Package 'grateful'

November 9, 2026

Title Facilitate Citation of R Packages

Version 0.3.1

Description Facilitates the citation of R packages used in analysis projects. Scans project for packages used, gets their citations, and produces a document with citations in the preferred bibliography format, ready to be pasted into reports or manuscripts. Alternatively, 'grateful' can be used directly within an 'R Markdown' or 'Quarto' document.

License MIT + file LICENSE

URL https://pakillo.github.io/grateful/

BugReports https://github.com/Pakillo/grateful/issues

Depends R (>= 3.0.0)

Imports desc, knitr, remotes, renv, rmarkdown, rstudioapi, utils

Suggests curl, testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

Config/roxygen2/version 8.1.1

NeedsCompilation no

Author Francisco Rodriguez-Sanchez [aut, cre, cph] (ORCID: <https://orcid.org/0000-0002-7981-1599>), Connor P. Jackson [aut] (ORCID: <https://orcid.org/0000-0002-4220-5210>), Shaurita D. Hutchins [ctb], James M. Clawson [ctb]

Maintainer Francisco Rodriguez-Sanchez <f.rodriguez.sanc@gmail.com>

Repository CRAN

Date/Publication 2026-11-09 11:40:02 UTC

Contents

cite_packages 2
get_citations 6
get_csl 7
get_pkgs_info 7
nocite_references 9
scan_packages 10

Index **12**

`cite_packages` *Cite R packages used in a project*

Description

Find R packages used in a project or package, create a BibTeX file of references, and generate a document with formatted package citations. Alternatively, `cite_packages` can be run directly within an 'R Markdown' or 'Quarto' document to automatically include a paragraph citing all used packages and generate a bibliography.

Usage

```
cite_packages(
  output = c("file", "paragraph", "table", "citekeys"),
  out.dir = NULL,
 out.format = c("html", "docx", "pdf", "Rmd", "md", "tex-fragment", "tex-document"),
  citation.style = NULL,
  pkgs = "All",
  omit = c("grateful"),
  cite.tidyverse = TRUE,
  dependencies = FALSE,
  include.RStudio = FALSE,
  passive.voice = FALSE,
  text.start = NULL,
  text.pkgs = NULL,
  text.RStudio = NULL,
  out.file = "grateful-report",
  bib.file = "grateful-refs",
  desc.path = NULL,
  skip.missing = FALSE,
  ...
)
```

Arguments

`output` Either

- "file" to generate a separate document with formatted citations for all packages;
- "paragraph" to return a paragraph with in-text citations of used packages, suitable to be used within an 'R Markdown' or 'Quarto' document;
- "table" to return a table with package name, version, and citations, to be used in 'R Markdown' or 'Quarto';
- "citekeys" to return a vector with citation keys.

In all cases, a BibTeX file with package references is saved on disk (see `bib.file`).

`out.dir` Directory to save the output document and a BibTeX file with the references. It is recommended to set `out.dir = getwd()`.

`out.format` Output format when `output = "file"`: either "html" (the default), "docx" (Word), "pdf", "tex-fragment" (LaTeX fragment to be inserted into another LaTeX document), "tex-document" (full LaTeX document), "Rmd", or "md" (markdown). (Note that choosing "pdf" requires a working installation of LaTeX, see https://yihui.org/tinytex/).

`citation.style` Optional. Citation style to format references for a particular journal (see https://pkg.yihui.org/rmarkdown-cookbook/bibliography). If the CSL is not available in `out.dir`, it will be downloaded automatically from the official GitHub repository using `get_csl()`. If using `cite_packages()` within an R Markdown or Quarto document, `citation.style` should be `NULL` (the default). The citation style should instead be defined in the YAML metadata of the document (see https://pakillo.github.io/grateful/#using-grateful-with-rmarkdown-or-quarto

`pkgs` Character. Either "All" to include all packages used in scripts within the project/folder (the default), "Session" to include only packages used in the current session, or the path to an R script (including `.R` extension), 'Rmarkdown' (`.Rmd`) or 'Quarto' document (`qmd`) to scan only the packages used in that single script or document. Alternatively, `pkgs` can also be a character vector of package names to get citations for. To cite R as well as the given packages, include "base" in `pkgs` (see examples). Finally, `pkgs` can be a character vector of `Depends`, `Imports`, `Suggests`, `LinkingTo` and their combination, to obtain the dependencies of an R package as stated in its DESCRIPTION file (see `desc.path`). Note that in this case, package versions will be 'NA' unless required versions are stated in the DESCRIPTION file (e.g. 'testthat (>= 3.0.0)'), and package citations will use the information from installed versions of those packages in the user computer.

