

Automating R Package Citations in Reproducible Research Documents

Christopher Gandrud

Abstract All of the R packages a reproducible computational research document relies on should be fully cited. Because the syntax and capabilities of an R package change over time, if the version used is not fully cited it will be difficult to actually reproduce the research that depends on it. In this short note, I introduce the `LoadandCite` command from the **repmis** R package. It makes it much easier to document the R packages a piece of research depends on. If included in a dynamic reproducible research document, `LoadandCite` can (a) load all the R packages used to create the document, (b) create a BibTeX file that is updated every time the document is compiled with each loaded package's full citation information. It also allows the user to automatically install all of the packages, including specific versions. Using this command makes it much easier to create really reproducible research with R.

One of R's main advantages is its very active community of package developers who are rapidly expanding and improving its capabilities through user-created packages. Many pieces of research depend on these packages. Though R's active package development community enables this research, the fact that it is so active and packages can change very quickly means that it can be difficult to reproduce research that depends on particular package versions. Therefore, for a piece of research to be reproducible, it is very important that a researcher fully cite—including the version number—all of the R packages that their results depend on (see Jackson, 2012, for more details).¹

Though very important, we can do even better than just creating a static reference list of package citations. First to make our research really reproducible it should be reasonably *easy* for other researchers to not only understand how results were achieved in the abstract, but also replicate those results. Literate programming technologies, such as **knitr** (Xie, 2013a), have made it much easier to reproduce research results. Would-be reproducers, can however face a problem when they compile these documents: they need to install any package dependencies that they don't already have. This may involve hunting down specific versions of the packages used in the original paper in order to have the versions of the packages used to create the original research. Second, though we may intend to fully cite the versions of the packages we used in our research, we may fail to accurately do so. This is especially true if there are a number of packages that are updated over the course of the research process. If the package version cited differs from the one actually used and there have been syntax or other changes to the packages between the two versions, then it could be very difficult to reproduce the research.

In this short note I propose and demonstrate a solution to these two problems. Researchers can use the `LoadandCite` command from the **repmis** package (Gandrud, 2013b)² in their reproducible research documents to (a) load all of the packages that they used to create the document, (b) create a BibTeX file that is updated every time the document is compiled with each loaded package's full citation information, and (c) automatically install specific package versions.

Basic Syntax

First let's look at the basic `LoadandCite` syntax. Below, we'll examine ways to use it in reproducible LaTeX documents. The main argument that `LoadandCite` takes is a character vector of package names. For example, to load the **car** (Fox and Weisberg, 2013) and **knitr** packages create the following character vector:

```
Packages <- c('car', 'knitr')
```

We can now simply add this vector object to `LoadandCite` and specify the name of the BibTeX file that we want to create and save the citations to.³ Let's use the file name 'Example.bib':

```
repmis::LoadandCite(pkgs = Packages, file = 'Example.bib')
```

¹Of course it is also important to cite the version of R you are using for the same reasons. This paper was written with R version 3.0.1. The full source code files used to create this paper are available at: <https://github.com/christophergandrud/LoadandCiteNote>

²I would like to thank Karthik Ram, who contributed code to the command.

³If no file is specified, then the packages are only loaded.

The two packages will be loaded and a BibTeX file with citation information for the loaded version created in the working directory. **Note** it is a good idea to use a file name that is different from the one you use for non-R package citations. Otherwise you will accidentally overwrite your other citations.

