Reproducible research

R

G. Durif (CNRS – IMAG – Univ Montpellier)

Charles Elie Rabier (Univ Montpellier – IMAG)

Christelle Pierkot (CNRS – IR Data Terra)

Sonia Tieo (CNRS – CEFE – Univ Montpellier)

June 23th 2021, Montpellier

Montpellier Bio-Stats (https://groupes.renater.fr/wiki/montpellier-biostat)

A few references

Rstudio

- https://rviews.rstudio.com/2018/01/18/
  package-management-for-reproducible-r-code
- https://rstudio.github.io/packrat
- https://rstudio.github.io/renv /articles/renv.html

Bioinfo-fr

- https://bioinfo-fr.net/packrat-ou-comment-gerer-ses-packages-r-par-projet

Challenges

- R: a fast-moving community and **many extremely valuable packages**
- Always use **latest version of R** to develop your code
- Upgrade R to the latest version once or twice per year
- **Dependencies are essential** to R development
- Good **dependency management**
  $\Rightarrow$ your project can be recomputed again in another place

Solutions

- **CRAN (=Comprehensive R Archive Network)** archives source code for all versions of R packages
- Possible to rebuild from source for package versions that you used to build an analysis
- **Track package dependencies per project** (e.g. package Packrat, package Checkpoint)

Package Checkpoint vs package Packrat

Both packages use the library-per-project approach

- **package Packrat**
  - stores the exact versions of packages that the project uses
- **package Checkpoint**
  - focuses on CRAN packages
  - determines dependencies based on the "snapshot" of CRAN that Microsoft stored on a given day

Package Packrat

A few examples from Bioinfo-fr

- Which version of the package is required to run the code that my colleague sent to me ?

- Who has never updated a package in a R project in order to make it work ? But now, other projects do not work anymore ...

- Who has never updated a package and all its dependencies ? But now, other projects do not work anymore ...

Package Packrat

Use **packrat** to make your R projects more

- **Isolated**: Installing a new or updated package for one project won't break your other projects

- **Portable**: Easily transport your projects from one computer to another, even across different platforms

- **Reproducible**: it records the exact package versions you depend on, and ensures those exact versions are the ones that get installed wherever you go

https://rstudio.github.io/packrat/

Package Packrat

Basic concepts

- Your package dependencies are stored inside the project directory
- It does not rely on your personal R library that is shared across all of your other R sessions
- This directory = your private package library

More precisely

- It is a Packrat project, not an ordinary R project
- A packrat project has its own private package library
- Any packages you install from inside a packrat project are only available to that project
- Packages you install outside of the project are not available to the project

Package Packrat

The different components

- **packarat.lock**: lists the package versions that were used to satisfy dependencies
- **packarat.opts**: project-specific packrat options
- **init.r**: allows to launch packrat automatically when you open your project
- **lib**: folder that contain your **compiled packages**. It is your **private package library** for this project
- **src**: folder that contains **source packages of all the dependencies** that packrat has been made aware of

How does it work ?

- **packrat::init**(" /Desktop/Workshop/Project/MyFirstProject")

```
Initializing packrat project in directory:
- "~/Desktop/Workshop/Project/MyFirstProject"
Adding these packages to packrat:
    packrat   0.6.0
Fetching sources for packrat (0.6.0) ... OK (CRAN current)
Snapshot written to '~/Desktop/Workshop/Project/MyFirstProject/
packrat/packrat.lock'
Installing packrat (0.6.0) ...
OK (downloaded binary)
Initialization complete!
Packrat mode on. Using library in directory:
 "~/Desktop/Workshop/Project/MyFirstProject/packrat/lib"
```

With Rstudio
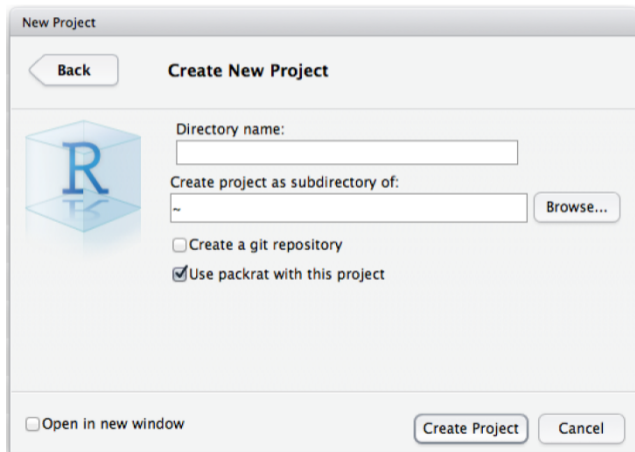
How does it work ?