`omit` Character vector of package names to be omitted from the citation report. `grateful` is omitted by default. Use `omit = NULL` to include all packages.

`cite.tidyverse` Logical. If `TRUE`, all tidyverse packages (dplyr, ggplot2, etc) will be collapsed into a single citation of the 'tidyverse', as recommended by the tidyverse team.

`dependencies` Logical. Include the dependencies of your used packages? If `TRUE`, will include all the packages that your used packages depend on.

`include.RStudio` Logical. If `TRUE`, adds a citation for the current version of RStudio.

`passive.voice` Logical. If `TRUE`, uses passive voice in any paragraph generated for citations.

text.start	Optional. Text string to use to start the citation paragraph. If NULL (the default), will use "We used" or "This work was completed using" if passive.voice = TRUE. Note this allows for customising the language of citation paragraphs.
text.pkgs	Optional. Text string to use to introduce the packages used. If NULL (the default), will use "(and) the following R packages".
text.RStudio	Optional. Text string to use to introduce RStudio. If NULL (the default), will use "running in".
out.file	Desired name of the citation report to be created if output = "file". Default is "grateful-report" (without extension).
bib.file	Desired name for the BibTeX file containing packages references ("grateful-refs" by default).
desc.path	Optional. Path to the package DESCRIPTION file from which to parse the package dependencies (see pkgs). If NULL, will default to the working directory.
skip.missing	Logical. If FALSE (the default), will return an error if some package(s) are used somewhere in the project but they are not currently installed. If TRUE, will skip those missing packages, issuing a warning. Note such packages will thus not be included in the citation list, even though they might have been used in the project.
...	Other parameters passed to renv::dependencies().

Details

cite_packages is a wrapper function that collects package names and versions and saves their citation information in a BibTeX file (using get_pkgs_info()).

Then, the function is designed to handle different use cases:

If output = "file", cite_packages() will generate an 'R Markdown' file which includes a paragraph with in-text citations of all packages, as well as a references list. This document can be knitted to various formats via out.format. References can be formatted for a particular journal using citation.style. Thus, output = "file" is best for obtaining a document separate from R, to just cut and paste citations.

If output = "paragraph", cite_packages() will return a paragraph with in-text citations of all packages, suitable to be used directly in an 'R Markdown' or 'Quarto' document. To do so, include a reference to the generated bib.file bibliography file in the YAML header of the document (see https://pakillo.github.io/grateful/index.html#using-grateful-within-rmarkdown).

Alternatively, if output = "table", cite_packages() will return a table with package names, versions, and citations. Thus, if using 'R Markdown' or 'Quarto', you can choose between getting a table or a text paragraph citing all packages.

Finally, you can use output = "citekeys" to obtain a vector of citation keys, and then call nocite_references() within an 'R Markdown' or 'Quarto' document to cite these packages in the reference list without mentioning them in the text.

Value

If output = "file", cite_packages will save a citation report in out.dir with formatted citations, and cite_packages will return the path to the citation report invisibly. If output = "table" or

`output = "paragraph"`, `cite_packages` will return a table or paragraph with package citations suitable to be used within 'R Markdown' or 'Quarto' documents. A BibTeX file containing package references is saved in all cases in `out.dir`.

Limitations

Citation keys are not guaranteed to be preserved when regenerated, particularly when packages are updated. This instability is not an issue when citations are used programmatically, as in the example below. But if references are put into the text manually, they may need to be updated periodically.

Examples

```
# To build a standalone document for citations
cite_packages(out.dir = tempdir())

# Format references for a particular journal:
cite_packages(citation.style = "peerj", out.dir = tempdir())

# Choose different output format:
cite_packages(out.format = "docx", out.dir = tempdir())

# Cite only packages currently loaded:
cite_packages(pkgs = "Session", out.dir = tempdir())

# Cite only user-provided packages:
cite_packages(pkgs = c("renv", "remotes", "knitr"), out.dir = tempdir())

# Cite R as well as user-provided packages
cite_packages(pkgs = c("base", "renv", "remotes", "knitr"), out.dir = tempdir())

# To change the language of the citation paragraph:
cite_packages(output = "paragraph", out.dir = tempdir(),
  text.start = "Para desarrollar este trabajo se utilizó",
  text.pkgs = "y los siguientes paquetes")


# To include citations in an R Markdown or Quarto file

# include this in YAML header:
# bibliography: grateful-refs.bib

# then call cite_packages within an R chunk:
cite_packages(output = "paragraph", out.dir = tempdir())

# To include package citations in the reference list of an Rmarkdown document
# without citing them in the text, include this in a chunk:
nocite_references(cite_packages(output = "citekeys", out.dir = tempdir()))
```

