Top of Form

Bottom of Form

**Upgrade R Without Losing Your Packages**

May 23, 2017

Since the first publication of this post, a couple of packages have emerged to automate this process. The installr for Windows and the updateR for OS X are particularly good. However, this continues to be a popular post, so I have decided to keep it up. This work-flow is short, sweet, and cross-platform.

**1. Before you upgrade, build a temp file with all of your old packages.**

tmp <- installed.packages()

installedpkgs <- as.vector(tmp[is.na(tmp[,"Priority"]), 1]) save(installedpkgs, file="installed\_old.rda")

**2. Install the new version of R and let it do it’s thing.**

**3. Once you’ve got the new version up and running, reload the saved packages and re-install them from CRAN.**

tmp <- installed.packages()

installedpkgs.new <- as.vector(tmp[is.na(tmp[,"Priority"]), 1])

missing <- setdiff(installedpkgs, installedpkgs.new) install.packages(missing) update.packages()

**If you had any packages from BioConductor, you will need to reload those as well.**

chooseBioCmirror()

biocLite()

load("installed\_old.rda")

tmp <- installed.packages()

installedpkgs.new <- as.vector(tmp[is.na(tmp[,"Priority"]), 1])

missing <- setdiff(installedpkgs, installedpkgs.new)

for (i in 1:length(missing)) biocLite(missing[i])

All done, now you can get back to cracking out R code. This method helped me save a lot of time, hope someone else finds it useful!