**Guidelines for version numbers of our R packages**

1. Versions of our R packages have the form x.y.z, where the numbers x, y, z are all integers greater than or equal to zero. We don’t use leading zeros unless the integer is exactly 0.

2. Before making any changes to a package on our GitHub repository (<https://github.com/pbs-software>), “pull” the current code to your local working copy. You may already be up-to-date, but make sure that someone else hasn’t changed something since you last checked.

3. Make any appropriate changes to the current version that you deem necessary. Then change the version number in DESCRIPTION; typically add 1 to z in x.y.z. (You might alternatively decide that y or x deserves changing. See <http://semver.org/>, but allow subjective choices in the definition of “minor” or “major” changes.) If you increase y, reset z to 0. If you increase x, reset both y and z to 0.

4. Ideally, update ChangeLog.txt, although Nick makes a good point that the commit message should accomplish something equivalent.

5. “Stage” your changes and “commit” them, with a “push” to the remote server. Be sure to add a meaningful commit message.

I’m assuming that the main actions (pull, push, stage, commit) have the same meaning in every context. So far, I’m using SourceTree (<https://www.sourcetreeapp.com/>) as my interface.

Jon Schnute

September 7, 2015