

Bite Size R

The Package Ecosystem

Keith Hurley

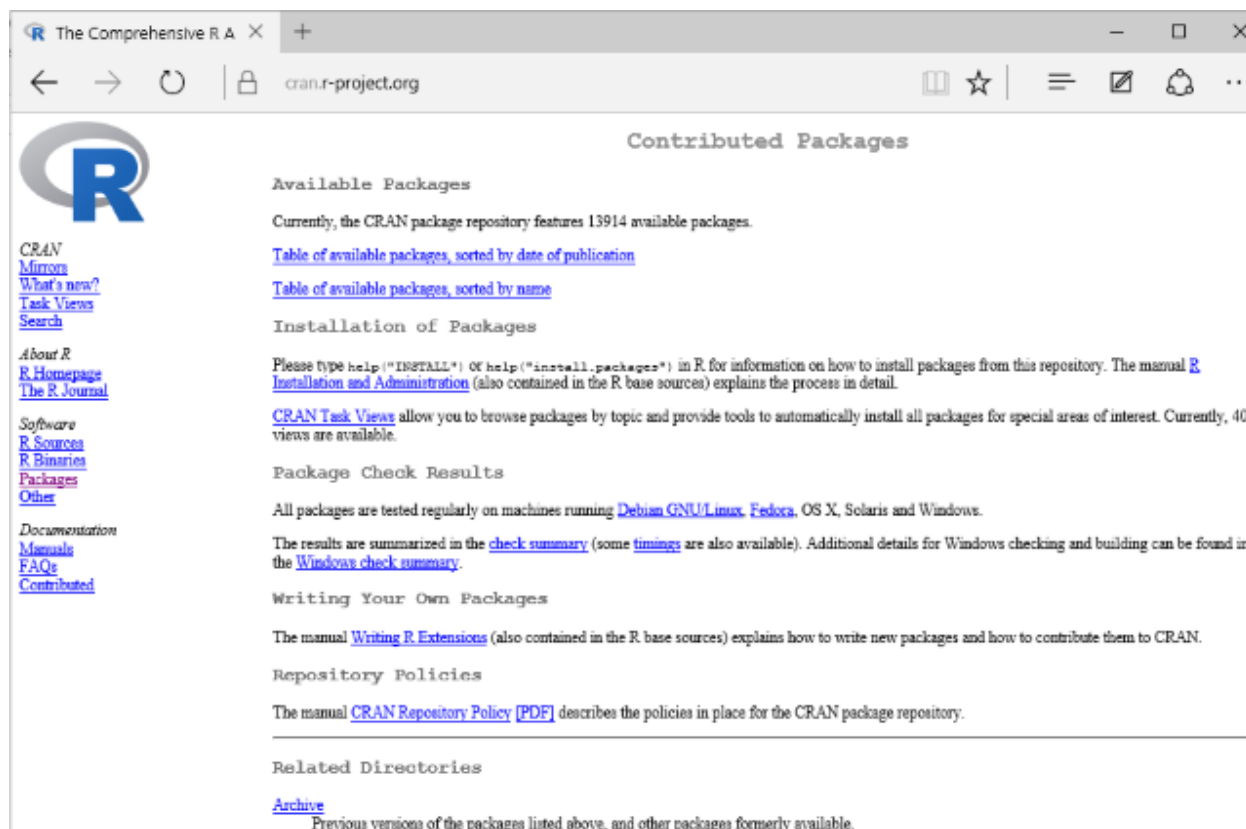
April 12, 2019

In today's snack we're going to take a look at the package ecosystem that's an integral part of using R.

The R software contains not only the R interpreter but also a large number of functions and capabilities that were part of R when it was first created. These functions are generally referred to now as "Base R".

R Packages

Over time R has evolved and gained many new capabilities through its package ecosystem. Packages are self-contained entities, usually revolving around a central topic or theme, that contain functions, code, datasets, and templates that expand the base capabilities of R. Not only that, but some packages will install other packages as well! CRAN (the Comprehensive R Archive Network) is the central repository for packages that have been submitted to the R-Project and meet some mandatory requirements and CRAN currently contains over 13,000 packages.



Many companies and organizations also have created their own packages specific to their domain which are distributed internally and NOT available publicly via CRAN. Many individuals, myself included, even maintain a personal package that encapsulates often-used pieces of code.

The Comprehensive R A X +

mirror.las.iastate.edu/CRAN

ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics

A system for 'declaratively' creating graphics, based on "The Grammar of Graphics". You provide the data, tell 'ggplot2' how to map variables to aesthetics, what graphical primitives to use, and it takes care of the details.

