Building a package that lasts
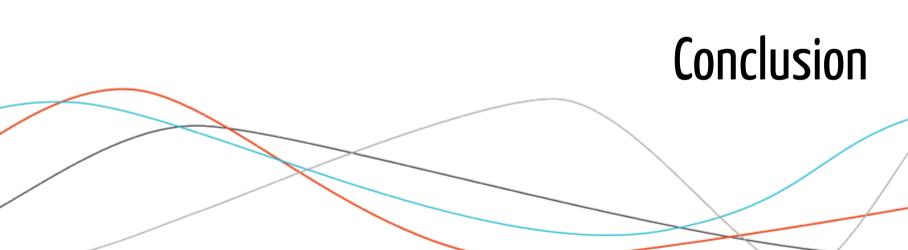
Conclusion

Further readings (in no precise order)

Books and blogs

- "R Packages": http://r-pkgs.had.co.nz/

- 🇫🇷 "Créer un package R en quelques minutes": https://thinkr.fr/creer-package-r-quelques-minutes

- "R package primer" : http://kbroman.org/pkg_primer/

- "Preparing your package for a CRAN submission" : https://github.com/ThinkR-open/prepare-for-cran

Further readings (in no precise order)

Webinars

- "Programming Part 3 (Package writing in RStudio)":
  https://www.rstudio.com/resources/webinars/rstudio-essentials-webinar-series-programming-part-3/

- "You can make a package in 20 minutes – Jim Hester":
  http://www.rstudio.com/resources/videos/you-can-make-a-package-in-20-minutes/

Further tools (in no precise order)

- sinew - "Generate roxygen2 skeletons populated with information scraped from the function script": https://github.com/metrumresearchgroup/sinew

- prefixer - "Prefix function with their namespace" : https://github.com/dreamRs/prefixer

- remedy - "RStudio Addins to Simplify Markdown Writing" : https://github.com/ThinkR-open/remedy

- revdepcheck - "R package reverse dependency checking": https://github.com/r-lib/revdepcheck

Building a package that lasts...

Think "UX first"

- The easier it looks for the user, the better

- Think user first, from function names to documentation

Help yourself

- Automate as much as you can

- Document as much as you can your development process

- Prevent bug at an early stage

- Keep testing

Merci !

**Any question?

*

.center[**Colin Fay**

colin@thinkr.fr

http://twitter.com/_colinfay

http:/s://github.com/ColinFay

http://twitter.com/thinkr_fr

https://github.com/ThinkR-open

https://thinkr.fr/

http://colinfay.me/