

# Develop Packages in R

## Part 3: Include Package Dependencies

**April 2019**

Isabella Gollini

[Isabella.Gollini@ucd.ie](mailto:Isabella.Gollini@ucd.ie)

[@IsabellaGollini](#)

Bruna Wundervald [@bwundervald](#)

Chiara Cotroneo [@selenocysteina](#)

Jo Nieć [@joannaniec](#)

[@RLadiesDublin](#)



<https://rladies.org/>

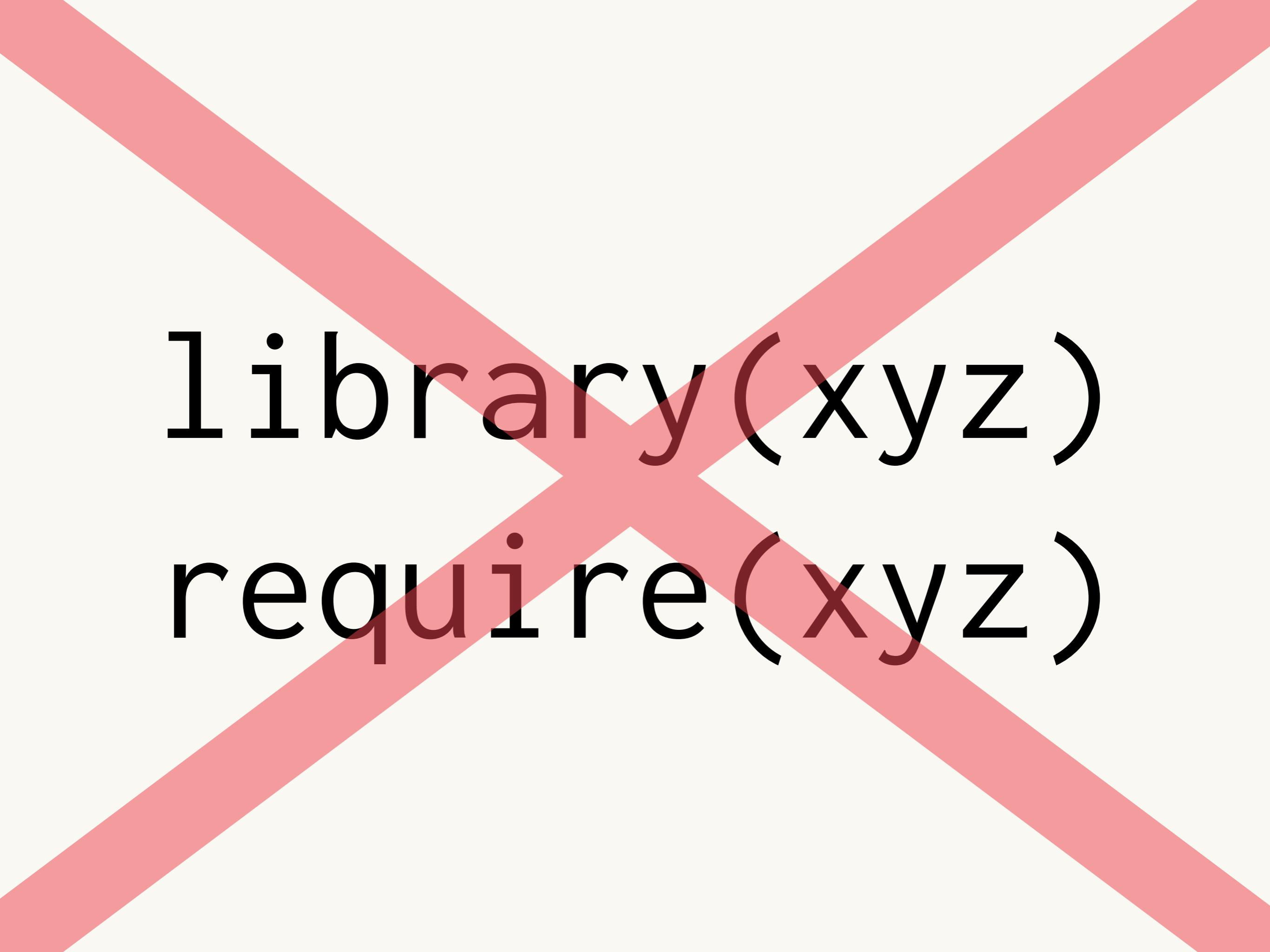
<https://forwards.github.io>

<https://github.com/forwards/workshops/>



1. Dependencies
2. Namespace: imports
3. Namespace: exports
4. R CMD check

# Dependencies



```
library(xyz)  
require(xyz)
```

I need you!

```
R (>= 3.0.2) # optional version spec
```

Imports:

```
stringr (>= 1.0.0),
```

```
lubridate
```

Suggests:

I like having you around

There are three types of dependency

**Imports** = required. Installed automatically.

**Suggests** = optional: development only; used in vignette or example. **Not** installed automatically.

**Depends** = basically deprecated for packages.(Correct uses exist, but beyond the scope of this class)

# Use :: to access functions in imported packages

```
# In DESCRIPTION
```

```
Imports: foo
```

```
# In bar.R
```

```
new_function <- function(x, y, z) {  
  foo::bar(x, y) + z  
}
```

# Should check if suggested package available

```
# In DESCRIPTION
```

```
Suggests: foo
```

```
# In bar.R
```

```
new_function <- function(x, y, z) {  
  if (!requireNamespace("foo", quietly = TRUE)) {  
    stop("Need foo! Use install.packages('foo') .")  
  }  
  foo::bar(x, y) + z  
}
```

```
# use_package() will modify the DESCRIPTION  
# and remind you how to use the function.  
usethis::use_package("ggplot2")  
usethis::use_package("ggplot2", "suggests")
```

# Namespace: imports

# You might get tired of using :: all the time

```
# Or you might want to use an infix function
`%>%` <- magrittr::`%>%`  
  
col_summary <- function(df, fun) {  
  stopifnot(is.data.frame(df))  
  
  df %>%  
    purrr::keep(is.numeric) %>%  
    purrr::modify(fun)  
}
```

