

Basics in R for people who are afraid of computers

Prof. Dr. Rik Vosters

Vrije Universiteit Brussel

www.rikvosters.be

Rik.Vosters@vub.ac.be

Who, what and for whom?

- Rik Vosters
- First, basic-level (but thorough!) introduction
- Working with numbers and text in R
- Descriptive statistics and graphs
- Hands-on workshop

This course

- A taste of the possibilities:
 - R as a text and data manipulation tool
 - R as a tool for descriptive statistics and graphics
- Data analysis ‘for the rest of us’
- Not an expert – but a satisfied customer
- Fairly ‘manual’ approach (~ basic), in addition to using some popular existing packages
- Stop me if you have any questions...
And give a yell if you are completely lost!
- Going round during exercises and for problems?

- **1. GETTING STARTED**
- **2. BASIC VECTORS**
 - 2.1 Numeric vectors
 - 2.2 Character vectors
- **3. DATAFRAMES, SUBSETTING AND DATA IMPORT**
 - 3.1 Data frames
 - 3.2 Subsetting
 - 3.3 Importing data
 - 3.4 An alternative approach: tidyverse/dplyr
 - 3.5 Other types of data structures
- **4. DATA CLEANING AND MANIPULATION**
 - 4.1 Data classes
 - 4.2 Basic manipulations
 - 4.3 Long and wide data
 - 4.4 Merging datasets and working with metadata
 - 4.5 Loading multiple files (loops and lapply)