Version: 3.1.0
Depends: R (≥ 3.1)
Imports: digest, grid, rtables (≥ 0.1.1), lazyeval, MASS, magritr, plyr (≥ 1.7.1), reshape2, rlang (≥ 0.2.1), scales (≥ 0.5.0), stats, tibble, viridisLite, withr (≥ 2.0.0)
Suggests: covr, dplyr, ggplot2movies, hexbin, Hmisc, lattice, mapproj, maps, maptools, multcomp, numstat, nlme, testthat (≥ 0.11.0), vdiff, quantreg, knitr, rgeos, rpart, markdown, sf (≥ 0.3-4), rvglite (≥ 1.2.0.9001)
Enhances: sp
Published: 2018-10-25
Author: Hadley Wickham [aut, cre], Winston Chang [aut], Lionel Henry [aut], Thomas Lin Pedersen [aut], Kohske Takahashi [aut], Claus Wilke [aut], Kara Woo [aut], RStudio [cph]
Maintainer: Hadley Wickham <hadley@rstudio.com>
BugReports: https://github.com/tidyverse/ggplot2/issues
License: GPL-2 | file LICENSE
URL: http://ggplot2.tidyverse.org, https://github.com/tidyverse/ggplot2
NeedsCompilation: no
Citation: ggplot2 citation info
Materials: README NEWS
In view: Graphics, Phylogenetics, TeachingStatistics
CRAN checks: ggplot2 results

Downloads:

Reference manual: ggplot2.pdf
Vignettes: Extending ggplot2
Aesthetic specifications
Package source: ggplot2_3.1.0.tar.gz
Windows binaries: r-devel: ggplot2_3.1.0.zip, r-release: ggplot2_3.1.0.zip, r-oldrel: ggplot2_3.1.0.zip
OS X binaries: r-release: ggplot2_3.1.0.tgz, r-oldrel: ggplot2_3.1.0.tgz
Old sources: ggplot2 archive

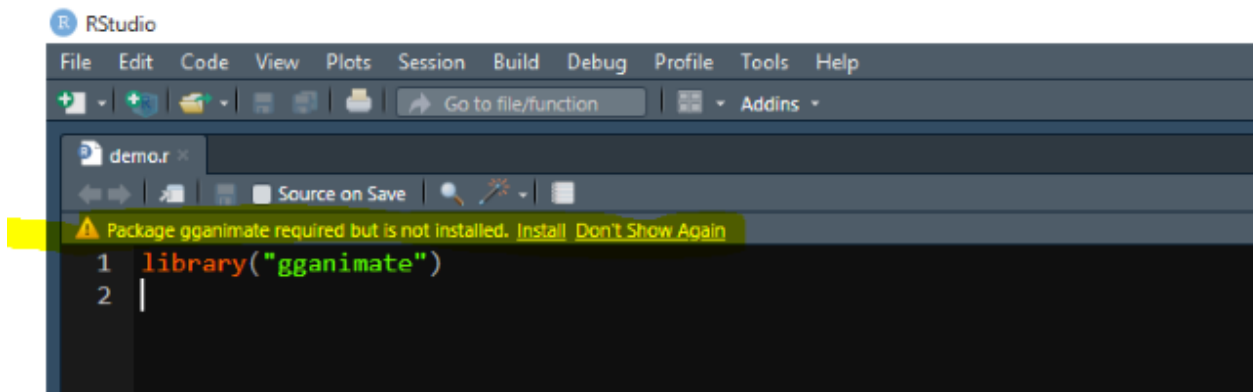
Reverse dependencies:

Reverse depends: ACSNMR, afnToolKit, alakazam, AmpliconDuo, asriscat, apsimr, BatchMap, bayesDP, BCellMA, bde, biomod2, bootstrapReports, btergm, CA3variants, caret, CAVariants, centralplot, ceterisParibus, cjoin, classifierplots, climwin, clustered, CNVScope,

Vignettes are tutorials to help you understand how to use the packages and a great way to get started with a new package.

