Introduction to R and its ecology

Kevin Little, Ph.D. Informing Ecological Design, LLC

Contents

What is R? Let's go to the Source! (www.r-project.org)

Quoted from the R Project website

Introduction to R

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open Source route to participation in that activity.

One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.

R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.

The R environment

R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes

- *an effective data handling and storage facility,*
- *a suite of operators for calculations on arrays, in particular matrices,*
- *a large, coherent, integrated collection of intermediate tools for data analysis,*
- *graphical facilities for data analysis and display either on-screen or on hardcopy, and*
- *a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.*

The term "environment" is intended to characterize it as a fully planned and coherent system, rather than an incremental accretion of very specific and inflexible tools, as is frequently the case with other data analysis software.

R, like S, is designed around a true computer language, and it allows users to add additional functionality by defining new functions. Much of the system is itself written in the R dialect of S, which makes it easy for users to follow the algorithmic choices made. For computationally-intensive tasks, C, C++ and Fortran code can be linked and called at run time. Advanced users can write C code to manipulate R objects directly.

Many users think of R as a statistics system. We prefer to think of it of an environment within which statistical techniques are implemented. R can be extended (easily) via packages. *There are about eight packages supplied with the R distribution and many more are available through the CRAN family of Internet sites covering a very wide range of modern statistics.*

R has its own LaTeX-like documentation format, which is used to supply comprehensive documentation, both on-line in a number of formats and in hardcopy.

Getting Started with R

There are many books (147 listed on the R project site, www.r-project.org on 21 Dec 2014), more written every day, but it helps to have a gentler start that doesn't assume you know a lot.

1. A short clear introduction to R: the basics

http://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf

2. A summary from Computerworld:
   http://www.computerworld.com/article/2497143/business-intelligence-beginner-s-guide-to-r-introduction.html

3. Free Introductory Course ("about 4 hours")

https://www.datacamp.com/courses/introduction-to-r?gclid=CKOW5OrExsICFQaPaQodWboAEA

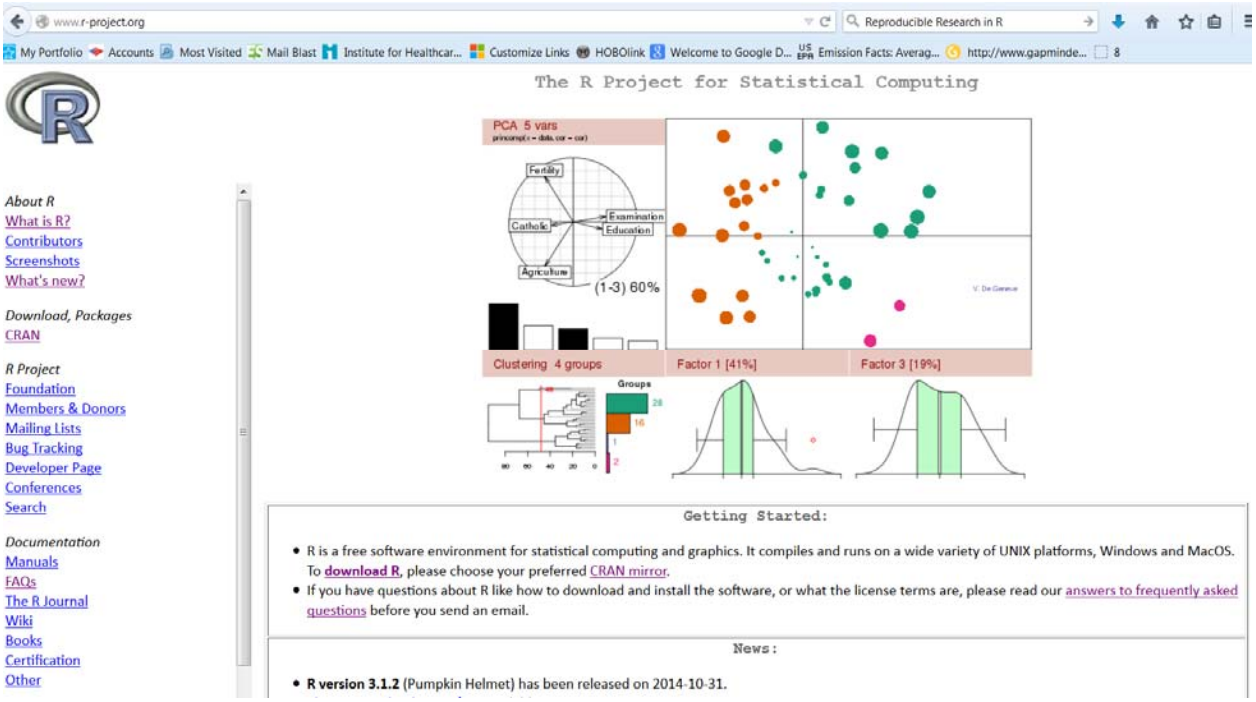
4. For people who may have learned SAS or Stata or SPSS, this blog post describes similarities and differences with R