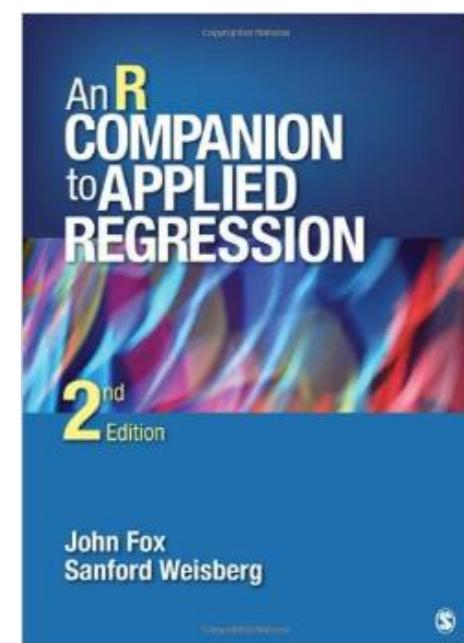
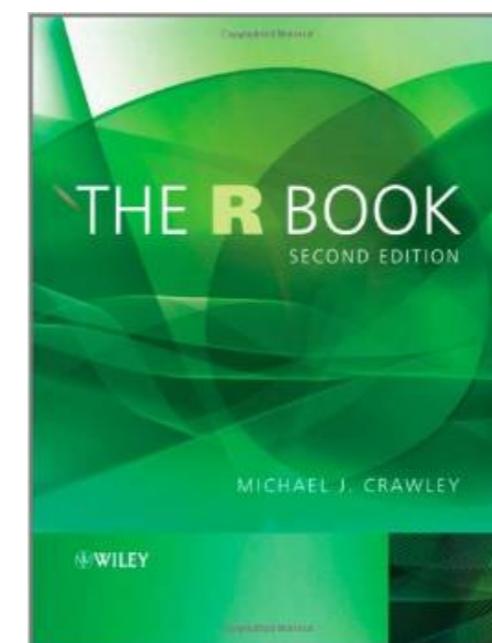
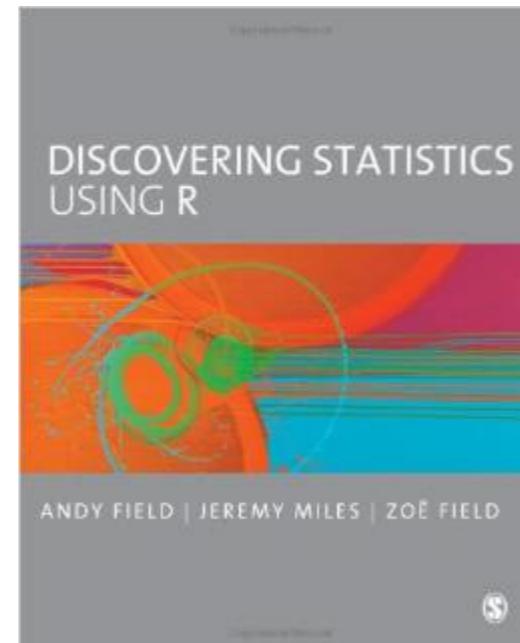
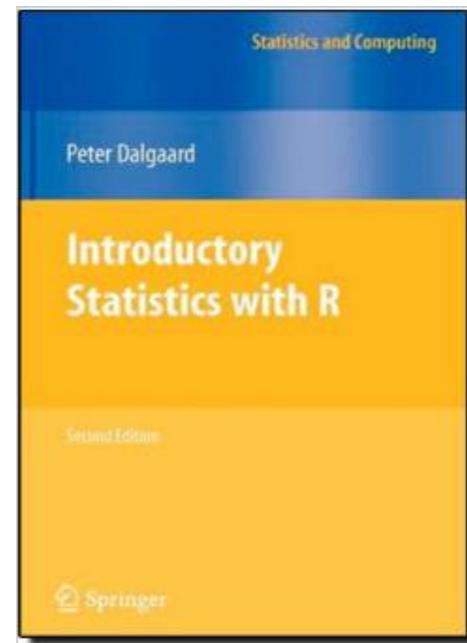
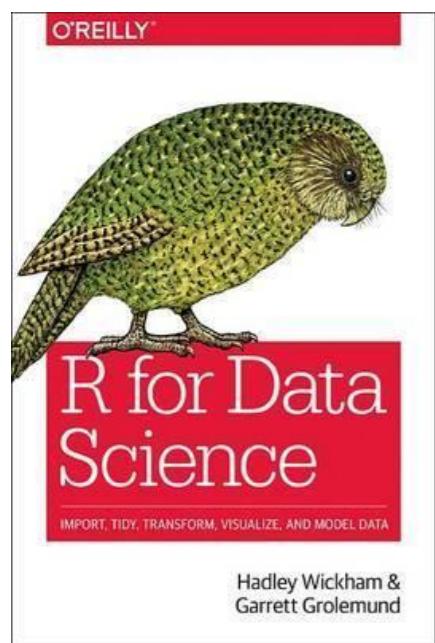
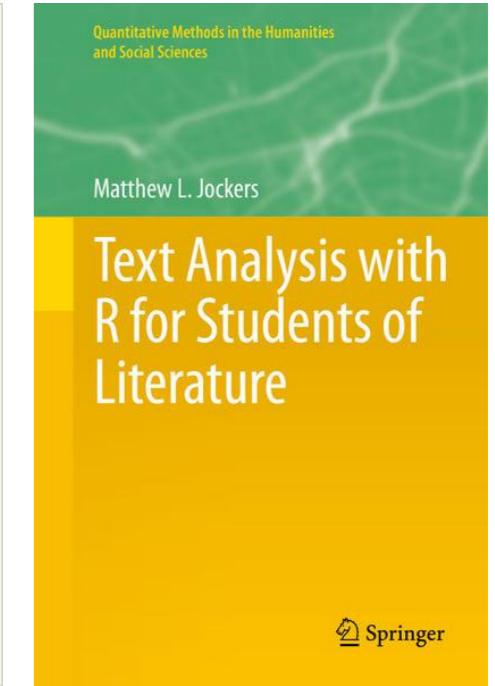
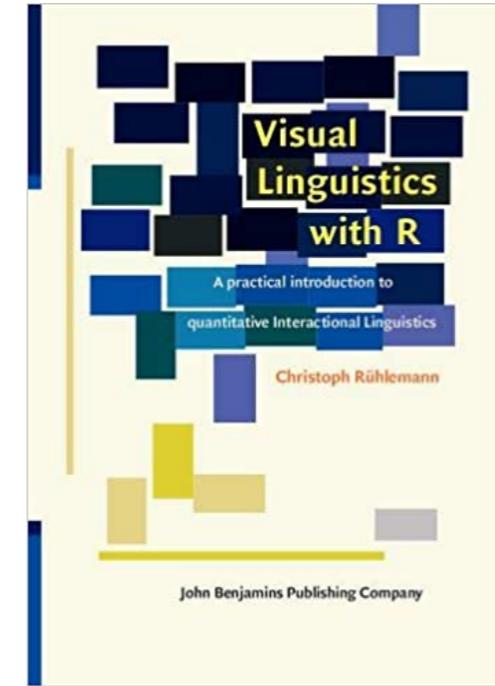
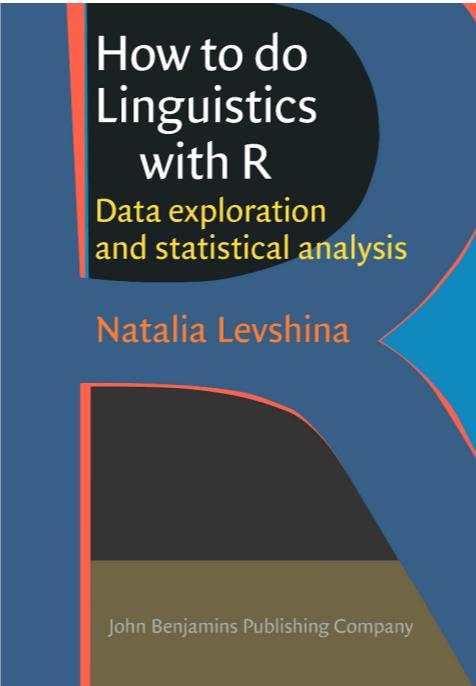
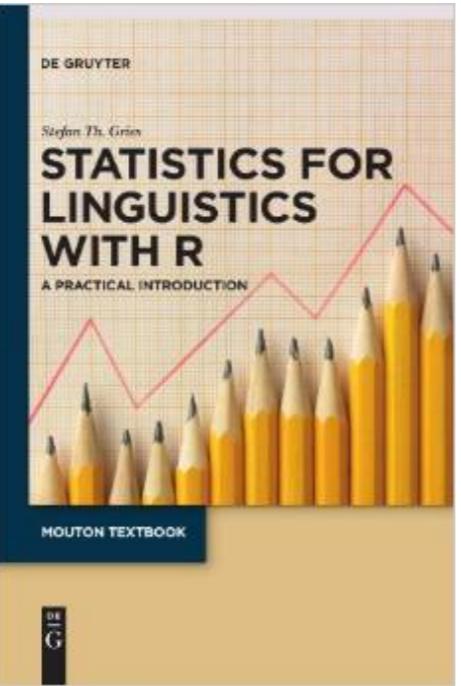
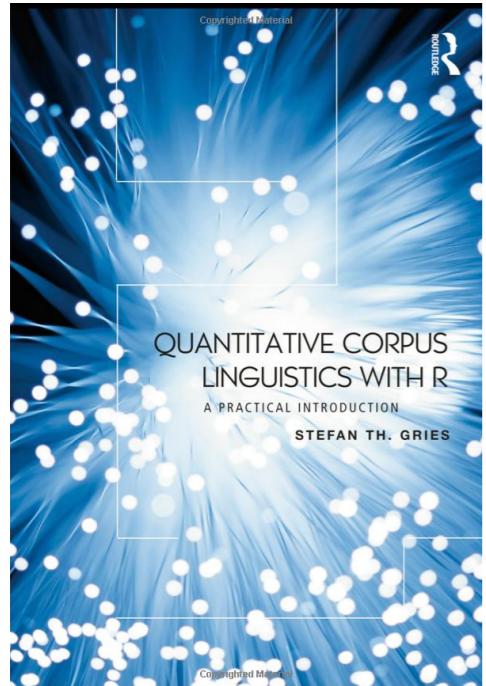
- **5. ANALYZING TEXTUAL DATA**
 - 5.1 Loading text files
 - 5.2 Basic text manipulation
 - 5.3 Powersearching and regular expressions
 - 5.4 Lemmatizing and POS tagging
 - 5.5 Frequency lists
 - 5.6 Keyness and word clouds
 - 5.7 Concordances and collocations
- **6. DATA EXPLORATION AND VISUALIZATION**
 - 6.1 Preparation
 - 6.2 Numerically summarizing - base package
 - 6.3 Visualization - base package
 - 6.4 Numerically summarizing - tidyverse/dplyr
 - 6.5 Visualization - tidyverse/ggplot
- **7. BASIC STATISTICS**
 - 7.1 Distributions
 - 7.2 Frequencies
 - 7.3 Means
 - 7.4 Correlations