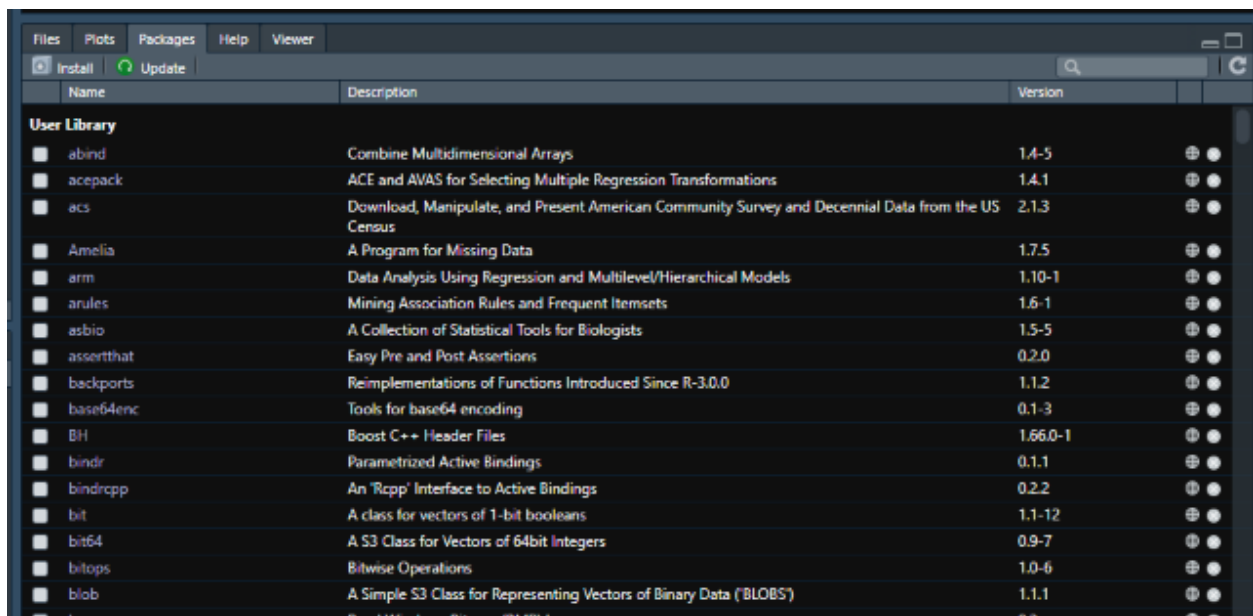
Installing R Packages

2

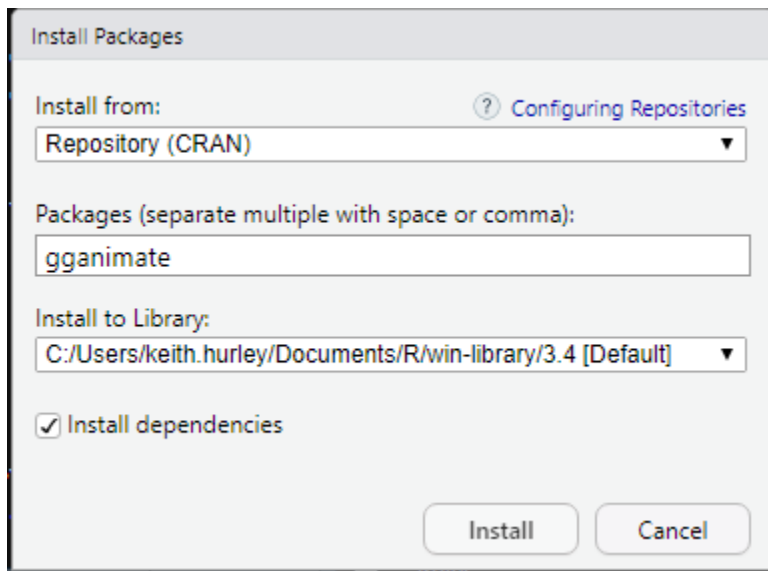


Install From The Package Tab

The package tab is normally located in the lower right corner of RStudio and provides a point and click method of installing packages. Packages that show up in this tab are already installed on your computer.



By clicking "Install", a pop-up window allows you to specify which package you want to install and then will make it happen!



Install From Code

It is also possible to install packages by writing it directly into R code - either in the console or a script file. When a package is loaded from the package tab pop-up window, you will notice that RStudio actually writes the code for you in the console.

```
Console  Jobs x
~/ ↗
> install.packages("gganimate")
Installing package into 'C:/Users/keith.hurley/Documents/R/win-library/3.4'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/gganimate_1.0.2.zip'
Content type 'application/zip' length 1612903 bytes (1.5 MB)
downloaded 1.5 MB

package 'gganimate' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\keith.hurley\AppData\Local\Temp\RtmpURKhTg\downloaded_packages
> |
```

To install an R package, the syntax is:

```
install.packages("gganimate")
```

Notice that the name of the package must be quoted so R knows that it's a name and not a variable that contains the name. It's also possible to load multiple packages at once by providing a vector of names:

```
install.packages(c("gganimate", "ggeffects", "ggedit"))
```

Loading a package

In order to use a package in your R code, you must load it into the memory of your R session. This session is reset every time you restart R. It's good practice to try to load only those packages that you'll be using rather than a laundry list of all of your favorites as they all take up memory that you may need for data! I

currently have over 400 packages installed on my desktop, but I usually will only use up to maybe a dozen at a time. For this reason, you **DO** need to load packages in each script or session of R.

The code for doing uses the library function like this:

```
library("gganimate")
```

Unlike packages, libraries can only be loaded one at a time. Also unlike packages, names do not need to be enclosed in quotes as the following will also work:

```
library(gganimate)
```

Wrap Up

That's it for this time. Now you know how to expand on R's capabilities using packages. If there is something that you want to do in R, it's likely a package exists that can make it happen! Happy snacking!