

# Statistical Computing

Michael Mayer

March 2023



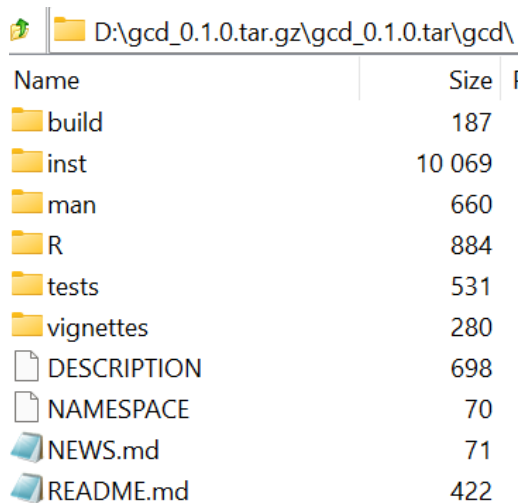
# R Packages: Do it yourself

- ▶ About 20'000 contributed packages on CRAN
- ▶ Many others not on CRAN
- ▶ Making the **second** package is easy

## Resources

- ▶ Writing extensions:  
<https://cran.r-project.org/doc/manuals/r-release/R-exts.html>
- ▶ R Packages (Hadley Wickham and Jennifer Bryan): <https://r-pkgs.org>

## Content of Simple Package “gcd”



Name	Size	I
build	187	
inst	10 069	
man	660	
R	884	
tests	531	
vignettes	280	
DESCRIPTION	698	
NAMESPACE	70	
NEWS.md	71	
README.md	422	

- ▶ R → R code
- ▶ man → Help files
- ▶ build, inst, vignettes  
→ Vignette
- ▶ tests → Unit tests
- ▶ Other files?

## How to create all this??

1. Put functions into one or more R scripts in folder “R”
2. Document functions with Roxygen

### The rest requires three things

- ▶ Package “usethis”: Cares about content of package
- ▶ Package “devtools”: Turns content into package
- ▶ Software bundle RTools: <https://cran.r-project.org/bin/windows/Rtools>

### Example

<https://github.com/mayer79/gcd>

# How to Contribute to other Projects?

## Github project

- ▶ You see issue
- ▶ or missing functionality
- ▶ or simply a typo

## Example

Fix typo in some project

## Example Workflow

1. Fork project on Github
2. Clone your fork locally
3. Create “feature” branch
4. Commit changes to this branch and push
5. Open pull request (PR) to original project
6. If not okay, add more commits locally and push
7. When merged, delete feature branch

## More Complex Packages?

- ▶ What if your package uses other packages like "ggplot2"?  
→ Dependency hell, so minimize their use
- ▶ What if you use `print()`, `summary()` etc.?

### Example

Pandemic apero: <https://github.com/mayer79/apero>