+ *GUEST SECTION: AUTHORSHIP ATTRIBUTION*
by Andrea Penso (FWO)

Tentative planning

	Wednesday	Thursday	Friday
Morning	Getting started Basic vectors	Data cleaning and manipulation	Data exploration and visualization Basis statistics
Afternoon	Dataframes, subsetting and data import	Analyzing textual data	Basis statistics + guest session: authorship attribution (A. Penso)

A few useful books



Quick introductions

- Who are you? Background? Expertise?
- What are your data?
- Your relationship with computers...?
- Experience with R / programming?

Getting started

- Installing R — <http://www.r-project.org>



- RStudio — <http://www.rstudio.com>



RStudio

Project: (None)

_script.R* workshop.R* Untitled1*

Source on Save

Run Source

To start off

Basic functions in R: addition, subtraction, etc.

3+6
3*6
3 + 6
3+6 # anything behind the hash is a comment and is not processed
3+6 # +100

Script

11:1 # To start off

Console ~/Dropbox/@ Documenten/Lezingen (afgewerkt)/2013.05 ijsland – jgl/Corpusonderzoek/

>
> 3+6
[1] 9
> 3*6
[1] 18
> 3 + 6
[1] 9
> 3+6 # anything behind the hash is a comment and is not processed
[1] 9
> 3+6 # +100
[1] 9
>
>
>
>
>
>
>
>
>
>
>

Environment History Help

R: Add Text to a Plot Find in Topic

text {graphics} R Documentation

Add Text to a Plot

Description

text draws the strings given in the vector labels at the coordinates given by x and y. y may be missing since `xy.coords(x, y)` is used for construction of the coordinates.

Usage

```
text(x, ...)

## Default S3 method:
text(x, y = NULL, labels = seq_along(x), adj =
    pos = NULL, offset = 0.5, v =
    cex = 1, col = NULL, font =
```

Arguments

x, y numeric vectors of coordinates where

Files Plots Packages Viewer

New Folder Delete Rename

Home

	Name	Size	Modified
□	.Rhistory	2.4 KB	Jul 24,
□	Applications		
□	Applications (Parallels)		
□	Desktop		
□	Documents		
□	Downloads		
□	Dropbox		
□	Library		
□	Movies		
□	Music		
□	Pictures		
□	pkcs11.log	0 B	Jul 24, 2014, 10:00 AM
□	Public		
□	tmp		

Help Files Plots Packages etc.

The screenshot shows the RStudio interface. In the top-left, there's a code editor with several tabs open, including `_script.R*`, `workshop.R*`, and `Untitled1*`. The main pane displays the following R code:

```
1 ###### To start off #####
2
3 # Basic functions in R: addition, subtraction, etc.
4
5 3+6
6 3*6
7 3 + 6
8 3+6 # anything behind the hash is a comment and is not processed
9 3+6 # +100
10
11
```

In the top-right, the help panel is open for the `text` function, specifically the `text {graphics}` version. It includes sections for Description, Usage, and Arguments.

