STAT 585X: Data Technologies for Statistical Analysis

Project Report: SASnatch: Using SAS Naturally

Spring 2014, Dr. Diane Cook, Dr. Heike Hofmann

Name:	Ian Mouzon
email:	imouzon@iastate

Instructions:	Due Friday April 7th at 3:10
Assignment:	Draft of report from course project

Background

> *R is a leading language for developing new statistical methods. [...] We know a lot of our users have both R and SAS in their toolkit, and we decided to make it easier for them to access R by making it available in the SAS environment.*
>
> - Bob Rodriguez, *Senior Director of Statistical Development at SAS*

> *On several levels, R represents a threat to SAS, which is the largest seller of commercial statistics software.*
>
> - Ashlee Vance, *Bits Blog (The New York Times)*, February 16, 2009

The first statistical programming language I learned to use was SAS, and the first few courses I used it in were made more difficult and frustrating as a result. I often felt like the assignments in the course were about struggling to get SAS to give me the results I wanted while the exams were focused on the non-SAS concepts that actual made up the material of the course, meaning a lot of my time outside of class was spent unproductively. While this has very little to do with SAS itself, which I have since found to be a very powerful tool and one that is very easy to use, my experience can hardly be unique. While I have met many students in graduate school who are willing to tweak minute details in the `R` code they have written, I have not met a single student who does the same in SAS: in fact, I know only a few students who are willing to use SAS at all.

I wind up using SAS on an almost daily basis now. Still, there are many aspects of SAS that I still find frustrating, oftern all the more so because R does it so well (anything that can be accomplished in qplot, say) or because SAS does them in such a frustrating way (how do you find the sum of 2 and 2 in R? There are many ways, but are any of them as fast and unconscious as typing `2 + 2`?).

The way programming in SAS even feels different to me than programming in R. R feels more open, with every object being manipulatable and easy to work with, and when you have written good code, you feel like you have actually made something yourself. Feeling like I have been creative is very personally rewarding and makes hours spent programming, even on minute details, enjoyable. I even customized my opening message in R, which is a waste of time by almost any criteria except one: I went from an idea (can I customize the opening message) to a result (I am greeted by a message *that I wrote*). I created a very small something. With all of its `proc`s and `data` steps, SAS feels like almost claustrophobic, and the opportunity to feel creative is harder to find. Even the SAS environment feels like a box to me, with an Output window, a log window, a results bar, and so on.

That is, my problems with using SAS fit roughly into two categories:

- There are steps that I would rather perform in R, and

- SAS's workflow feels restrictive.

SAS has provided many tools that I could take advantage of (for instance, using R through SAS/IML and the Output Delivery System (ODS) to control the amount of output) but these tools are not as convenient or simple as I would like them to be. `knitr` on the other hand feels much more natural, leading me to work on this tool: `SASnatch`.

SASnatch: Using SAS naturally

Or perhaps `knitr`ally. The idea is simple: SAS has powerful features that would be great to take advantage of but unfortunately lacks many perks of using R (even if the main perk is merely that you find R more familiar). `knitr` is a great tool for creating reproducible documents but does not work with SAS. `SASnatch` is a bridge that allows R and SAS to work together by executing SAS code on R objects and returning the results to R.

Usage

When setting options for a `knitr` chunk, simply set the `useSAS` chunk option to `TRUE` and the `SASnatch` option as being equal to a character vector containing the names of the objects that are intended to be shared with SAS. For example:

and then `Snatch`ing the results back to R.