

Rsquared Academy



Working with R Packages

In this module, we will learn about R packages. There are ~12000 packages available on [CRAN](#) or the Comprehensive R Archive Network. Packages are available for different [topics](#) and you should always look for a package before writing code from scratch. In case you have written your own codes for a new analysis or topic, do share it with the R community by converting the R scripts into a package. You can learn more about building R packages from [R Packages](#), a book written by [Hadley Wickham](#).

In this post, we will learn to:

- install R packages from
 - CRAN
 - GitHub
 - BitBucket
 - Bioconductor
 - rForge
- install different versions of a package
- load, update & remove installed packages
- access package documentation

install packages

Packages from CRAN can be installed using `install.packages()`. The name of the package must be enclosed in single or double quotes.

```
install.packages('ggplot2')
```

Some R packages are made available on [GitHub](#) before releasing them on CRAN. Such packages can be installed using `install_github()` from [devtools](#) or [remotes](#) package. You need to specify the name of the repository and the package. For example, to download [ggplot2](#) or [dplyr](#), below is the code:

```
devtools::install_github("tidyverse/ggplot2")  
remotes::install_github("tidyverse/dplyr")
```

Bitbucket is similar to GitHub. You can install packages from Bitbucket using `install_bitbucket()` from `devtools` or `remotes` package.

```
devtools::install_bitbucket("dannavarro/lsr-package")  
remotes::install_bitbucket("dannavarro/lsr-package")
```

Bioconductor provides tools for analysis and comprehension of high throughput genomic data. Packages hosted on Bioconductor can be installed in multiple ways:

devtools

Use `install_bioc()` from devtools.

```
install_bioc("SummarizedExperiment")
```

biocLite

Use `biocLite()` function.

```
source('http://bioconductor.org/biocLite.R')  
biocLite('GenomicFeatures')
```


Many R packages are hosted at [R-Forge](https://r-forge.r-project.org/), a platform for development of R packages.

```
install.packages('quantstrat', repos = 'https://r-forge.r-project.org/')
```

Install Different Versions

Now that we have learnt how to install packages, let us look at installing different versions of the same package.

```
remotes::install_version('dplyr', version = 0.5.0)
```

If you want to install the latest release from GitHub, append `@*release` to the repository name. For example, to install the latest release of `dplyr`:

```
remotes::install_github('tidyverse/dplyr@*release')
```

Installed Packages

Function	Description
<code>installed.packages()</code>	View currently installed packages.
<code>library('package_name')</code>	Load package into the current R session.
<code>available.packages()</code>	List of packages available for installation.
<code>old.packages()</code>	List of packages which have new versions available.
<code>new.packages()</code>	List of packages already not installed.
<code>update.packages()</code>	Update packages which have new versions available.
<code>remove.packages('package_name')</code>	Remove installed packages.

Library is a directory that contains all installed packages. Usually there will be more than one R library in your system. You can find the location of the libraries using `.libPaths()`.

```
.libPaths()
```

```
## [1] "C:/Users/HP/Documents/R/win-library"  
## [2] "C:/Program Files/R/R-3.5.0/library"
```

You can use `lib.loc` when you want to install, load, update and remove packages from a particular library.

Load

To use the functionalities offered by a package, we must load it into the current R session. Use `library()` to load a R package and specify the location of the package using the `lib.loc` argument.

```
library(lubridate, lib.loc = "C:/Program Files/R/R-3.4.1/library")
```



Thank You

For more information please visit our website
www.rsquaredacademy.com