# You can import functions into the package

```
#' @importFrom purrr keep modify
#' @importFrom magrittr %>%
col_summary <- function(df, fun) {
  stopifnot(is.data.frame(df))
  df %>%
    keep(is.numeric) %>%
    modify(fun)
}
```

# Alternatively, create R/imports.R

```
# Imports belong to the package, not to
# individual functions, so you might want
# to recognise this by storing in a central
# location

#' @importFrom purrr keep map
#' @importFrom magrittr %>%
NULL
```

# Importing everything from a package seems easy

```
#' @import purrr
col_summary <- function(df, fun)  {
  stopifnot(is.data.frame(df))
  df %>%
    keep(is.numeric) %>%
    map_dfc(fun)
}
```

# But is dangerous...

```
#' @import foo
#' @import bar
fun <- function(x) {
  fun1(x) + fun2(x)
}

# Works today
# But next year, bar package adds fun1 function
```

## Description

## NAMESPACE

Makes **package** available

Makes **function** available

Mandatory

Optional  
(can use :: instead)

`use_package()`

`#' @importFrom`

# Namespace: exports

# A namespace splits functions into two classes

Internal	External
Only for use within package	For use by others
Documentation optional	Must be documented
Easily changed	Changing will break other people's code

# The default NAMESPACE exports everything

```
# Generated by roxygen2: fake comment so
# roxygen2 overwrites silently.
exportPattern("^[^\\.]")
```

# Better to export function explicitly

```
#' @export  
fun1 <- function(...) { }
```

```
#' @export  
fun2 <- function(...) { }
```

Most important if you're  
planning on sharing with others

# Export functions that people should use

```
# Don't export internal helpers

# Defaults for NULL values
`%||%` <- function(a, b) if (is.null(a)) b else a

# Remove NULLs from a list
compact <- function(x) {
  x[!vapply(x, is.null, logical(1))]
}
```

R CMD check

# Automated checking

Runs automated checks for common problems in R packages.

Useful for local packages, even with some false positives.