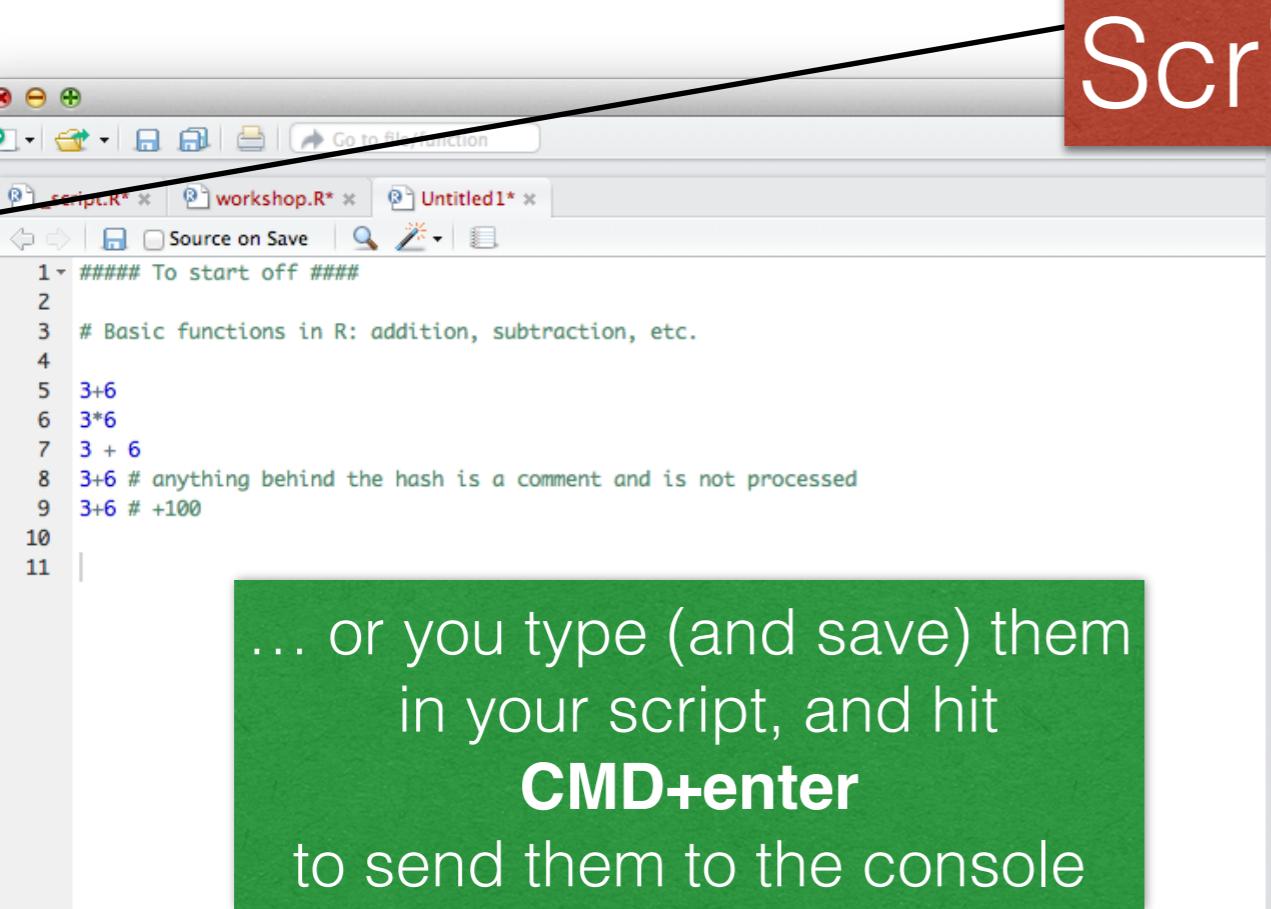
Console

The screenshot shows the R console window. The title bar says "Console ~/Dropbox/@ Documenten/Lezingen (afgewerkt)/2013.05 ijsland - jgl/Corpusonderzoek/". The console history shows the following commands:

```
>
>
> 3+6
[1] 9
> 3*6
[1] 18
> 3 + 6
[1] 9
> 3+6 # anything behind the hash
[1] 9
> 3+6 # +100
[1] 9
>
>
>
>
>
>
>
>
>
>
>
>
>
```

A green callout box with white text is overlaid on the console area, containing the text: "You either type directly into the console...".

Script



```
1 ###### To start off #####
2
3 # Basic functions in R: addition, subtraction, etc.
4
5 3+6
6 3*6
7 3 + 6
8 3+6 # anything behind the hash is a comment and is not processed
9 3+6 # +100
10
11
```

... or you type (and save) them in your script, and hit **CMD+enter** to send them to the console

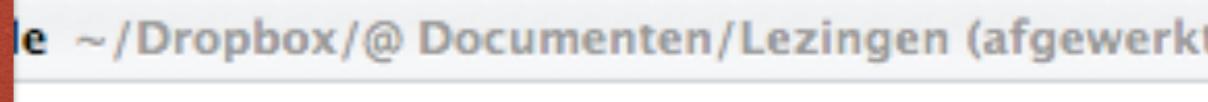


```
1 ###### To start off #####
2
3 # Basic functions in R: addition, subtraction, etc.
4
5 3+6
6 3*6
7 3 + 6
8 3+6 # anything behind the hash is a comment and is not processed
9 3+6 # +100
10
```