`get_citations` *Get citations for packages*

Description

Get citations for packages

Usage

```
get_citations(
  pkgs = NULL,
  out.dir = NULL,
  bib.file = "grateful-refs",
  include.RStudio = FALSE,
  skip.missing = FALSE
)
```

Arguments

`pkgs`	Character vector of package names, e.g. obtained by `scan_packages()`.
`out.dir`	Directory to save the output document and a BibTeX file with the references. It is recommended to set `out.dir = getwd()`.
`bib.file`	Desired name for the BibTeX file containing packages references ("grateful-refs" by default).
`include.RStudio`	Logical. If `TRUE`, adds a citation for the current version of RStudio.
`skip.missing`	Logical. If FALSE (the default), will return an error if some package(s) are used somewhere in the project but they are not currently installed. If TRUE, will skip those missing packages, issuing a warning. Note such packages will thus not be included in the citation list, even though they might have been used in the project.

Value

A file on the specified `out.dir` containing the package references in BibTeX format. If assigned a name, `get_citations` will also return a list with citation keys for each package (without @).

Examples

```
citekeys <- get_citations(c("knitr", "renv"), out.dir = tempdir())

pkgs <- scan_packages()
citekeys <- get_citations(pkgs$pkg, out.dir = tempdir())
```

get_csl *Get a journal citation style from the official internet repository*

Description

Get a journal citation style from the official internet repository

Usage

```
get_csl(name = NULL, out.dir = NULL)
```

Arguments

name	Name of the journal, exactly as found in https://github.com/citation-style-language/styles.
out.dir	Directory to save the CSL file.

Value

The CSL file is saved in the selected directory, and the path is returned invisibly.

Examples

```
get_csl("peerj", out.dir = tempdir())
```

get_pkgs_info *Get information about packages used in a project*

Description

This function scans the project for R packages used, saves a BibTeX file with package references, and returns a data frame with package names, version, and citation keys.

Usage

```
get_pkgs_info(
  pkgs = "All",
  out.dir = NULL,
  omit = c("grateful"),
  cite.tidyverse = TRUE,
  dependencies = FALSE,
  bib.file = "grateful-refs",
  include.RStudio = FALSE,
  desc.path = NULL,
  skip.missing = FALSE,
  ...
)
```

Arguments

pkgs	Character. Either "All" to include all packages used in scripts within the project/folder (the default), "Session" to include only packages used in the current session, or the path to an R script (including .R extension), 'Rmarkdown' (.Rmd) or 'Quarto' document (qmd) to scan only the packages used in that single script or document. Alternatively, pkgs can also be a character vector of package names to get citations for. To cite R as well as the given packages, include "base" in pkgs (see examples). Finally, pkgs can be a character vector of Depends, Imports, Suggests, LinkingTo and their combination, to obtain the dependencies of an R package as stated in its DESCRIPTION file (see desc.path). Note that in this case, package versions will be 'NA' unless required versions are stated in the DESCRIPTION file (e.g. 'testthat (>= 3.0.0)'), and package citations will use the information from installed versions of those packages in the user computer.
out.dir	Directory to save the BibTeX file with references. It is recommended to set out.dir = getwd().
omit	Character vector of package names to be omitted from the citation report. grateful is omitted by default. Use omit = NULL to include all packages.
cite.tidyverse	Logical. If TRUE, all tidyverse packages (dplyr, ggplot2, etc) will be collapsed into a single citation of the 'tidyverse', as recommended by the tidyverse team.
dependencies	Logical. Include the dependencies of your used packages? If TRUE, will include all the packages that your used packages depend on.
bib.file	Desired name for the BibTeX file containing packages references ("grateful-refs" by default).
include.RStudio	Logical. If TRUE, adds a citation for the current version of RStudio.
desc.path	Optional. Path to the package DESCRIPTION file from which to parse the package dependencies (see pkgs). If NULL, will default to the working directory.
skip.missing	Logical. If FALSE (the default), will return an error if some package(s) are used somewhere in the project but they are not currently installed. If TRUE, will skip those missing packages, issuing a warning. Note such packages will thus not be included in the citation list, even though they might have been used in the project.
...	Other parameters passed to renv::dependencies().

Value

A data.frame with package info, and a file containing package references in BibTeX format.

Examples

```
get_pkgs_info(out.dir = tempdir())
get_pkgs_info(pkgs = c("renv", "remotes"), out.dir = tempdir())
```

`nocite_references` *Generate Nocite Block*

Description

Include a metadata block of citation keys for including citations in references file without in-text citations.