If you want to submit to CRAN, you **must** pass R CMD check cleanly.

A man with short brown hair and a mustache is shown in profile, running on a black treadmill. He is wearing a blue t-shirt and dark shorts. His right arm is bent, and his hand is near his chin, suggesting he is thinking or concentrating. In front of him is a black laptop computer, which is tilted slightly to the left. The background is a solid red color.

To avoid frustration run  
early and run often

```
==> R CMD build rv2
```

```
* checking for file 'rv2/DESCRIPTION' ... OK
* preparing 'rv2':
* checking DESCRIPTION meta-information ... OK
* installing the package to build vignettes
* creating vignettes ... OK
* checking for LF line-endings in source and make files
* checking for empty or unneeded directories
* building 'rv2_0.1.tar.gz'
```

```
==> R CMD check rv2_0.1.tar.gz
```

```
* using log directory '/Users/hadley/Documents/courses/13-devtools/rv2.Rcheck'
* using R version 3.0.2 (2013-09-25)
* using platform: x86_64-apple-darwin10.8.0 (64-bit)
* using session charset: UTF-8
* checking for file 'rv2/DESCRIPTION' ... OK
* this is package 'rv2' version '0.1'
* checking package namespace information ... OK
* checking package dependencies ... OK
* checking if this is a source package ... OK
* checking if there is a namespace ... OK
* checking for executable files ... OK
* checking for hidden files and directories ... OK
* checking for portable file names ... OK
```

```
* checking for sufficient/correct file permissions ... OK
* checking whether package 'rv2' can be installed ... OK* checking installed package size ... OK
* checking package directory ... OK
* checking DESCRIPTION meta-information ... OK
* checking top-level files ... OK
* checking for left-over files ... OK
* checking index information ... OK
* checking package subdirectories ... OK
* checking R files for non-ASCII characters ... OK
* checking R files for syntax errors ... OK
* checking whether the package can be loaded ... OK
* checking whether the package can be loaded with stated dependencies ... OK
* checking whether the package can be unloaded cleanly ... OK
* checking whether the namespace can be loaded with stated dependencies ... OK
* checking whether the namespace can be unloaded cleanly ... OK
* checking loading without being on the library search path ... OK
* checking dependencies in R code ... OK
* checking S3 generic/method consistency ... OK
* checking replacement functions ... OK
* checking foreign function calls ... OK
* checking R code for possible problems ... OK
* checking Rd files ... OK
* checking Rd metadata ... OK
* checking Rd cross-references ... OK
* checking for missing documentation entries ... OK
* checking for code/documentation mismatches ... OK
```

```
* checking Rd \usage sections ... OK
* checking Rd contents ... OK
* checking for unstated dependencies in examples ... OK
* checking installed files from 'inst/doc' ... OK
* checking files in 'vignettes' ... OK
* checking examples ... OK
* checking for unstated dependencies in tests ... OK
* checking tests ...
  Running 'testthat.R' OK
* checking for unstated dependencies in vignettes ...
OK
* checking package vignettes in 'inst/doc' ... OK
* checking running R code from vignettes ...
  'clt.Rmd' ... OK
OK
* checking re-building of vignette outputs ... OK
* checking PDF version of manual ... OK
```

R CMD check succeeded

# Types of problem

	Local	CRAN
ERROR	✓	✓
WARNING		✓
NOTE		✓

**ERROR**

Must fix!

**WARNING**

Fix if submitting to CRAN

**NOTE**

Fix if submitting to CRAN

It is possible to submit with a NOTE, but it's best avoided

```
# Cmd/Ctrl + Shift + E  
devtools::check()
```

```
# If you don't understand an error,  
# google it!
```

# Share it!

## Build the source package to share it!

```
devtools::build()
```

Now it can be installed by using:

```
install.packages("hadcol_0.0.0.9000.tar.gz",
                 repos = NULL, type = "source")
library(hadcol)
?add_col
```

Notice that 0.0.0.9000 is the version you specified in the description file.