If we want to have the packages installed as well as loaded and cited, we can set the argument `install = TRUE`. This will install the most recent version of the packages from the repositories specified in the `repo` argument.⁴ If we want to install specific package versions, we can include a character vector of package version numbers to have `LoadandCite` install from the Comprehensive R Archive Network (CRAN). The order of the package versions must match the order of the packages listed in the `pkgs` argument. For example, to install **car** version 2.0-17 and **knitr** version 1.1 use:

```
Vers <- c('2.0-17', '1.1')

repmis::LoadandCite(pkgs = Packages, install = TRUE,
                    versions = Vers, file = 'Example.bib')
```

You should avoid using old package versions in active research projects. So, it is probably better to specify the versions only in documents you are making available for replication.⁵

If you are reproducing a piece of research and installing old versions of packages, it can be a good idea to install these into a separate library from the one that you normally use. You can specify what library to have `LoadandCite` install the packages into with the `lib` argument.⁶

Using LoadandCite in a Knitted LaTeX Document

Possibly the best way to use `LoadandCite` with your research is placing it in a ‘code chunk’ at the beginning of a **knitr** created LaTeX document⁷ and using it to load every R package used to create the document. For example:

```
<<include=FALSE>>=
  Packages <- c('car', 'knitr')
  repmis::LoadandCite(pkgs = Packages, file = 'Example.bib')
@
```

The code `<<include=FALSE>>=` and `@` delimit the beginning and ending of the R code chunk. The `include=FALSE` argument tells **knitr** to not include any messages, warnings, and so on created by running the code in the compiled presentation document. For more information on how to create reproducible research documents with **knitr** see Xie (2013b) and Gandrud (2013a).

You can cite the packages using BibTeX cite keys as usual. The BibTeX keys created by `LoadandCite`⁸ follow the following formula `R-PACKAGE_NAME`. For example, the **car** package’s cite key will be `R-car`. To insert a citation like ‘(Fox and Weisberg, 2013)’ simply type: `\citep{R-car}` in the text of your LaTeX document where you would like the citation to appear. Additionally add the name of the BibTeX file created by `LoadandCite` to your `bibliography` command near the end of your LaTeX document, i.e. `\bibliography{Example}`. Note that because **repmis** is itself an add-on R package, you should clearly inform anyone who would want to reproduce your research that they need to install it first before running `LoadandCite`. You could do this in a README file accompanying the replication files.

Summary

R’s capabilities are developing very quickly. To be able to take advantage of these expanding capabilities while also making our research easily reproducible in the future, we need to make sure that the add-on packages we use are fully cited and that it is easy for future users to install the versions we used. Using `LoadandCite` in dynamically created research documents helps us do this.

⁴The default is the user’s default.

⁵If `install = FALSE` (the default) then specific package versions will not be installed, i.e. `LoadandCite` will ignore any values in the `versions` argument.

⁶If needed, remember to then add this library path to R with `.libPath`.

⁷The command also works with a **knitr** Markdown document rendered with <http://johnmacfarlane.net/pandoc/>. This can be useful for creating reproducible webpages. See the <http://yihui.name/knitr/demo/pandoc/> for how to use Pandoc from R.

⁸The keys are generated with code based on **knitr**’s `write_bib` command. `LoadandCite` does not formally depend on **knitr** to make it possible to install old versions of that package.

Bibliography

- J. Fox and S. Weisberg. *car: Companion to Applied Regression*, 2013. URL <http://CRAN.R-project.org/package=car>. R package version 2.0-18. [p1, 2]
- C. Gandrud. *Reproducible Research with R and RStudio*. Chapman and Hall/CRC Press, Boca Raton, FL, 2013a. [p2]
- C. Gandrud. *repmis: A collection of miscellaneous tools for reproducible research with R.*, 2013b. URL <http://cran.r-project.org/web/packages/repmis/>. R package version 0.2.5.1. [p1]
- M. Jackson. How to cite and describe software. *Software Sustainability Institute*, 2012. <http://software.ac.uk/so-exactly-what-software-did-you-use>. [p1]
- Y. Xie. *knitr: A general-purpose package for dynamic report generation in R*, 2013a. URL <http://CRAN.R-project.org/package=knitr>. R package version 1.2. [p1]
- Y. Xie. *Dynamic Documents with R and knitr*. Chapman and Hall/CRC Press, Boca Raton, FL, 2013b. [p2]

Christopher Gandrud
Yonsei University
1 Yonseidae-Gil, Wonju
Gangwon-do, 220-710
Republic of Korea christopher.gandrud@gmail.com