Xây dựng các gói

với devtools Cheat Sheet

RStudio

Cấu trúc của gói

Một gói là một tập hợp các tập tin được sắp xếp trong các thư mục theo trật tự nhất định

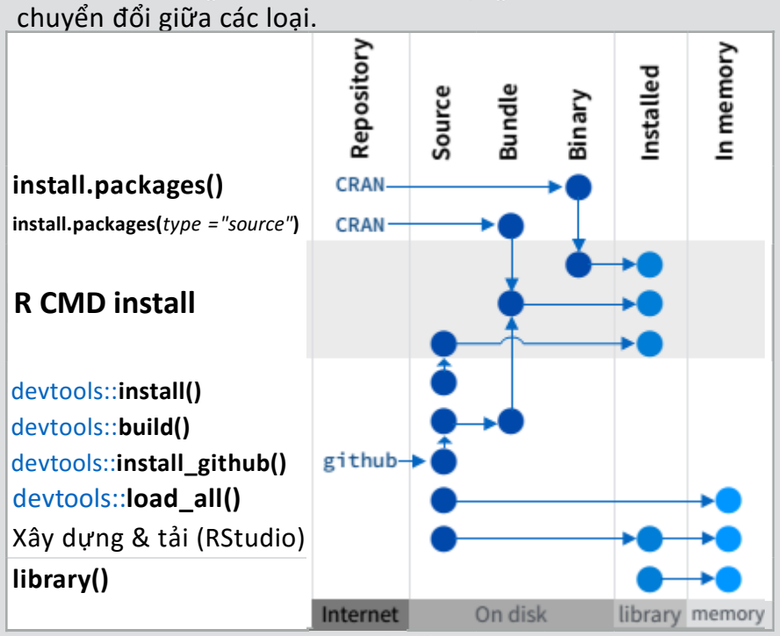
7 thành phần cơ bản của một gói (package) trên R:

Package - Gói	
DESCRIPTION	Thiết lập
R/	Viết code
tests/	Kiểm tra
man/	Tài liệu
vignettes/	Dạy
data/	Dữ liệu
NAMESPACE	Tổ chức

Nội dung của một gói có thể được lưu trên ổ đĩa giống như:

- **source** – là một thư mục với các thư mục con (như trên)
- **bundle** – là một tập tin được nén lại (dạng.*tar.gz*)
- **binary** – là một tập tin được nén lại nhằm hỗ trợ cho một hệ điều hành xác định

Nội dung của 1 gói cũng có thể được tải về thư mục R library (được lưu trên bộ nhớ trong quá trình chạy R) hoặc được lưu online trong 1 kho lưu trữ. Sử dụng các hàm sau để chuyển đổi giữa các loại.



devtools::**add_build_ignore(***"file"***)**

Bổ sung tập tin về .Rbuildignore, một danh sách các tập tin sẽ không được đính kèm khi gói được xây dựng xong.

Thiết lập (DESCRIPTION)

Tập tin DESCRIPTION mô tả thông tin về gói và thiết lập quan hệ giữa gói được tạo với các gói khác.

Bắt buộc phải có tập tin Mô tả (DESCRIPTION)

Bổ sung các gói cần phải sử dụng đi kèm

devtools::**use_package()**

Bổ sung một gói tới mục Nhập (Imports) (hoặc mục Gợi ý - Suggest) (nếu điều kiện thứ 2 là "Gợi ý - Suggest").

CC0	MIT	GPL-2
Không có chuỗi đính kèm	Giấy phép MIT được áp dụng cho code của bạn nếu được chia sẻ lại	Giấy phép GPL-2 được áp dụng cho code của bạn, *và code của bất kỳ ai được đính kèm với nó khi chia sẻ lại.*

```
Package: Tên của gói
Title: Tiêu đề của gói
Package Version: phiên bản của gói (vd: 0.1.0)
Authors@R: person("Hadley", "Wickham", email =
"hadley@me.com", role = c("aut", "cre"))
Description: Tính năng của gói
Depends: R(>=3.1.0) - phiên bản của R
License: GPL-2 - thông tin về giấy phép sử dụng
gói
LazyData: true
Imports:
    dplyr (>= 0.4.0),
    ggvis (>= 0.2)
Suggests:
    knitr (>= 0.1.0)
```

Import Nhập tên các gói đi kèm bắt buộc. R sẽ tải các gói này về khi tải gói của bạn.

