



Introduction to R Flex Presentation of Fall '10

Geoff Hagopian

August 26, 2010

What is R?

This is the first question in the R-FAQ document that comes bundled with the R installation. Paraphrasing that document:

R is a system for doing statistical computation and creating illustrative graphics. This includes

- the R programming language
- a run-time environment with graphics
- a debugger
- access to certain system functions, and
- the ability to run programs stored in script files



Figure 1: R, me matey

The R distribution contains functionality for a large number of statistical procedures. Among these are:

- linear and generalized linear models
- nonlinear regression models
- time series analysis
- classical parametric and nonparametric tests
- clustering and smoothing

A strong point of R is its robust set of functions providing a flexible graphical environment for data presentations. Additional modules (add-on packages) are available for a variety of specific purposes. The R home page is <http://www.R-project.org/>. It is free software distributed under a GNU-style copyleft, and an official part of the GNU project (GNU S).

Who uses R?

- researchers in public and private industry
- faculty and students in mathematics, science and social science
- pirate radio wannabes who end up flopping on the flex circuit

Obviously, there are many R tutorials that have been created for people just getting started in R.

I like the [introduction to R at Illinois State](#), for instance. The useR! user group has an annual meeting and you can read their papers at <http://user2010.org/>
There are tutorials: [Tutorials](#)

You can view the [presentations sorted by topics](#)

Some of the lectures and discussion panels from the useR! conference may be of interest:

[Invited Lectures Page](#)

In particular, the videos with imbedded slide shows are nice. The lecturers are typically professors from major universities and there's a panel discussion involving a number of different industrial practitioners - which is a bit droll, to me anyway.

[Richard Stallman](#) gave an inspirational address at the useR! conference. He says, "I decided to develop a free operating system or die trying...of old age, of course." He followed the design of UNIX because he wanted it to be portable to various kinds of systems. He wanted it to be upward compatible with UNIX. He called it "GNU" which is a recursive acronym for "GNU is not UNIX" What does the 'G' stand for? Well, it's silent...hardy har har! It's not 'new' so, Doug was quite right about it's pronunciation. There is a deep philosophical debate going on about free software and the GNU system. Go [here](#) for some documents on this, or if you have a couple of hours in the car (or wherever) to listen to a lecture, check out [Stallman's lecture at the useR! conference](#).

In my view, the R environment is probably much more illustrative of how mainstream statistics is performed now and will be in future than are the licensed softwares such as MS Office S-plus, SAS and SPSS, etc. The advantages of being able to

- Data ownership: free open source softwares (FOSS) allow users to open files they create and/or own as long as needed. This is especially crucial when upgrading hardware.
- FOSS components have been used by many people in different devices and have proved their reliability.
- FOSS components undergo checks by many people and companies and can't hide their flaws or traps for long.
- FOSS offer a wider choice of higher quality software and freeware.
- The customer is in control and can fine tune the system according to their needs.
- FOSS offer greater security and privacy.
- FOSS allows users to create proprietary applications unless these involve Copyleft libraries or extending existing Copyleft programs.
- Multiple companies compete to offer support for FOSS.

Yet many of the information science courses at COD rely on the commercial (costs money to buy) software when the GNU packages are superior and free.

There are technical challenges associated with FOSS such as R. These include:

- Frequent FOSS upgrades and releases and the need to freeze components early enough. (This, as Doug pointed out, is also an issue with proprietary software.)
- Too many solutions to choose from - this is where expert consulting may be critical.
- A fine tuned custom solution may be superior to a general purpose solution.
- International law governing licenses may be hard to keep up with.