http://r4stats.com/articles/why-r-is-hard-to-learn/

5. Read the overview about R Programming on the RStudio site
   http://www.rstudio.com/resources/training/online-learning/#R

R core and R Packages

Go to http://www.r-project.org/ to download the appropriate core package for your operating system….and poke around to see there's a lot more on the site.



You can also download packages from CRAN (Comprehensive R Archive Network). As of 21 December 2014, there are 6129 packages listed on CRAN (http://cran.r-project.org/web/packages/)

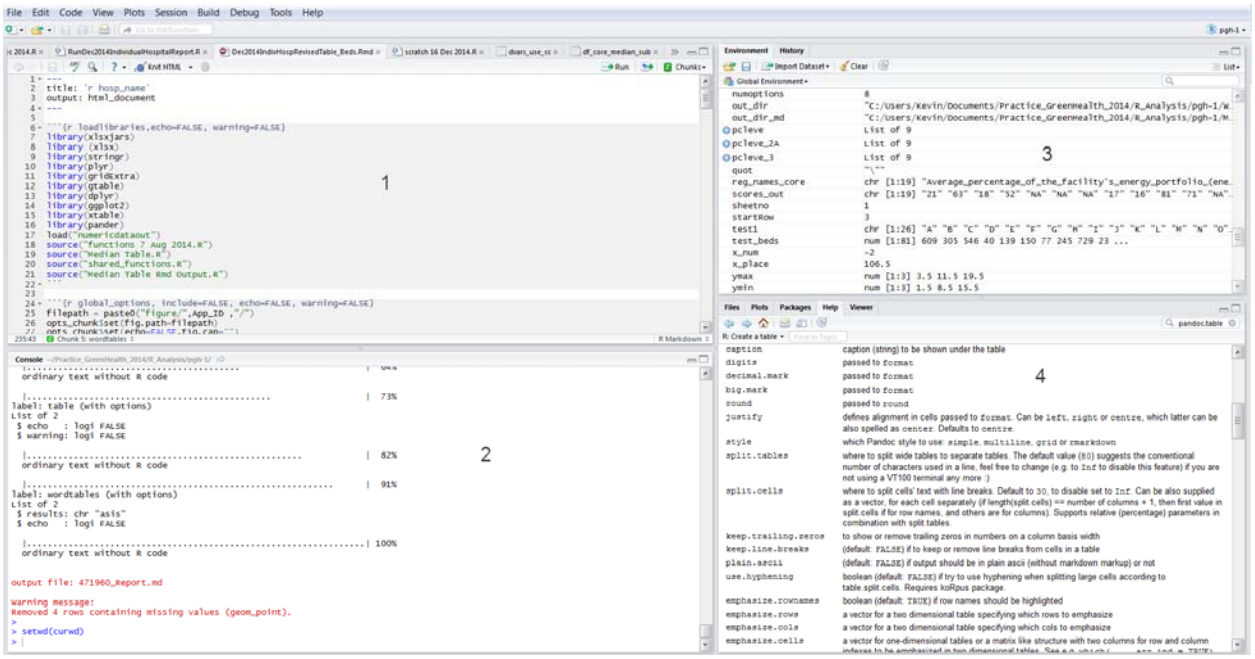
Packages are published sets of code and documentation that meet minimal standards of completeness and correct functioning. If you need to do a specialized analysis, chances are that somebody, somewhere, has written a package that will do some or all of what you are trying to do.