# Learning more

# Further topics

- Including data
- Licenses
- CRAN submission
- Good practice
  - Version control (git and GitHub)
  - Code coverage (code covered by unit tests)
  - Continuous integration (automated checking)
- Advanced programming
  - Interfacing with C++
  - Object-oriented programming

...

The screenshot shows the R Studio Community forum interface. At the top, there's a navigation bar with icons for back, forward, search, and user profile. The main header says "R Studio Community". Below the header, there's a search bar and a user icon with a notification count of 4.

The main content area lists several discussion categories:

- Package development** (devtools icon): 2 / week, 2 unread, 1 new.
- RStudio Cloud** (cloud icon): 2 / week.
- uncategorized** (question mark icon): 9 / week, 5 unread, 4 new.
- meta** (m icon): 37, 3 unread.
- rstudio::conf 2018** (dots icon): 9.
- Lounge** (lounge chair icon): 1 / week, 1 unread.
- Staff** (person icon): 11.

community.rstudio.com

The screenshot shows the R-project.org mailing list interface for the **R-package-devel** list. At the top, there's a navigation bar with icons for back, forward, search, and user profile. The main header says "www.r-project.org/mail.html".

The left sidebar contains links to other mailing lists and resources:

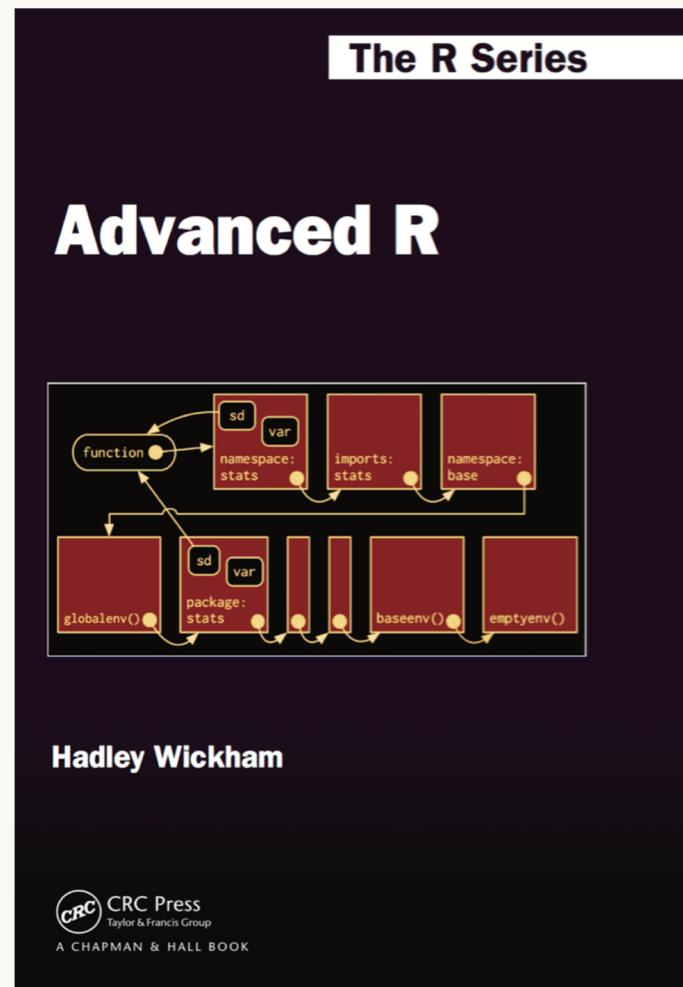
- Manuals**: [FAQs](#), [The R Journal](#), [Books](#), [Certification](#), [Other](#).
- Links**: [Bioconductor](#), [Related Projects](#), [GSoC](#).

The main content area for the **R-package-devel** list includes:

- R-package-devel**: A brief description stating it's for help about package development in R, a forum for learning about the package development process, and a community of R package developers.
- A note about overlap with the **R-devel** list.
- R-devel**: A detailed description of the list's purpose, mentioning it's for questions and discussion about code development in R, and noting that questions likely to prompt discussion unintelligible to non-programmers or topics that are too technical for R-help's audience should go to R-devel.
- A note about subscribing and digest options.