Suggest Gợi ý các gói không phải là tối quan trọng với gói của bạn. Người dung có thể tải các gói này theo nhu cầu riêng.

Viết code (R/)

Tất cả code trên R trong một gói sẽ được lưu tại thư mục R/. Dù chỉ với thư mục R/, một gói vẫn có thể hoạt động được

Tạo một gói mới:

devtools::**create(***"Đường dẫn/đến/tên"***)**

Tạo một mẫu để phát triển thành một gói .

Lưu code trong thư mục R/ dưới dạng tập tin(extension .R)

Quy trình

1. Chỉnh sửa code
2. Tải code bằng cách

devtools::**load_all()**

Tải lại tất cả các tập tin đã được lưu trong thư mục R/ lên bộ nhớ.

Ctrl/Cmd + Shift + L (phím tắt)

Lưu tất cả các tập tin được mở sau đó gọi hàm load_all().

3. Thử nghiệm câu lệnh trong giao diện của R (console).
4. Lặp lại quy trình trên.

- Thống nhất cách viết code: **r-pkgs.had.co.nz/r.html#style**
- Nhấp chuột vào 1 hàm rồi nhấn F2 để biết định nghĩa
- Tìm kiếm 1 hàm bằng phím tắt **Ctrl + .**

Tìm hiểu thêm tại

r-pkgs.had.co.nz

Kiểm tra (tests/)

Sử dụng thử mục tests/ để thực hiện Unit Test và kiểm tra lỗi trong các câu lệnh.

Bổ sung thư mục tests/ và tải gói testthat với hàm

devtools::**use_testthat()**

Thiết lập gói để sử dụng các bài kiểm tra tự động từ testthat.

Viết các test case bằng hàm **context()**, **test()** và kết quả kỳ vọng

Lưu bài kiểm tra dưới dạng tập tin.R trong **test/testthat/**

Quy trình

1. Sửa đổi code và kiểm tra
2. Kiểm tra code bằng cách:

devtools::**test ()**

Chạy tất cả tests được lưu trong thư mục test/.

Ctrl/Cmd+Shift+T

(phím tắt)

3. Lặp lại cho đến khi hoàn thành tất cả các bài kiểm tra

Ví dụ

```
context("Arithmetic")

test_that("Math works", {
  expect_equal(1 + 1, 2)
  expect_equal(1 + 2, 3)
  expect_equal(1 + 3, 4)
})
```

expect_equal()	có bằng với khoảng dung sai số nhỏ?
expect_identical()	có bằng một cách chính xác?
expect_match()	khớp với chuỗi ký tự đã được xác định?
expect_output()	có in kết quả đầu ra nhất định?
expect_message()	có trình bày một thông điệp nhất định?
expect_warning()	có trình bày một cảnh báo nhất định?
expect_error()	có thông báo lỗi nhất định?
expect_is()	kết quả đầu ra có thừa hưởng từ một loại nào nhất định?
expect_false()	có trả về giá trị SAI-FALSE?
expect_true()	có trả về giá trị ĐÚNG-TRUE?

Translator: Ranalytics.vn

Tài liệu (man/)

Thư mục man/ chứa tài liệu về các hàm và các trang hỗ trợ cho một gói

- Bên cạnh định nghĩa của hàm, ta có thể sử dụng roxygen để đặc tả từng hàm
- Đặc tả tên của từng tập dữ liệu được xuất ra
- Bổ sung các ví dụ cụ thể cho từng hàm

Quy trình

1. Bổ sung chú giải với roxygen trong tập tin dạng.R
2. Chuyển đổi thông tin chú giải của roxygen thành tài liệu bằng cách

   devtools::**document()**

   Chuyển đổi chú giải của roxygen thành tập tin dạng .Rd và lưu trong thư mục man/. Builds NAMESPACE.

   Ctrl/Cmd + Shift + D (phím tắt)
3. Mở trang hỗ trợ bằng dấu **?** để xem trước tài liệu
4. Lặp lại quy trình

.Rd thẻ định dạng

```
\emph{italic text}
\strong{bold text}
\code{function(args)}
\pkg{package}

\dontrun{code}
\dontshow{code}
\donttest{code}

\deqn{a + b (block)}
\eqn{a + b (inline)}

\email{name@@foo.com}
\href{url}{display}
\url{url}

\link[=dest]{display}
\linkS4class{class}
\code{\link{function}}
\code{\link[package]{function}}

\tabular{lcr}{
  left \tab centered \tab right \cr
  cell \tab cell     \tab cell  \cr
}
```

Gói roxygen

Gói **roxygen** giúp ta viết tài liệu dưới dạng tập tin .R với cú pháp ngắn gọn.

