Making a package from base R files

John C. Nash

11/06/2021

Background

This article tries to explain an approach to developing alternative versions of functions which are in the distributed base of R. Our interest was in developing improvements to the nls() function and related features in R as part of a Google Summer of Code project for which Arkajyoti Bhattacharjee is the funded student. However, nls() has many tentacles involving a number of files and functions that may or may not be called as `nls()` is executed.

Part of the difficulty in carrying out such development of alternative versions is that one needs to be able to execute the new variants in parallel with the existing ones. A heavy-effort approach would be to have separate full sets of R code and build each system and run them separately.

Attempts to provide functions to switch between sets of functions were not promising, and not pursued.

Duncan Murdoch suggested, and generously provide an almost-complete prototype of, a package of interlocked functions that provide the set of `nls()` capabilities. This package `nlspkg` has been cloned to `nlsalt` which is being modified to provide the new capabilities or new expression of the code. Moreover, these capabilities, such as a function `Anyfn()` can be tested side by side by use of the double colon syntax, namely,

"' nlspkg::Anyfn() # existing functionality or code # and nlsalt::Anyfn() # New functionality of code

Options

Packaging from base R files

Developing an alternative set of functions and alternative package