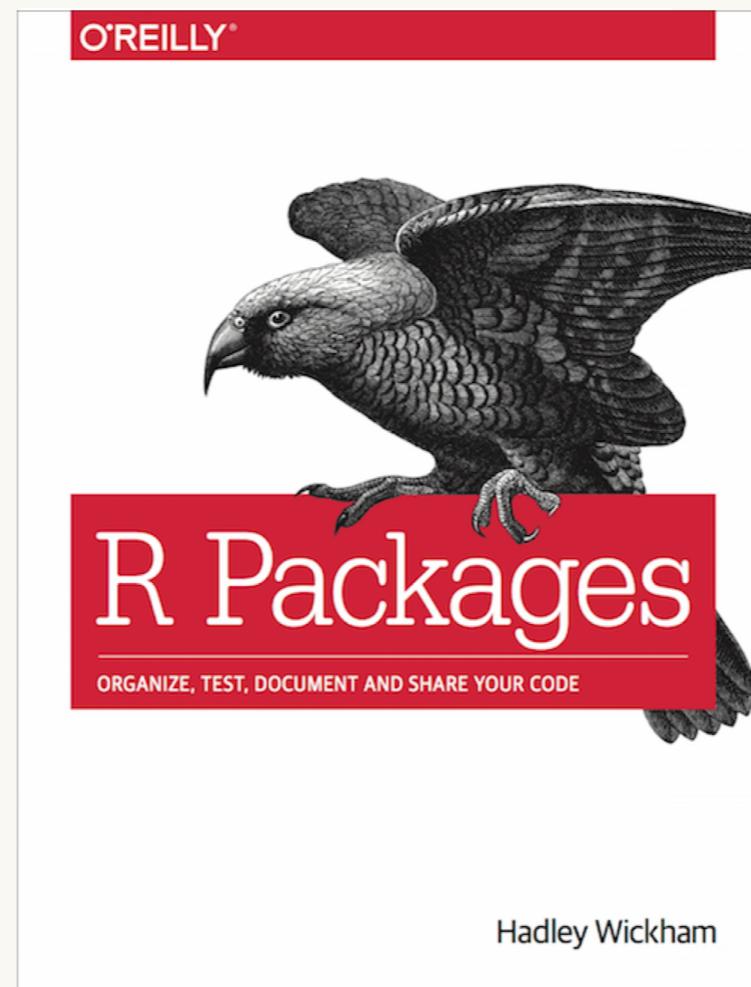
R-package-devel mailing list

# More details on many topics in books



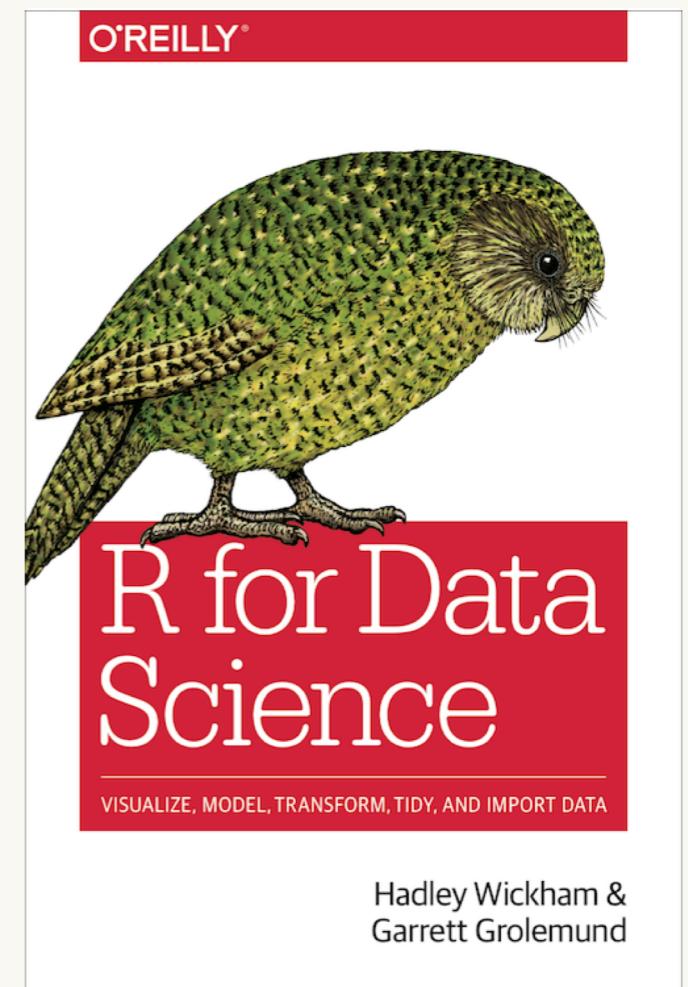
<http://adv-r.hadley.nz/>

<http://amzn.com/1466586966>



<http://r-pkgs.had.co.nz>

<https://amzn.com/1491910399>



<http://r4ds.had.co.nz>

<https://amzn.com/1491910399>

# rweekly.org

R Weekly 2017-37

Social Science, Time, Compare

Highlight

Insights

R in the Real World

R in Academia

R Internationally

R in Organizations

Resources

New Packages

Package Releases

Tutorials

R Project Updates

Upcoming Events

Jobs

Quotes of the Week

**RWeekly Podcast**

Deep and meaningful feedback

**Live**

- Forecasting Data using Alpha-Sutte Indicator ([cran.r-project.org](#))
- Power Calculations for Longitudinal Multilevel Models ([cran.r-project.org](#))
- Interface for OpenDota API ([cran.r-project.org](#))

**More**

**R Weekly 2017-37 Social Science, Time, Compare**

11 Sep 2017

**Highlight**

- Data Visualization for Social Science - A practical introduction with R and ggplot2 ([socviz.co](#))
- It's tibbletime: Time-Aware Tibbles ([business-science.io](#))
- Thou shalt not compare numeric values (except when it works) ([jcarroll.com.au](#))

**Insights**

# #rstats

Home 99+ Notifications Messages #rstats Tweet

## #rstats

Top Latest People Photos Videos News Broadcasts

