

# Introduction to RStudio

Ulrich Halekoh

Epidemiology and Biostatistics, SDU

May 4, 2018

# R

R is a language that

- ▶ started by Ross Ihaka and Robert Gentleman in 1991 as an open source alternative to S
- ▶ emphasizes interactive data analysis: computational and graphical
- ▶ extensible via functions (and packages)
- ▶ lingua franca of statistical computation: very often new statistical algorithms are published in R
- ▶ important computational projects are developed in R (e.g. Bioconductor for high-throughput genomic data)
- ▶ R is free (but you have to invest time to learn it!)

# CRAN

Access to many features of the R-project is via the The Comprehensive R Archive Network (CRAN)

Here you find information about

- ▶ Download of new versions
- ▶ (add-on) Packages
- ▶ The R Journal
- ▶ Manuals
- ▶ FAQ

# RStudio

In order to use R it is an advantage to have a versatile development environment:


- ▶ Console (command) window, such you can easily work with your data interactively
- ▶ Source (Editor) for your programs (scripts), which you can be reused after a long days work
- ▶ neat 'project' organizer for
  - ▶ plots
  - ▶ data files locations

RStudio is such an *integrated development environment* (IDE).

## Setting up a project directory - 1

- ▶ In any statistical work it is advantageous to have ones script- and data files well organized.
- ▶ RStudio helps you by setting up a project directory.
- ▶ We assume that you have already a folder on your computer which contains the .R and data files, supplied for this course.
- ▶ For example we assume that this folder is called (on a windows machine)  
C:\users\Rtwin2018

## Setting up a project directory - 2

1. Start RStudio by clicking on the icon , which you may find on your desktop, or you find it via `Start → RStudio`
2. In RStudio follow now the path `File → New Project → Existing Directory`
- 3.
4. Use the `Browse window` to browse to the directory `C:\users\Rtwin2018`.
5. Click at the bottom `Create project`
6. A Console-window should pop up. Type

```
getwd()
```

and press ENTER. Your project directory should appear.

## Command window (Console) and shortcuts

On the left side you see the **command window**, which is used for interactive work.

Useful shortcuts in R

- ▶ Enter simple R code that defines the variable called `z` and assigns it the value 5

```
z <- 5
```

Equivalently you could have used

```
z = 5
```

- ▶ press now  
ENTER  
The command is executed
- ▶ To print the value of `z` either type

```
z
```

followed by ENTER. Alternatively you can use explicitly the `print()` function of R

```
print(z)
```

## An R-script and the source window

Any serious statistical analysis comprises several lines of code, which should be saved for documentation and later use.

Open the RStudio built-in-editor via the path

File → New → R Script

A window will open onto an empty script file.

We enter a simple program:

```
library(mets)      # loading an add-on R package
data(twinbmi)     # make a data-set from the package available to the session
summary(twinbmi)  # an overview summary
```

Execute the little program by

- ▶ Mark all the lines **Ctrl A** (Mac: **Cmd A**)
- ▶ Press **Ctrl ENTER** (Mac: **Cmd ENTER**)



## Execute a single line from a script

Just put the cursor in the line and press **ENTER**

If you forget the closing parenthesis )

```
summary(twinbmi
```

R shows a new continuation line with a +

```
print(twinbmi
```

```
+
```

reminding you to close the parenthesis.

A command in R may run across several lines

```
print (  
twinbmi  
)
```

To execute it you must mark the three lines and hit **Ctrl ENTER**  
( **Cmd ENTER** ).

## Save your script

To save your script follow the path **File → Save**

A window opens to your project directory.

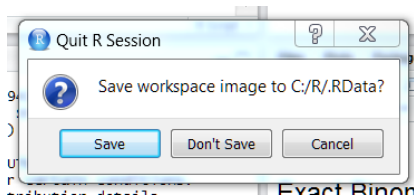
- ▶ In **File name** enter some name: e.g. analysis1
- ▶ Hit **Save** and the file is saved as analysis1.R

You find your file by going to the lower right window and pressing **Files**.

During longer work it is advisable to save your file frequently by hitting **Ctrl S**.

## Closing a session

To close a session click on the right upper  $\times$ .  
The pop-up window



asks you to save all your created objects. In most cases to can safely answer **Don't Save**.

## Resuming a session

Either

- ▶ Click on the RStudio icon or
- ▶ Go to your project directory and double click on your script file

## Add-on packages

One reason for the usefulness of R is the numerous add-on packages that are developed and made available via The Comprehensive R Archive Network (CRAN)

<http://cran.r-project.org/web/packages/>

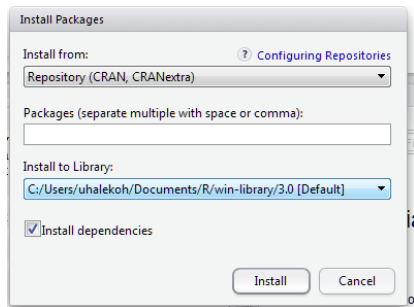
You may browse the list of available packages.

For overview about general subjects you may check the Task views

## Installing and loading a package

Assume you want to install the package `lava` for fitting SEMs.

In RStudio: **Tools** → **Install Packages**



Enter `lava` in **Packages(separate ..)** and click **Install**.

To make the package available to your R-session execute

```
library(lava)
```

# Installation vs. loading a package

Note the difference

- ▶ Installation: Must be done only once (and if you update a package)
- ▶ Loading: Must be done in each new R-session

Remember that you should regularly update your packages (RStudio: **Tools → Check for Package Updates** ).