- **install.packages("reshape2")**

```
# Adding a package in a Packrat project
```

- **packrat:: snapshot()**

```
Adding these packages to packrat:
    Rcpp       1.0.6
    glue       1.4.2
    magrittr   2.0.1
    plyr       1.8.6
    reshape2   1.4.4
...
```

How does it work ?

- **packrat:: status()**

```
Up to date
```

- **remove.packages("plyr")**

- **packrat::status()**

```
The following packages are tracked by packrat,
but are no longer available in the local
library nor present in your code: plyr 1.8.6
```

How does it work ?

- **packrat :: restore()**

```
Installing plyr (1.8.6) ...
OK (downloaded binary)
```

- Once your project has a snapshot $\Rightarrow$ install the packages from that snapshot into your private library at any time
- Useful when copying the project to a new computer (e.g. with a different operating system)

The package Renv replaces the package Packrat

- It is a robust and a stable replacement for the Packrat package
- It helps to manage library paths
- **packrat::init()** : by default, attempt to retrieve package sources from CRAN.
- **renv::init()** : no longer downloads package sources. Instead, the copies already available on your system will be copied and re-used.

Package Checkpoint

package Checkpoint

- focuses on CRAN packages
- determines dependencies based on the "snapshot" of CRAN that Microsoft stored on a given day

A "snapshot" of CRAN from 2016

https://mran.microsoft.com/timemachine

Microsoft R Archived Network

Package Checkpoint

Some useful commands for your R project:

- **create_checkpoint("2021-06-23")**: Creates a checkpoint by scanning a project folder and downloading and installing any packages required from MRAN

- **use_checkpoint()**: Uses a previously created checkpoint, by setting the library search path to the checkpoint path, and the CRAN mirror to MRAN

- **checkpoint()**: similar to create_checkpoint() + use_checkpoint()

- **delete_checkpoint()**: Deletes an existing checkpoint

The package Ape in 2021 : version 5.5

ape: Analyses of Phylogenetics and Evolution

Functions for reading, writing, plotting, and manipulating phylogenetic trees, analyses of comparative data in a phylogenetic framework, ancestral character analyses, analyses of diversification and macroevolution, computing distances from DNA sequences, reading and writing nucleotide sequences as well as importing from BioConductor, and several tools such as Mantel's test, generalized skyline plots, graphical exploration of phylogenetic data (alex, trex, kronoviz), estimation of absolute evolutionary rates and clock-like trees using mean path lengths and penalized likelihood, dating trees with non-contemporaneous sequences, translating DNA into AA sequences, and assessing sequence alignments. Phylogeny estimation can be done with the NJ, BIONJ, ME, MVR, SDM, and triangle methods, and several methods handling incomplete distance matrices (NJ*, BIONJ*, MVR*, and the corresponding triangle method). Some functions call external applications (PhyML, Clustal, T-Coffee, Muscle) whose results are returned into R.

Version:	5.5
Depends:	R (≥ 3.2.0)
Imports:	nlme, lattice, graphics, methods, stats, tools, utils, parallel, Rcpp (≥ 0.12.0)
LinkingTo:	Rcpp
Suggests:	gee, expm, igraph, phangorn
Published:	2021-04-25
Author:	Emmanuel Paradis [aut, cre, cph], Simon Blomberg [aut, cph], Ben Bolker [aut, cph], Joseph Brown [aut, cph], Santiago Claramunt [aut, cph], Julien Claude [aut, cph], Hoa Sien Cuong [aut, cph], Richard Desper [aut, cph], Gilles Didier [aut, cph], Benoit Durand [aut, cph], Julien Dutheil [aut, cph], RJ Ewing [aut, cph], Olivier Gascuel [aut, cph], Thomas Guillerme [aut, cph], Christoph Heibl [aut, cph], Anthony Ives [aut, cph], Bradley Jones [aut, cph], Franz Krah [aut, cph], Daniel Lawson [aut, cph], Vincent Lefort [aut, cph], Pierre Legendre [aut, cph], Jim Lemon [aut, cph], Guillaume Louvel [aut, cph], Eric Marcon [aut, cph], Rosemary