Console

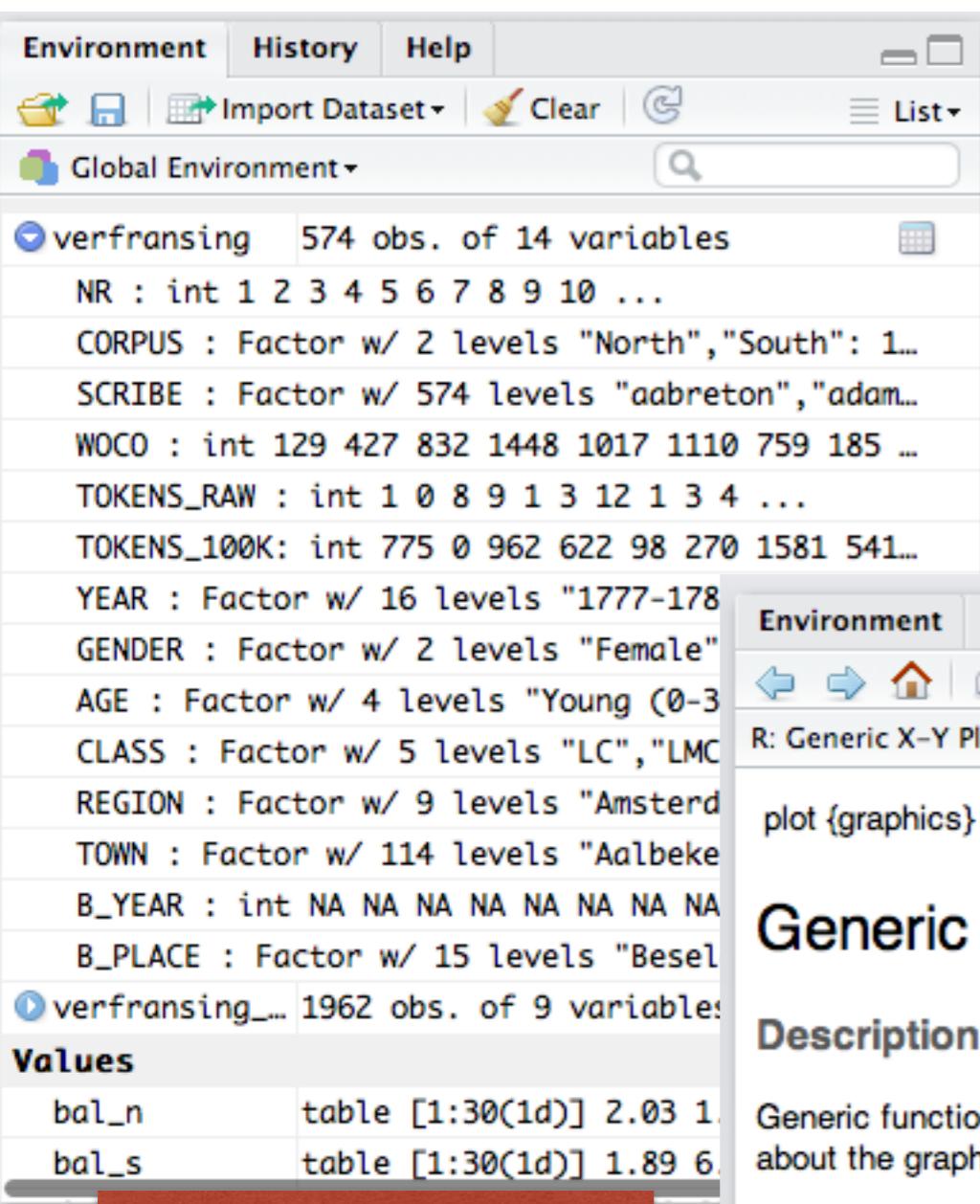


```
11:1 # To start off
>
> 3+6
[1] 9
> 3*6
[1] 18
> 3 + 6
[1] 9
> 3+6 # anything behind the hash is a comment and is not processed
[1] 9
> 3+6 # +100
[1] 9
>
>
>
>
>
>
>
>
>
>
>
>
```



```
>
> 3+6
[1] 9
> 3*6
[1] 18
> 3 + 6
[1] 9
> 3+6 # anything behind the hash is a comment and is not processed
[1] 9