You find out about packages from your colleagues. What do they use? If you are working on a special analysis and you want to find a package, go to the R documentation database and use their search tools, http://www.rdocumentation.org/

As you read and learn more, you'll pick up suggestions for useful packages. For example, the package **ggplot2** is number one on the download list of all packages tracked by the R documentation site (1,100,000+ downloads as of 21 December 2014).

RStudio

RStudio is a company creating many useful R-connected tools. The first tool you should be using is the RStudio **i**ntegrated **d**evelopment **e**nvironment, also referred to as RStudio, which is a little bit confusing. The RStudio IDE helps you organize your R work in a very convenient way that promotes efficient work. Free from their website, www.rstudio.com. The IDE has four panels.



Panel 1 is where you create your R scripts (as well as more advanced types of R code).

Panel 2 is the R console, where numerical and text results appear when you run the scripts.

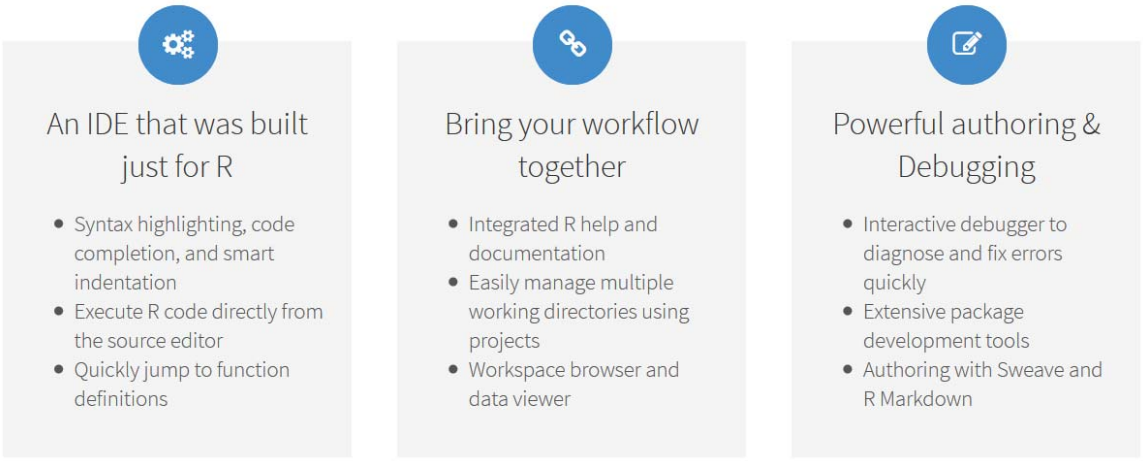
Panel 3 Shows the "environment", listing the data objects you've created, with summary information to help you keep track of details. You can toggle from the Environment to History to look back at what's been done if the console gets cleared.

Panel 4 has multiple tabs:

Files Window shows the files that are in your "working directory" (where R looks when you reference a data file, also the place R puts output by default)
Plots Window in which your graphs are drawn by default
Packages A list of the R packages you've downloaded and are avaialble for use (though you may need to **load** the package before you can actually use the functions in the package, you need the library function that is part of the core installation to do this.)
Help Search for R functions and wrestle with the sometimes opaque answers
Viewer A window for interactive graphs

Here's what the RStudio people say about their IDE:

RStudio is the premier integrated development environment for R. It is available in open source and commercial editions on the desktop (Windows, Mac, and Linux) and from a web browser to a Linux server running RStudio Server or RStudio Server Pro.



http://www.rstudio.com/products/rstudio/features/

Information about ggplot2, a very useful but idiosyncratic package for graphing

ggplot2 is a comprehensive graphical package that is my workhorse method for making graphs in R. Sometimes I use the "core" (base) graphic functions when I want a quick picture but I rarely use anything else than ggplot2 to make plots for clients these days. You can get publication quality output if you work hard enough.

