R packages

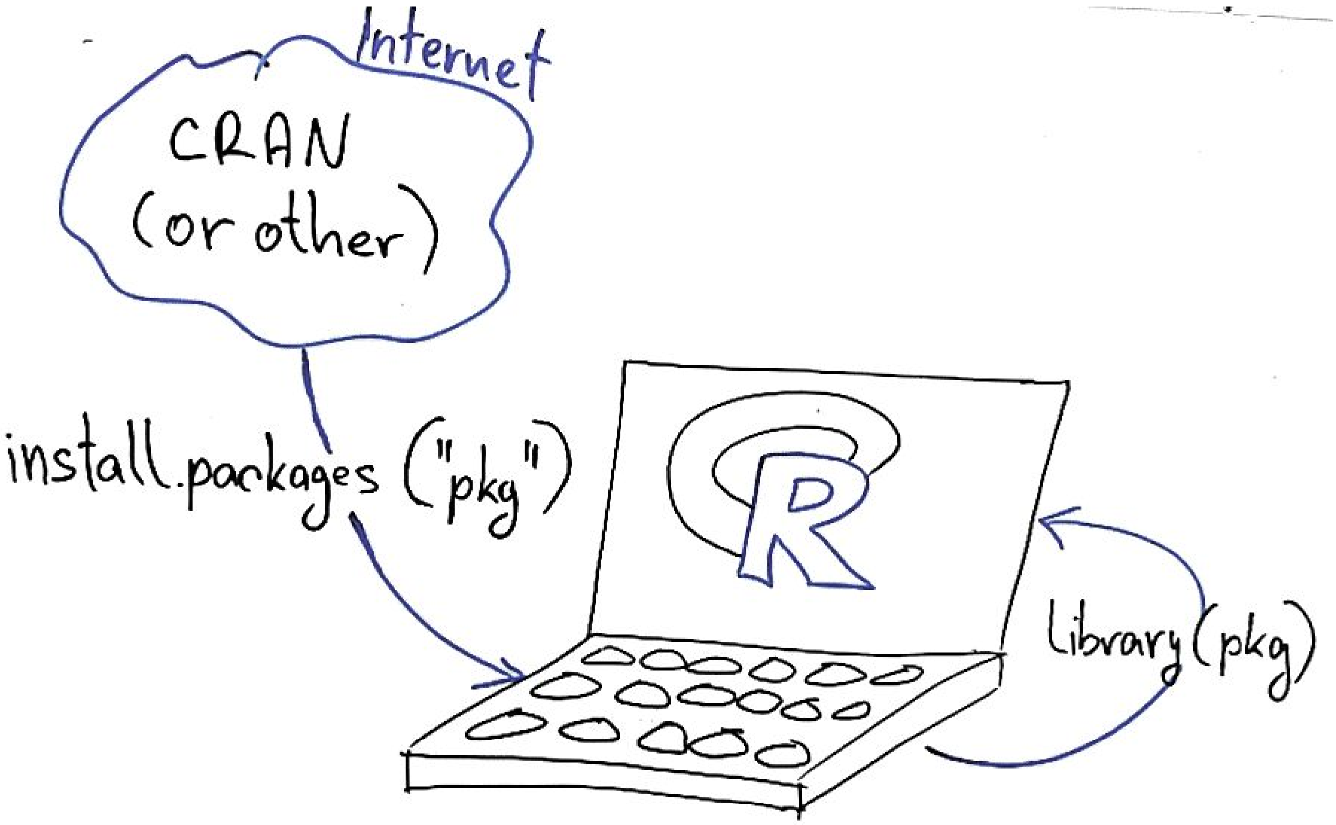
R has a great option to extend the basic functionality for other functions wrapped into packages. There are thousands of packages in central repositories like CRAN or Bioconductor, and also in developers repositories like R-Forge or GitHub. For an overview of documentation for packages aggregated from these repositories check the website R Documentation.

Installing package is a two step process. First you need to install it from online repository (eg CRAN), and second you need to upload it into R workspace in order to use it.

