R: Créer un package

Eric Marcon

02 mai 2018

Cadre

Outil diffusable

Trois fois le même code : écrire une fonction.

Trois utilisations de la même fonction : écrire un package.

Formalisme

Standardisation du code.

Vérifications.

Documentation obligatoire.

Outils

devtools et *roxygen2* simplifient le travail.

Création d'un package

Conception

A quoi va servir le package ?

- Consolidation de sa recherche : le package rassemble et organise des méthodes. Exemples : *entropart*, *ade4*.
- Outil d'intérêt général. Exemples : *EcoFoG*, *vegan*, *spatstat*, *ggplot2*.

Un package ne doit traiter qu'un sujet. Si nécessaire, écrire plusieurs packages.

Exemple travaillé

Package *multiple* :

- trois fonctions : `double(x)`, `triple(x)` et `multiple(x, n)`.
- une représentation graphique de `multiple(x, n)` en fonction de `x` (une droite).

Création

Le package est créé dans le dossier indiqué.

```
devtools::create(path = "multiple")
```

Le dossier final a le nom du package.

Ouvrir le projet du package.

Etudier les fichiers créés.

DESCRIPTION


```
Package: multiple
Title: Compute multiples
Version: 0.0.0.9000
Authors@R: person("First", "Last", email =
  "first.last@example.com", role = c("aut", "cre"))
Description: Efficiently calculate multiples of numbe
  following Me et al. (2018).
Depends: R (>= 3.4.3)
License: GNU General Public License
Encoding: UTF-8
LazyData: true
```

Fonctions

Code dans des fichiers .R

Organisation libre, les fichiers ne seront pas dans le package.

Choix :

- un fichier `Project.R` pour le code commun à tout le package
- un fichier R par groupe de fonctions, ici : `Compute.R` et `Plot.R`

Project.R

Code commun

```
#' multiple
#'
#' Spatially Explicit Measures of Diversity
#'
#' @name multiple
#' @docType package
#' @import ggplot2
NULL
```

Compute.R

Première version de `double(x)`

```
double <- function(number) {
    return(2 * number)
}
```

Exécuter le code pour charger la fonction en mémoire.

Tester.

Documenter

Insert Roxygen sleleton :

```
#' double
#'
#' Compute the double value of a vector
#'
#' The double is calculated by multiplying each value
#'
#' @param number A numeric vector
#'
#' @return A vector containing the double values.
#' @export
#'
#' @examples
#' double(runif(3))
```

Oxygéner et installer

Build Source package : crée multiple_0.0.0.9000.tar.gz avec sa documentation.

Etudier le contenu de `man`

Charger le package :

- Nettoyer l'environnement (fonctions en mémoire)
- *Install and Restart*

```
`?`(multiple)
```

```
## No documentation for 'multiple' in specified packa
## you could try '??multiple'
```

Vérifier le package

Check

```
(...)
checking package dependencies ... ERROR
Namespace dependency not required: 'ggplot2'

(...)
R CMD check results
1 error  | 0 warnings | 0 notes
```

Ecrire un générique

Problème : `double()` transforme les entiers en réels.

```
str(double(1L))
```

```
##  num 2
```

Solution :

```
str(1L * 2L)
```

```
##  int 2
```

Ecrire un générique

```
double <- function(x, ...) {
    UseMethod("double")
}
double.default <- function(x) {
    return(2 * x)
}
double.integer <- function(x) {
    return(2L * x)
}
```

Exécuter et tester.

```
str(double(2))
```

```
##  num 4
```

```
str(double(2L))
```

```
##  int 4
```

Documenter

Un seul fichier d'aide.

```
#' double
(...)
#' @param x A vector
#' @param ... Unused
(...)
#' @name double
#NULL

#' @rdname SpeciesDistributions
#' export
double <- function(x, ...) {
  UseMethod("double")
}
```