There is a logic in the construction of ggplot2 ("a grammar of graphics") that helps you keep track of the commands as you build plots up, bit by bit.

1. A cheatsheet:

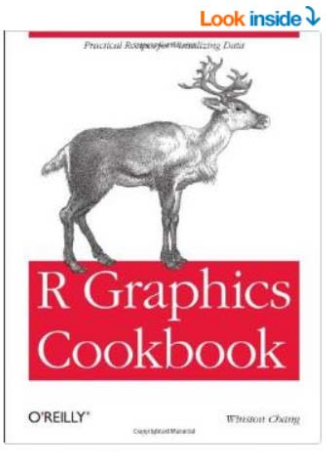
http://zevross.com/blog/2014/08/04/beautiful-plotting-in-r-a-ggplot2-cheatsheet-3/

2. Another set of quick tips when you can't remember how to to something: http://www.noamross.net/blog/2013/11/20/formatting-plots-for-pubs.html

3. A comprehensive (but not exhaustive) book: Winston Chang's *R Graphics Cookbook*.

Buy the Kindle edition and then you can cut and paste code into your R console!

http://www.amazon.com/R-Graphics-Cookbook-Winston-Chang/dp/1449316956



R Graphics Cookbook Paperback – January 6, 2013

by Winston Chang (Author)

42 customer reviews

ISBN-13: 978-1449316952 | ISBN-10: 1449316956 | Edition: 1st

Buy New

Price: $31.99

40 New from $23.53 | 16 Used from $29.19

	Rent from	Amazon Price	New from	Used from
Kindle	$7.95	$17.27	—	—
Paperback	—	$31.99	$23.53	$29.20

This practical guide provides more than 150 recipes to help you generate high-quality graphs quickly, without having to comb through all the details of R's graphing systems. Each recipe tackles a specific problem with a solution you can apply to your own project, and includes a discussion of how and why the recipe works.

Making Interactive Web Applications

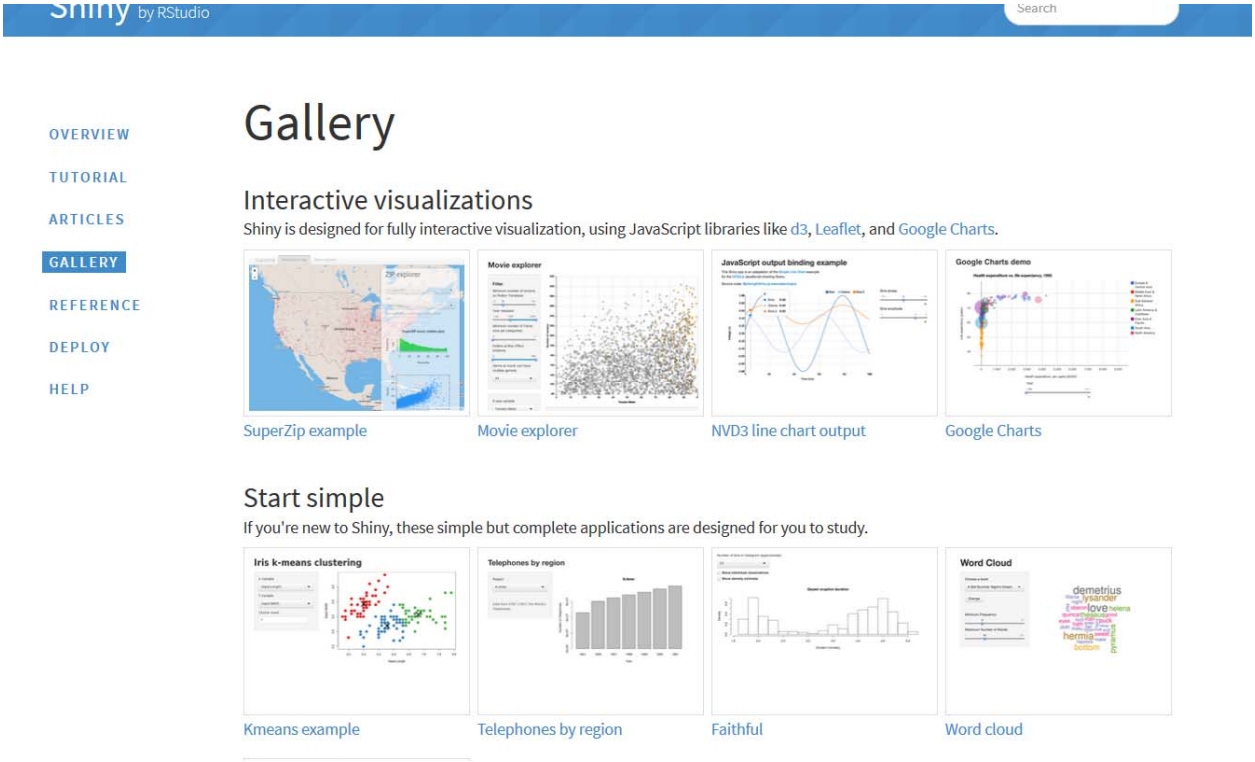
With some R skills in hand, you can build interactive web applications without knowing about web programming (or at least, not very much knowledge is required).

