Linguistics	1928	NA
Lakoff	George	Cognitive Linguistics	1941	NA

Exporting your data from R

```
> write.table(mydata, file =  
"C:/Your/Directory/Exported.txt", quote = FALSE, sep = "\t",  
row.names = FALSE)
```

A practical tip

- Instead of typing in the path, you can type `file = file.choose()` and select the directory interactively