1. `install.packages (“pkg”)`: From the online repository, you need to install the package. You need to do this only once for given R version (unless the package neeeds to be updated). Installing R package will download the package somewhere into your computer, see below. Details about how to install packages from different online storage places is also below.
2. `library (pkg)`: If you want to use it in current R session, you need to upload it into R workspace. You need to do this every time you open R session and do the coding, in which you need some functionality of the package.

Note: in `install.packages`, the name of the package should be enclosed in quotation marks (`install.packages (“pkg”)`), while in `library` it does not need to (both `library (pkg)` and `library (“pkg”)` work).

Rule: You need to install the package on your computer before you can upload it. You need to upload the package into R before you can use it.



How to install R packages

If you have administrator rights for the computer, the packages will be installed into the `library` subfolder, usually located in *c:\\Program Files\R\R-x.x.x\* [1]. If you don't have administrator rights, you will be asked to install the libraries into some writeable location of the hard disc, by default into the `Documents` subfolder of your account (here, the following structure of subfolders will be created: *Documents\R\win-library\3.2*).

Installing from CRAN

Most of the commonly used R packages for analysis of ecological data are available from CRAN, and to install them, use simply `install.packages` command in R (note that the name of the package should be separated by quotation marks, and if more than one library should be installed, wrap the names into character vector):

```
install.packages ("ade4")
```

If you are using RStudio, you may use built-in function to install the packages, which is (in default setting of RStudio) located in bottom right panel, in the folder *Packages* - click the button *Install*, in *Install from* choose *Repository (CRAN)*, and in *Packages* type the name of the package.

There are, however, examples of packages which are not available from CRAN for some reason - e.g. because they are still under development, or because they do not fit rather strict CRAN rules for hosted packages. Alternative places are e.g. R-Forge or GitHub.

Installing from GitHub

GitHub doesn't offer precompiled package versions - you need to compile the packages from source on your computer. However, there is a convenient R function in R package `devtools` developed by Hadley Wickham, which can help you with installing packages hosted on GitHub: `install_github`. To install some package (here called `MyPackage`, first install `devtools` from CRAN, and then apply the function `install_github`:

```
install.packages ("devtools")
devtools::install_github ("MyPackage")
```

(note that `::` means that `install_github` function will be read from the `devtools` namespace without need to actually upload the whole `devtools` library). If the GitHub package contains C/C++/Fortran code which needs to be compiled, you still need to install the set of compilation tools called Rtools on your computer.

Installing from R-Forge

R-Forge contains both source packages (files with extension `*.tar.gz` containing the original R code, which needs to be compiled during installation) and also already compiled binaries for Windows (`*zip` files, do not need to be built). The binaries are available only for the latest R version available, so if you are using the older version of R, you need to build the library from source. For example, to install package `packfor`[2] for forward selection of variables directly from R-Forge, type

```
install.packages("packfor", repos="http://R-Forge.R-project.org")
```

If you are using the latest R version, the package will probably install without problems. If not, you may receive the following warning message:

```
Warning in install.packages :
  package ‘packfor’ is not available (for R version 3.1.2)
```

In that case, you need to install the package from source. Manually download the binary file from R-Forge (with `*.zip` extension) to your computer, and use the command `install.packages` with argument `repos = NULL`:

```
install.packages ("C:/Users/Downloads/packfor_0.0-8.zip", repos = NULL)
```

(note that this works if your downloaded file is the version of packfor 0.0-8 and it is saved in Downloads folder - modify this for your current needs). If even this option doesn't work, you may need to install the package from the source - download `*.tar.gz` version to your computer and use `install.packages` function with `repos = NULL` and `type = “source”`:

```
install.packages ("C:/Users/Downloads/packfor_0.0-8.tar.gz", repos = NULL,
type = "source")
```

However, installing R packages from source is nontrivial task and may require that your computer has installed set of compilation tools, so called Rtools, because installing from source consists of several steps (compiling C/C++/Fortran codes, making documentations, checking for code errors, running examples etc.).

[1] R-x.x.x stems for the R version, e.g. R-3.2.1

[2] Note that both forward selection function from this package have been included in the `adespatial` package available on CRAN.