There is a whole world of possibilities to use Shiny, a web application framework for R, http://shiny.rstudio.com/

Shiny apps can have truly interactive graphics, including the Google Charts graphics tools from Google (GoogleVis is the R interface to Google Charts API) and beyond.

Check out the Gallery to make you want to build these apps!

http://shiny.rstudio.com/gallery/



Reproducible Research, R is not just the initial letter of the words

Biostatistics (2009), **10**, 3, *pp.* 405–408
doi:10.1093/biostatistics/kxp014
Advance Access publication on xxx xx, xxxx

Editorial

As coeditors of *Biostatistics*, we wish to encourage the practice of making research published in the journal reproducible by others. The following invited piece by Roger Peng sets out our policy on this; Roger will be assuming the role of Associate Editor for reproducibility as set out in his piece.

While we consider reproducibility to be a desirable goal, we wish to emphasise that our policy is to encourage our authors to consider this as an opportunity that they may wish to take, rather than as a requirement that we impose upon them. All submissions to the journal will continue to be reviewed using our established system; the issue of reproducibility will be considered only when a paper had been accepted for publication on the basis of its scientific merit as judged by our peer-review process.

PETER J. DIGGLE, SCOTT L. ZEGER

Reproducible research and *Biostatistics*

ROGER D. PENG

1. INTRODUCTION AND MOTIVATION

The replication of scientific findings using independent investigators, methods, data, equipment, and protocols has long been, and will continue to be, the standard by which scientific claims are evaluated. However, in many fields of study there are examples of scientific investigations that cannot be fully replicated because of a lack of time or resources. In such a situation, there is a need for a minimum standard that can fill the void between full replication and nothing. One candidate for this minimum standard is "reproducible research", which requires that data sets and computer code be made available to others for verifying published results and conducting alternative analyses.

