Introduction to Ecological Data Analysis with R

Workshop Overview

Dates and Times: Wednesday (Feb 25th) and Thursday (26th) 8:00am-5:00pm and Friday (27th) 8:00am-2:00pm. The first two days (see schedule of topics below) will build upon each other, and provide an introduction to the fundamentals of R. My intent is to have each session consist of a presentation, followed by at least one "hands-on" experience using real world examples. For example, the presentation in the session on "Data import & export" would cover various text file formats (e.g., .csv, .dat, .txt), spreadsheets, relational databases (e.g., Access), and simple geospatial data (GIS files), with simple examples. The second half of that session would include examples of loading data directly from Excel files using the xlsx package, pulling tables & stored queries from Access using the RODBC package, and importing vector & raster geospatial data (e.g., ESRI .shp and ERDAS .img files) using the rgdal and raster packages.

Each day will end with a one-hour block of time for you to practice the concepts and techniques learned using your own data (example datasets can be provided). In addition, a couple of assigned readings will be provided at the end of the first two sessions that will provide useful context for the subjects to be covered the next day.

Why R?

R is an open-source implementation of the S language for statistical computing. For over 20 years, applied statisticians have been submitting implementations of their new techniques to StatLib. When most of those implementations were written as libraries for the commercial S-plus implementation of the S language, statisticians were providing software for free, but users (including those same statisticians) had to pay a third party (Insightful Corp.) to be able to run the software. A very small group of statisticians took it upon themselves to write a complete open-source implementation of S that would run under most operating systems, which they called R. Since then, the vast majority of implementations of new statistical techniques have been made available as R packages, which include the code as a library of functions and at least some documentation. In fairness to S-Plus, while there are several GUI interfaces available for R, S-Plus provides a much more polished and complete GUI interface and user experience.

Because R is very useful for "computing with data", experts in many fields use it for their work. Because R is open source, many of those experts make their field-specific code and functions freely available as packages (currently over 6000 packages, see http://cran.cnr.berkeley.edu/web/packages/). For example, climate researchers use R with netCDF files, so there are packages for reading and writing netCDF files (netCDF, ncdf4) as well as for generating standard climate diagrams, imputing missing weather data, downscaling from coarse data, etc. (climtol, clim.pact, seas, anm, zyp). Phenology researchers provide packages bise and pheno as well as a package for pulling data directly from the National Phenology Network. Jari Oksanen (with help from others) provides package vegan for vegetation analysis (ordination, classification, analysis of similarity, and much more). There are several packages for species richness, diversity, rarefaction, etc.. Wildlife biologists provide several packages for estimating occupancy & abundance from various forms of data: unmarked, mra, Rcapture, secr, PresenceAbsence. The key point is that by learning how to use R, at least to the level of writing code to reshape our data into the required structures and call the provided functions, we can leverage their efforts and expertise, and not reinvent those wheels. In order to produce informative and valid results we still have to understand the topic (e.g., dendrochronology or wildlife population assessment), but we do not need to translate the approaches and equations in the literature into computer code, as experts have done that, tested it, and (to varying degrees) documented it.

Why Clicking Buttons is Overrated!

There are two major reasons you may want to learn to write R code rather than using a Graphical User Interface (GUI). First, while more and more of the general statistical methods are being added to GUI-based interfaces, like Rcmdr, almost all the field-specific packages require R code to use. Packages are sets of one or more functions useful for a set of tasks. The advantage of functions in R is that we don't need to understand or modify anything inside the function in order to use that package (although the source code is available if we need to inspect it or improve it). We only need to know what parameters we need to pass to the function, and how to use the objects (figures, analysis results, or data objects) it returns. Therefore, the amount of coding required of the user is quite limited: mostly creating the data objects the functions require, then calling the functions in the desired order.

Second, scripts document the analysis and workflow in an unambiguous manner, and make the work reproducible. Most scientific work in ecology involves decisions about outliers and missing values, and many options during the statistical analysis, far too many decisions and options to be documented in a standard methods section of a paper. In addition, the methods can also be difficult to rerun later when editors and reviewers want one “simple” change, or a colleague needs help to perform a similar analysis. Because these details can greatly affect the results, some ecological journals and data scientists now encourage (or require) some form of documentation or journaling of the entire scientific workflow. R code (or SAS code or SPSS code) that includes the querying of the database, merging and cleansing data, generating the figures and tables, and performing the analyses themselves are one way to meet that requirement.