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals

The package Ape in 2019 : version 5.3

ape: Analyses of Phylogenetics and Evolution

Functions for reading, writing, plotting, and manipulating phylogenetic trees, analyses of comparative data in a phylogenetic framework, ancestral character analyses, analyses of diversification and macroevolution, computing distances from DNA sequences, reading and writing nucleotide sequences as well as importing from BioConductor, and several tools such as Mantel's test, generalized skyline plots, graphical exploration of phylogenetic data (alex, trex, kronoviz), estimation of absolute evolutionary rates and clock-like trees using mean path lengths and penalized likelihood, dating trees with non-contemporaneous sequences, translating DNA into AA sequences, and assessing sequence alignments. Phylogeny estimation can be done with the NJ, BIONJ, ME, MVR, SDM, and triangle methods, and several methods handling incomplete distance matrices (NJ*, BIONJ*, MVR*, and the corresponding triangle method). Some functions call external applications (PhyML, Clustal, T-Coffee, Muscle) whose results are returned into R.

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed

Version:	5.3
Depends:	R (≥ 3.2.0)
Imports:	nlme, lattice, graphics, methods, stats, tools, utils, parallel, Rcpp (≥ 0.12.0)
LinkingTo:	Rcpp
Suggests:	gee, expm, igraph
Published:	2019-03-17
Author:	Emmanuel Paradis [aut, cre, cph], Simon Blomberg [aut, cph], Ben Bolker [aut, cph], Joseph Brown [aut, cph], Julien Claude [aut, cph], Hoa Sien Cuong [aut, cph], Richard Desper [aut, cph], Gilles Didier [aut, cph], Benoit Durand [aut, cph], Julien Dutheil [aut, cph], RJ Ewing [aut, cph], Olivier Gascuel [aut, cph], Thomas Guillerme [aut, cph], Christoph Heibl [aut, cph], Anthony Ives [aut, cph], Bradley Jones [aut, cph], Franz Krah [aut, cph], Daniel Lawson [aut, cph], Vincent Lefort [aut, cph], Pierre Legendre [aut, cph], Jim Lemon [aut, cph], Eric Marcon [aut, cph], Rosemary McCloskey [aut, cph], Johan Nylander [aut, cph], Rainer Opgen-Rhein [aut, cph], Andrei-Alin Popescu [aut, cph], Manuela Royer-Carenzi [aut, cph], Klaus Schliep [aut, cph], Korbinian Strimmer [aut, cph], Damien de Vienne [aut, cph]
Maintainer:	Emmanuel Paradis <Emmanuel.Paradis at ird.fr>

Creating a checkpoint from 2021

```
create_checkpoint("2021-06-21")
Creating checkpoint directory
/Users/CE/.checkpoint/2021-06-21/lib/
Got foreach 1.5.1 (macos) (133.74 kB)
Got iterators 1.0.13 (macos) (334.82 kB)
Got MASS 7.3-54 (macos) (1.17 MB)
Got Rcpp 1.0.6 (macos) (3.21 MB)
Got ape 5.5 (macos) (3.61 MB)
Installed MASS 7.3-54  (180ms)
Installed foreach 1.5.1  (252ms)
Installed iterators 1.0.13  (283ms)
Installed ape 5.5  (341ms)
Installed Rcpp 1.0.6  (1.8s)
```

Creating a checkpoint from 2019

create_checkpoint("2019-06-23")
```
Creating checkpoint directory
/Users/CE/.checkpoint/2019-06-23/lib/
Got foreach 1.4.4 (macos) (414.07 kB)
Got ape 5.3 (macos) (2.61 MB)
Installed foreach 1.4.4  (291ms)
Installed iterators 1.0.10  (237ms)
Installed ape 5.3  (223ms)
Installed Rcpp 1.0.1  (1.7s)
```

use_checkpoint("2019-06-23")
```
Using checkpoint directory
/Users/CE/.checkpoint/2019-06-23/lib/
```

Thank you for your attention

Rstudio

- https://rviews.rstudio.com/2018/01/18/package-management-for-reproducible-r-code
- https://rstudio.github.io/packrat
- https://rstudio.github.io/renv /articles/renv.html

Bioinfo-fr

- https://bioinfo-fr.net/packrat-ou-comment-gerer-ses-packages-r-par-projet