http://biostatistics.oxfordjournals.org/content/10/3/405.full.pdf+html

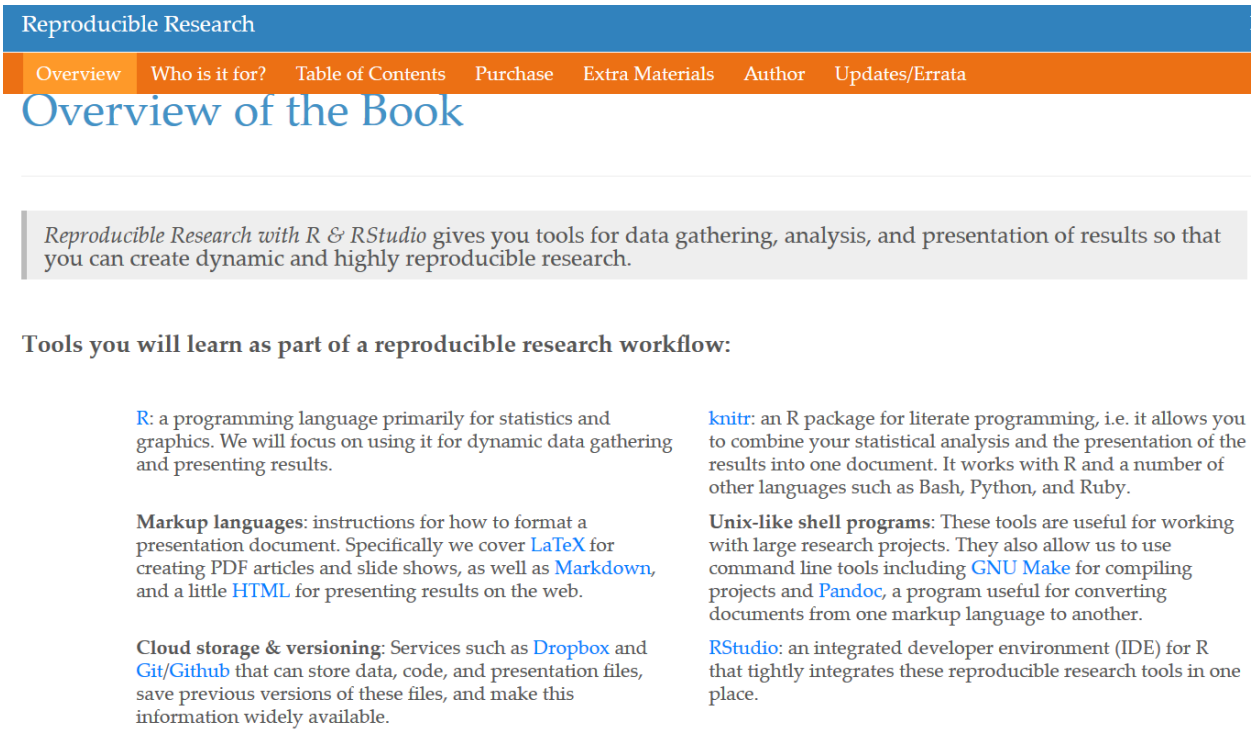
R, implemented in the RStudio environment, has the tools to create reproductible research.

You can document your code and demonstrate that others can get the same answers—summary descriptions, graphs, and model fits—as you did when you first created the analysis.

There are many benefits from changing your daily work to take advantage of these tools. At the very least, you will avoid the panic and rework when you have to go back to an analysis

from weeks, months or years earlier and are unable to get the same results because somewhere you changed something or forgot a step.

Here are the relevant tools, described by Christopher Gandrud in the overview of his 2014 book *Reproducible Research with R and RStudio* (CRC Press, http://christophergandrud.github.io/RepResR-RStudio/):



There is plenty of information on the web about how to make these tools work together. Since you are or should be using RStudio, start here: http://rmarkdown.rstudio.com/

Learn about new tricks and developments

1. Find buddies who will work with you on your projects, they can point out tips and traps.
2. Investigate on-line courses e.g. at DataCamp https://www.datacamp.com/ or Coursera https://www.coursera.org/specialization/jhudatascience/1?utm_medium=catalog
3. Keep up with the crowd http://www.r-bloggers.com/ subscribe to email summary for daily tips and links.

Keeping Your version of R and many packages updated: Important trick for Windows

Use installr package and run installR function (>installR()) in the R console. R will ask if you want to copy over packages and check for updates. Very handy to keep all those packages up to date!

Contact: Kevin Little, Ph.D., klittle@iecodesign.com 608-251-4355