Usage

```
nocite_references(citekeys, citation_processor = c("pandoc", "latex"))
```

Arguments

`citekeys`	Vector of citation keys in reference to a relevant BibTeX file.
`citation_processor`	Mechanism for citation processing when knitting to PDF or otherwise using LaTeX. Selects the appropriate formatting of the nocite command. Either "pandoc" or "latex". If processing with pandoc-citeproc, use the nocite metadata block. If processing via a LaTeX processor such as natbib or biblatex, put in the LaTeX `\nocite{}` command directly. If knitting to any non-LaTeX format, this parameter is ignored, and a pandoc-citeproc style block is used.

Details

When passed a list of citation keys, adds the @ to each, then builds the nocite metadata field, returning via "as-is" output. Run this function either inline or within a code chunk (with `echo = FALSE`) to include this metadata block within an RMarkdown document. The code chunk need not explicitly state `results = 'asis'`.

Call `nocite_references` with either `style = 'pandoc'` or `style = 'latex'` depending on whether you are processing citations with pandoc-citeproc or a LaTeX citation processor such as biblatex or natbib.

This function is intended to cite R packages with citation keys passed from `get_citations()` or `cite_packages(output = "citekeys")`, but can accept an arbitrary vector of citation keys (without @) found in a BibTeX file referenced in the YAML header.

Value

"As is" text of metadata block, with comma-separated list of citation keys.

Author(s)

Connor P. Jackson

Examples

```
# include in YAML header:
# bibliography: grateful-refs.bib

# Get citation keys for the current RMarkdown document
# (run after all packages have been loaded).
citekeys <- cite_packages(output = "citekeys", out.dir = tempdir())

# Include in RMarkdown body for use with pandoc-citeproc:
# `r nocite_references(citekeys, citation_processor = 'pandoc')`
```

`scan_packages` *Scan a project or folder for packages used*

Description

Scan a project or folder for packages used

Usage

```
scan_packages(
  pkgs = "All",
  omit = c("grateful"),
  cite.tidyverse = TRUE,
  dependencies = FALSE,
  desc.path = NULL,
  skip.missing = FALSE,
  ...
)
```

Arguments

`pkgs`	Character. Either "All" to include all packages used in scripts within the project/folder (the default), "Session" to include only packages used in the current session, or the path to an R script (including `.R` extension), 'Rmarkdown' (`.Rmd`) or 'Quarto' document (`qmd`) to scan only the packages used in that single script or document. Alternatively, `pkgs` can also be a character vector of package names to get citations for. To cite R as well as the given packages, include "base" in `pkgs` (see examples). Finally, `pkgs` can be a character vector of `Depends`, `Imports`, `Suggests`, `LinkingTo` and their combination, to obtain the dependencies of an R package as stated in its DESCRIPTION file (see `desc.path`). Note that in this case, package versions will be 'NA' unless required versions are stated in the DESCRIPTION file (e.g. 'testthat (>= 3.0.0)'), and package citations will use the information from installed versions of those packages in the user computer.
`omit`	Character vector of package names to be omitted from the citation report. `grateful` is omitted by default. Use `omit = NULL` to include all packages.

`cite.tidyverse`	Logical. If `TRUE`, all tidyverse packages (dplyr, ggplot2, etc) will be collapsed into a single citation of the 'tidyverse', as recommended by the tidyverse team.
`dependencies`	Logical. Include the dependencies of your used packages? If `TRUE`, will include all the packages that your used packages depend on.
`desc.path`	Optional. Path to the package DESCRIPTION file from which to parse the package dependencies (see `pkgs`). If NULL, will default to the working directory.
`skip.missing`	Logical. If FALSE (the default), will return an error if some package(s) are used somewhere in the project but they are not currently installed. If TRUE, will skip those missing packages, issuing a warning. Note such packages will thus not be included in the citation list, even though they might have been used in the project.
`...`	Other parameters passed to `renv::dependencies()`.

Value

a data.frame with package names and versions

Examples

```
scan_packages()
scan_packages(pkgs = "Session")
scan_packages(pkgs = c("renv", "remotes", "knitr"))
```

Index

cite_packages, 2
cite_packages(), *3*

get_citations, 6
get_citations(), *9*
get_csl, 7
get_csl(), *3*
get_pkgs_info, 7
get_pkgs_info(), *4*

nocite_references, 9
nocite_references(), *4*

renv::dependencies(), *4*, *8*, *11*

scan_packages, 10
scan_packages(), *6*