

Upgrade R Environment/Libraries (Mac OSX) without a Disruption of the Present CRAN and Bioconductor Infrastructure

Aleksandr B. Sahakyan

22 October 2015

R programming language and many of its packages are under continuous development, with frequent major releases happening several times in a year. For computational biologists/developers, the best practice is to work on the most up-to-date version of R, so that the code would benefit from the increased execution performance and robustness. Starting from the latest version of R would also assure the absence of the usage of obsolete language features. However, as of now, a magical command that would simply upgrade your whole R environment into both the latest version of R and the latest versions of installed libraries, does not exist. During the course of the usage of the R environment, you would certainly accumulate/install many add-on libraries, the continuous presence of which could be vital for your projects. Here, I describe a way of upgrading R and all its installed libraries on Mac OSX, without leaving a trace from the previous version, and without losing any add-on feature due to a library missed upon the upgrade.

Step 1. Find out where your current R is installed. You can do this by either consulting with the installation notes specific to Mac OSX found in the R home page (<https://www.r-project.org/>), or by searching your file system (the commands “which” and “find” might be handy). As of October 2015, my R executable was at

```
# /Library/Frameworks/R.framework/Resources/bin/R
```

and had a soft link at

```
# /usr/local/bin/R
```

The GUI app was installed at

```
# /Applications/R.app
```

Make a note of this information, to be used later (**Step 4**) for cleaning the old version.

Step 2. Find out the names of the packages installed on your present R environment, along with their location.

To retrieve information on all the installed libraries, open R from the command line and type the following

```
liblist <- installed.packages()
```

The location(s) where the packages are installed can be found through

```
unique(liblist[, "LibPath"])
```

```
## [1] "/Library/Frameworks/R.framework/Versions/3.2/Resources/library"
```

Make a note of this information, to be used later (**Step 4**) for cleaning the old version.

Step 3. Differentiate the packages that are installed from R CRAN repository from the ones installed from Bioconductor and save the information for further re-installation/upgrade of the libraries at **Steps 5** and **7**.

Load *biocLite* from Bioconductor to have access to the *all_group* function that lists all the possible packages in the latest release of Bioconductor.

```
source("http://bioconductor.org/biocLite.R")
allbioc <- all_group()
```

Find out the names of installed CRAN and Bioconductor packages, by mapping the names on *allbioc* to identify the installed Bioconductor packages, then save the names in plain text files for further usage on an upgraded R environment.

```
liblist <- as.vector(liblist[, "Package"])
bioc.ind <- match(liblist, allbioc)
cran.ind <- which(is.na(bioc.ind))
bioc.ind <- bioc.ind[-cran.ind]

bioclist <- allbioc[bioc.ind]
cranlist <- liblist[cran.ind]

write(bioclist, file="bioclist.txt")
write(cranlist, file="cranlist.txt")
```

Step 4. Delete the locations where R and the add-on libraries are installed, as learned from the **Steps 1** and **2**.

As an example, my commands were:

```
$ sudo rm -rf /Library/Frameworks/R.framework
$ sudo rm /usr/local/bin/R
$ sudo rm -rf /Applications/R.app
```

Step 5. Install the latest version of R from <https://www.r-project.org/>.

First, make sure your X11 system is up to date by opening the XQuartz app on Mac OSX and checking for updates.

Install latest R from the corresponding Mac binaries available from the R home page. I usually prefer a source-code-based installation at a custom location on Linux systems, but a binary-based installation for Mac OSX, since compilation from the source code requires many dependencies that are normally missing in the default Mac OSX setup.

The installation process should have automatically created the soft link at */usr/local/bin/R*.

If this is no longer the case, as verified by the inability of calling R from the command prompt (terminal), then create the link at */usr/local/bin* manually:

```
$ sudo ln -sf /Library/Frameworks/R.framework/Resources/bin/R /usr/local/bin/R
```

where the first address is the full path to the installed R executable.

Step 6. Install all the CRAN packages, the names of which have been stored at *cranlist.txt* file at **Step 3**, from within R:

```
install.packages(readLines("cranlist.txt"))
```

Be attentive to spot possible error messages and find the packages that rise problems with the new version of R. You can address the problems by trying the installation process for those problematic packages once again, at the end, after their possible dependences are already installed on the system.

Step 7. Install Bioconductor and all the previously installed Bioconductor packages.

```
source("http://bioconductor.org/biocLite.R")  
biocLite()  
biocLite(pckgs=readLines("bioclist.txt"))
```

Step 8. Have a slice of pizza; you are done!