**People** [View all](#)

**Search filters** [Show](#)

**Related searches**

#datascience  
#machinelearning

**Who to follow** [Refresh](#) [View all](#)

- Pablo Barberá @p\_bar... [Follow](#)
- Timo Grossenbacher ... [Follow](#)
- Statistics Blog @statistic... [Follow](#)

**boB Rudis** [Following](#)

@hrbrmstr Follows you

Don't look at me...I do what he does—just slower. #rstats avuncular • 🍊 Resistance Fighter • Cook • Christia...

Followed by R World News and 56 others

**One R Tip a Day** [@RLangTip](#)

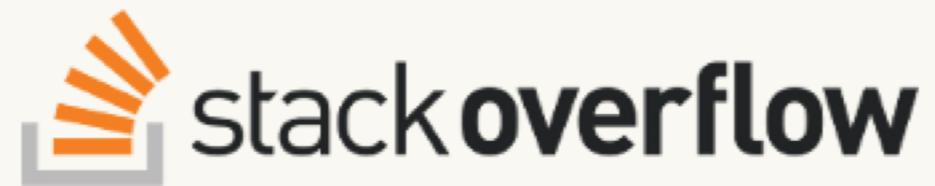
One tip per day M-F on the R programming language #rstats. Brought to you by the R community team at...

Followed by Joseph Rickert and 72 others

**Kieran Healy** [@kjhealy](#) · Sep 7

"Data Visualization for Social Science" is slowly becoming a book. Here's the draft version as a website: [socviz.co](#) #rstats

**for Social Science**



[r] score:5 is:question closed:no

Join us:



<https://rladies.org/>

[@RLadiesDublin](#)

<https://www.meetup.com/R-Ladies-Dublin/>

# Summary:

Modify the following files:

- The DESCRIPTION file
- The files in the R

Do not modify:

- NAMESPACE
- The files in the man folder

This work is licensed under the  
**Creative Commons Attribution-Noncommercial 3.0  
United States License.**

To view a copy of this license, visit

<http://creativecommons.org/licenses/by-nc/3.0/us/>