> 3+6 # +100
[1] 9
>
```

Other useful windows



environment

The RStudio interface showing the Help tab. It displays the help documentation for the `plot` function, which is a generic X-Y Plotting function. The documentation includes:

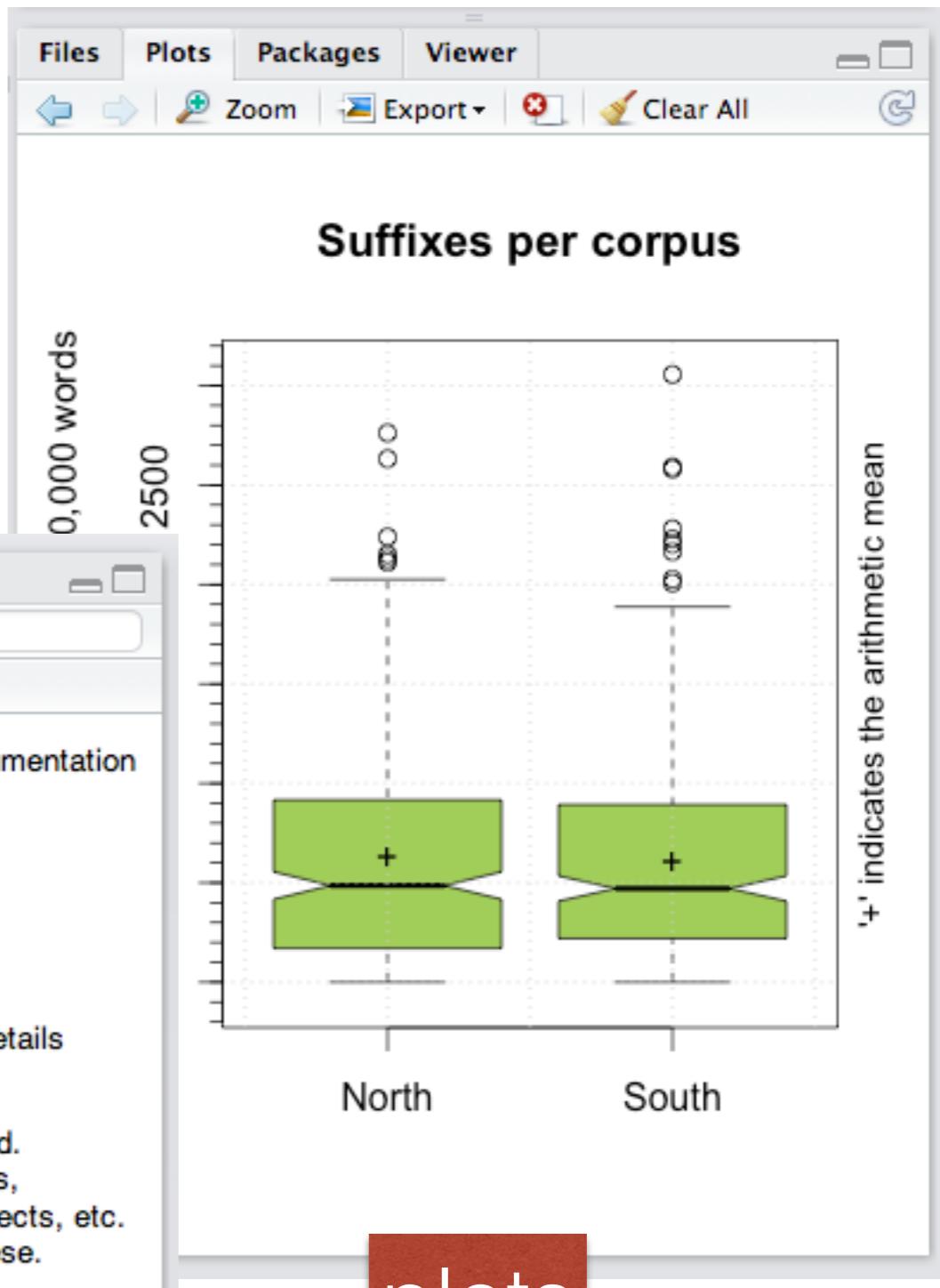
- plot {graphics}**
- R Documentation**
- ## Generic X-Y Plotting
- ### Description