Other Learning Options

We learn in very different ways, and we have a wide range of backgrounds. Even if it goes well, this workshop will not be the most efficient way for some of us to learn to use R in our work (it likely would not have for me). You may do better simply working through the slides and examples associated with this workshop at your own (likely faster) pace. CRAN has a list of user contributed documentation (http://cran.r-project.org/other-docs.html), including several substantial books for learning R. The main R-project website has a broader list of resources at http://www.r-project.org/other-docs.html. Coursera has offered several massive, open, online courses (MOOCs) on data analysis with R. There are a number of dead-tree books that provide an introduction to R. Past versions of this workshop have used Crawley’s “The R Book" or Kabacoff’s "R in Action: Data Analysis and Graphics with R." Any of these resources will help you learn the basics of R.

Pre-course Installation

We will start on February 25th with the expectation that everyone has R and RStudio installed and running on their computer. I recommend that you either install R and RStudio yourself, following the directions below, or let your IT folks do the install if you don't have administrative rights or are not comfortable installing software. My recommendations for a quick installation are as follows:

1. Download R from http://cran.us.r-project.org/ (click on “Download R for Windows” > “base” > “Download R 3.1.2 for Windows”). Install R. Leave all default settings in the installation options. If you are running a 64-bit version of Windows, this version will install both 32-bit and 64-bit R, which is what you want.
2. When prompted, install R in c:/R/R-3.1.2 not c:/Program Files/R/... This is especially important if you don't have administrative rights on your computer and don't have permissions to install software to the C:/Program Files/ folder. You may be downloading additional packages, and they will need to write files under your R directory.
3. Download RStudio from http://rstudio.org/download/desktop (click on “RStudio 0.98.1102 - Windows XP/Vista/7/8”) and install it. Just select the default settings in the installation.
4. If you don't have a favorite ASCII/text/programming editor, or you prefer a slightly more advanced text editor, I recommend Notepad++ - http://notepad-plus-plus.org/.
5. Lastly, we’ll be using numerous packages in R which can be downloaded prior to the beginning of the workshop or as needed over a wi-fi connection. However, don’t worry about this now as we’ll be proving the necessary packages (and their dependencies) on a thumbdrive to ensure that we don’t “break” the internet at the facility.

Session Topics

This schedule is tentative, and subject to change if the workshop is going to fast or too slow for the majority of the participants.

Day	Time	Topic
Wednesday, Feb 25th	8:00-8:30am	Introductions and "Why R?"
		Getting started
	8:30-9:00	Installation & configuration
	9:00-9:30	Jumping in with both feet - a familiar example
	9:30-9:40	Break
	9:40-10:00	R's online community & packages
		R fundamentals
	10:00-10:20	Data structures and objects
	10:20-11:00	Vectors, matrices array, data frames, lists & factors
	11:00-11:10	Break
	11:10-11:30	Data import & export
		Data management
	11:30-12:00pm	Missing values, factors, dates and time, & manipulating character variables
	12:00-1:00	Lunch
	1:00-1:30	Subsetting, reshaping and merging dataframes
	1:30-2:00	Controls and user-defined functions
		Basic topics
	2:00-2:30	Data exploration
	2:30-2:40	Break
	2:40-3:00	Descriptive statistics, frequency & contingency tables
	3:00-4:00	Graphics
	4:00-5:00	Begin working with your own data

Day	Time	Topic
Thursday, Feb 26th	8:00-8:30am	Differences, association, and correlation
	8:30-9:00	Example session
	9:00-9:20	Linear models
	9:20-9:30	Break
	9:30-10:00	Example session
		Intermediate topics
	10:00-10:30	Maximum Likelihood
	10:30-11:00	Example session
	11:00-11:10	Break
	11:10-11:20	Model Selection
	11:20-11:30	Example session
	11:30-12:00pm	Analysis of Variance
	12:00-1:00	Lunch
	1:00-1:30	Example session
	1:30-2:00	Generalized linear models
	2:00-2:20	Example session
	2:20-3:00	Break
		Advanced topics
	2:30-3:00	ggplot2 and lattice graphics
	3:00-3:30	Example session
	3:30-4:00	Overview of other useful packages
	4:00-5:00	Continue working with your own data
Friday, Feb 27th	8:00-8:30am	Overview of the unmarked package
	8:30-9:00	Single season occupancy
	9:00-9:30	Example session
	9:30-10:00	Multi-season occupancy
	10:00-10:10	Break
	10:10-10:30	Example session
	10:30-11:00	Species co-occurrence
	11:00-11:20	Abundance
	11:30-12:00pm	Example session
	12:00-12:15	Wrap-up
	12:15-1:15	Lunch/Adjourn
	1:15-2:00	Continue working with your own data