- Bổ sung tài liệu roxygen dưới dạng chú giải bằng cách bắt đầu với dấu **#'**.
- Đặt dòng chú giải ngay phía trên code định nghĩa các đối tượng được đặc tả.
- Đặt dấu **@** roxygen (bên phải) sau dấu **#'** để bổ sung một phần cụ thể trong tài liệu
- Không đánh dấu các dòng sẽ được sử dụng để tạo tiêu đề, mô tả, và các mục chi tiết (theo thứ tự đã đặt)

```
#' Bổ sung 2 số cùng nhau.
#'
#' @param x là biến số.
#' @param y là biên số.
#' @return trả về tổng của \code{x} và \code{y}.
#' @examples
#' add(1, 1)
#' @export
add <- function(x, y) {
  x + y
}
```

Các thẻ phổ biến trong roxygen

@aliases	@inheritParams	**@seealso**
@concepts	@keywords	@format
@describeIn	**@param**	@source data
@examples	@rdname	@include
@export	**@return**	@slot S4
@family	@section	@field RC

Dạy (vignettes/)

Thư mục vignettes/ chứa các tài liệu hỗ trợ người dùng cách xử lý các vấn đề phân tích dữ liệu với gói của bạn

- Tạo thư mục vignettes/ và một mẫu mô tả chi tiết với hàm

  devtools::**use_vignette()**

  Bố sung mẫu dưới dạng vignettes/tên.Rmd.
- Kết nối tiêu đề YAML với tài liệu mô tả (xem ví dụ bên phải)
- Việt nội dung bản mô tả chi tiết trong R Markdown (rmarkdown.rstudio.com)

```
---
title: "tiêu đề của Vignette"
author: "tác giả của Vignette"
date: "`r Sys.Date()`"
output: rmarkdown::html_vignette - kết quả đầu ra
vignette: >
  %\VignetteIndexEntry{tiêu đề của Vignette}
  %\VignetteEngine{knitr::rmarkdown}
  \usepackge[utf8]{inputenc}
---
```

Dữ liệu (data/)

Thư mục data/ cho phép đính kèm dữ liệu trong một gói

- Lưu dữ liệu trong 1 trong thư mục **data/**, **R/Sysdata.rda**, **inst/extdata**
- Luôn sử dụng **LazyData: true** trong tập tin Mô tả (DESCRIPTION)
- Lưu dữ liệu với định dạng .Rdata

devtools::**use_data()**
Thêm một đối tượng dữ liệu vào thư mục data/ (R/Sysdata.rda nếu **internal = TRUE**)

evtools::**use_data_raw()**
Thêm một cú pháp R để làm sạch một tập dữ liệu vào thư mục data-raw/. Đính kèm thư mục data-raw/ vào tập tin .Rbuildignore.

Lưu dữ liệu trong các thư mục

- **data/** cho phép dữ liệu sẵn sàng sử dụng
- **R/sysdata.rda** cho phép dữ liệu được sẵn sàng sử dụng khi gọi các hàm.
- **inst/extdata** cho phép dữ liệu được sẵn cho việc tải và đọc các ví dụ. Truy cập dữ liệu này bằng hàm **system.file()**

Tổ chức (NAMESPACE)

Tập tin NAMESPACE giúp tạo một gói khép kín: gói này sẽ không tác động đến các gói khác và ngược lại

- Xuất các hàm cho người dùng bằng cách đặt **@export** trong chú giải với roxygen
- Nhập các đối tượng từ các gói khác bằng hàm **package::object** (nên dùng) hoặc **@import**, **@importFrom**, **@importClassesFrom**, **@importMethodsFrom** (không nên dùng)

Quy trình

1. Chỉnh sửa code hoặc các bài kiểm tra (tests)
2. Mô tả gói (devtools::**document()**)
3. Kiểm tra NAMESPACE
4. Lặp lại cho tới khi NAMESPACE chính xác

Xuất bản gói của bạn

r-pkgs.had.co.nz/release.html

Translator: Ranalytics.vn