Generic function for plotting of R objects. For more details about the graphical parameter arguments, see [par](#).
- For simple scatter plots, [plot.default](#) will be used. However, there are `plot` methods for many R objects, including [functions](#), [data.frames](#), [density](#) objects, etc. Use `methods(plot)` and the documentation for these.
- ### Usage

```
plot(x, y, ...)
```
- ### Arguments

 - x**: the coordinates of points in the plot. Alternatively, a single plotting structure, function or any R object with a `plot` method can be provided.

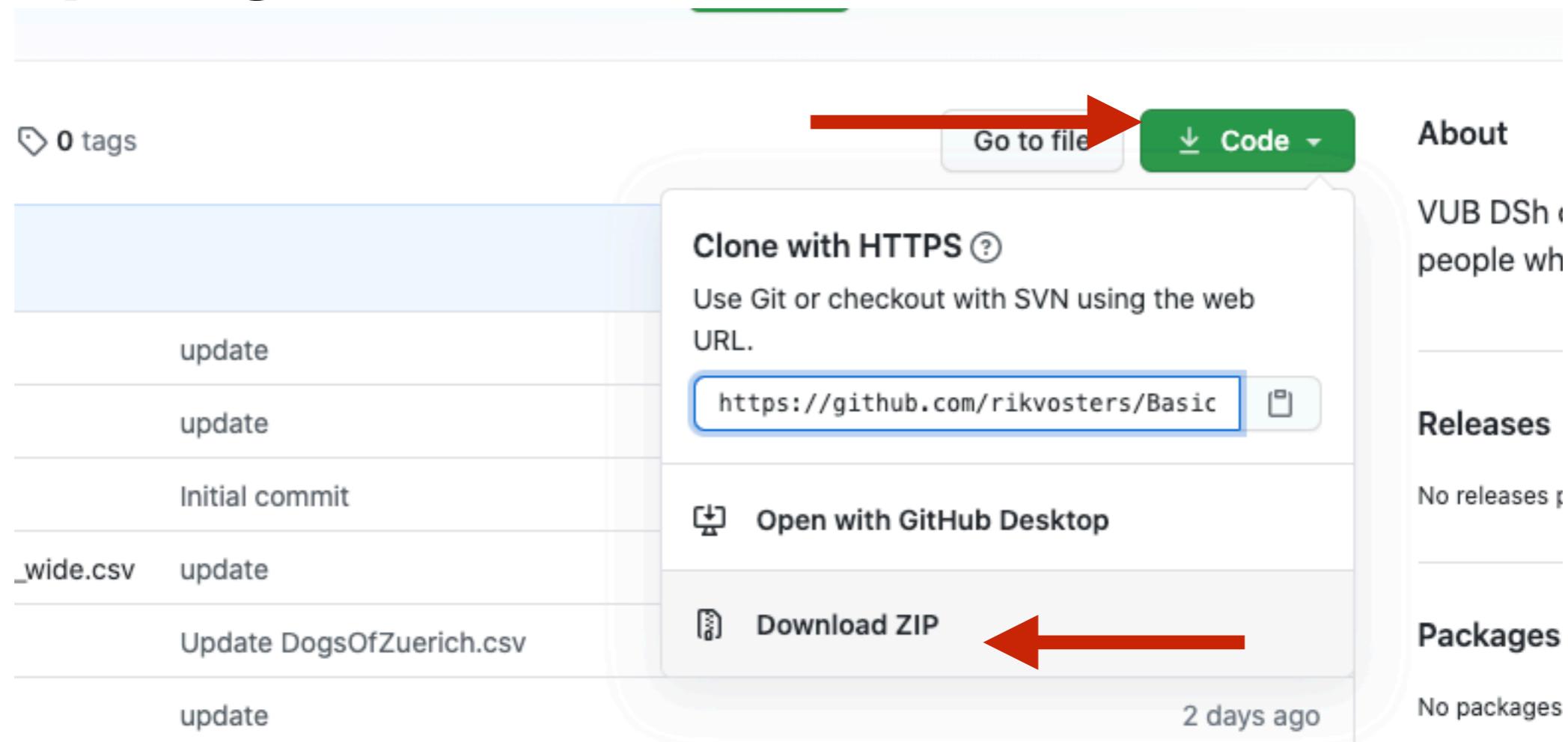
help



plots

- Open a browser and go to:

https://github.com/rikvosters/Basics-in-R



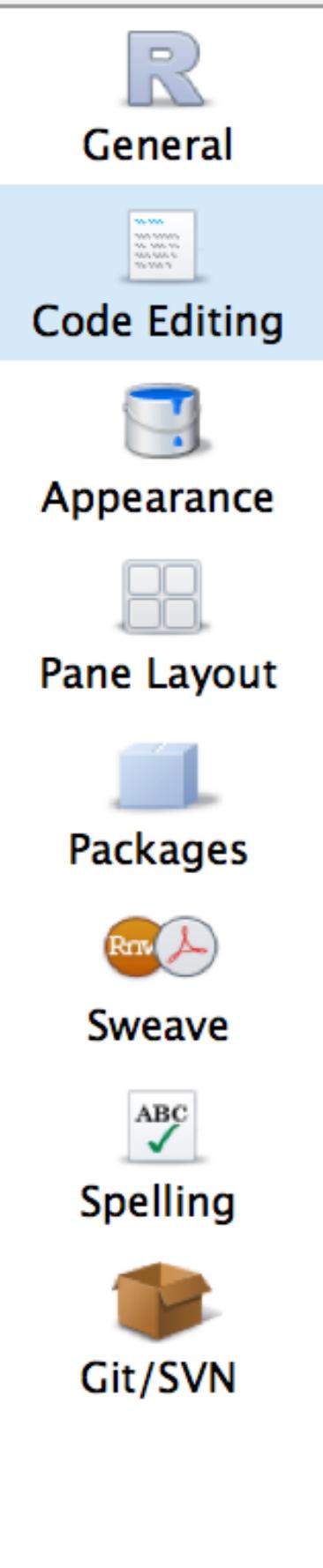
- Give it a few seconds to download, and then click to *open* the downloaded file.
- It will unpack the zip archive. Once it's done, you'll *see a folder* with many files in it.

Getting started

- Make a **new folder** on the desktop:
Go to the *Desktop*, *right click* there, and select *New... + folder* (new folder). *Type ‘workshop’* as the name for this new folder.
- **Select** all unzipped files:
Click in the window with all your files again, and press *CTRL + A* on your keyboard.
- **Move** unzipped files to new folder:
You’ll *see* all of the files are *highlighted*. Now *drag and drop* all of them *to the new folder ‘Workshop’* on your desktop.

Getting started

- **Open script in RStudio**
 - locate the file
workshop - script.R
 - *double-click* it to open
 - for the first time, we need to tell Windows to open these sorts of files with Rstudio
 - next, click on *RStudio* and hit **OK** again.



- Highlight selected word
- Highlight selected line
- Show line numbers
- Insert spaces for tab
 - Tab width
- Show margin
 - Margin column
- Show whitespace characters
- Show indent guides
- Blinking cursor
- Insert matching parens/quotes
- Auto-indent code after paste
- Vertically align arguments in auto-indent
- Soft-wrap R source files
- Ensure that source files end with newline
- Strip trailing horizontal whitespace when saving
- Focus console after executing from source
- Show syntax highlighting in console input
- Enable vim editing mode

Options:
“softwrap R
Source files”
(Tools >
Global Options)

OK

Cancel

Apply

setwd()

- **CTRL + SHIFT + H**

