Basics of R

Data import and manipulation, graphs, functions, modeling

R. Condit
Ohio State University Department of Music
May 2013

1 Course overview

The course will start with basic R, covering how to import and manipulate data. We will subsequently cover three main topics: graphs, functions, and basic regression models.

During the first half of each session, I will explain methods and present examples of their use; in the second half, students will work on their own using the same methods. Datasets will be provided, but students are encouraged to bring their own data as well. A course web site will provide sample code, data, and a list of key *R* functions. Students should have their own computer to work on during the sessions, know the file system on it (ie, how to find files by typing the path plus file name), and be familiar with ascii text files.

1.1 To join

- Contact Nat Condit-Schultz

1.2 Schedule

- When: Four Sessions, 10AM-2PM (with lunch break), 7-10 May 2013
- Where: Huron Lab, OSU Music

2 Software required

- *R* base package `http://cran.us.r-project.org/`
- a programming editor such as Geany (NOT Wordpad NOT Notepad)

3 Course web site

- `http://richardcondit/workshops/OSU`

  `http://www.richardcondit.org/workshops/OSU/outline.html`, and outline.pdf
  `http://www.richardcondit.org/workshops/OSU/assignments.html`, and assignments.pdf
  sample R and csv datasets `http://www.richardcondit.org/workshops/data`
  R scripts `http://www.richardcondit.org/workshops/source`
  history or R commands I enter during the course `http://www.richardcondit.org/workshops/OSU/history`

- Each will be updated regularly throughout the course

4 Contents and approximate scheduling

1. **Basics of R** [day 1]
   - Data types

     Atomic (or scalar): a single value
     Vector: one dimensional array of values
     Dataframe: two dimensional table
     Advanced: matrix, array, list
   - Character vs. numeric variables
   - Command line assigning and manipulating
   - Import

     Reading ascii
     Loading: load vs. attach
   - Export

     Saving
     Writing
   - Working with dataframes

     Array elements
     The $ symbol
     Extracting one or a few rows (columns)
     Subsetting rows
     Adding a new column
   - Repeatability

     Scripts
     The source command
     Functions (see day 4)
2. **Graphs** [start day 1, continue day 2]
   - Scatter plot

     The functions plot and points
     Data on tempo and temperature (studyDataOneClean.csv): Tempo Vs. AverageHigh
   - Lines
   - Manipulating the appearance

     Arguments to plot: xlab, ylab, pch, col, lty
     Log-transformation
     Command-line export
     Advanced: manipulating axes with axis and box
3. **Power of R (a brief introduction of things spreadsheets can't accomplish)** [day 2]
   - Vectorized calculations
   - Filtering data to get one or more rows (columns)

   - colSums and rowSums
   - table
   - tapply
   - subset

4. **Modeling with standard regression** [day 3]
   - Linear regression

     The function lm
     Regression line
     Capturing the results
     Data treemass: log(agb) vs. log(dbh)

5. **Creating your own functions** [day 4]
   - Function definition and the curly braces
   - Understanding arguments

     Local names and values
     Default values
   - Loops

     for
     while
     Requires curly braces
   - Using if and else
   - Subroutines

     Functions within functions
     Adds work now to save work later
   - Returns value

     Single variable
     Multiple variable returns with list data type

6. Advanced topic: **Multi-level models** [if time and interest are available]
   - Why multi-level modeling?
   - Multi-level vs. standard regression
     Bates Chap 4, Section 4.4; Gelman & Hill pp. 251-259
   - Regression with one group using lmer
     - output of display
     - graphs using the coefficients
     - variable intercept, slope, or both
   - Random and fixed effects
   - Good references
     - Gelman: `http://www.amazon.com/Analysis-Regression-Multilevel-Hierarchica` `dp/052168689X`
     - Bates' online book `http://lme4.r-forge.r-project.org/`

5 Key *R* functions

- Basics
  1. length
  2. dim
  3. str
  4. read.table
  5. write.table
  6. attach
  7. load
  8. colnames
- Data creation
  1. read.table (also read.delim, read.csv)
  2. write.table
  3. numeric
  4. c
  5. : (a colon)
  6. character
  7. data.frame
  8. matrix
  9. array
- Numeric
  1. math operators: * + - : / ^
  2. math functions: log, log10, sin, cos, sqrt (and more)
  3. mean
  4. median
  5. sd
  6. var
  7. summary
- Character
  1. table
  2. unique
  3. nchar
- Graphics
  1. hist
  2. plot
  3. points
  4. line
  5. curve
  6. abline
  7. box

8. axis
9. X11
10. dev.set

- Functions
  1. function
  2. args
  3. loops
  4. if, else
  5. return
  6. browser
- Data extraction
  1. subset
  2. apply
  3. tapply
  4. cut
  5. dim
- Modeling
  1. model
     lm
     - return of lm: coefficients, fitted.values, residuals
     - result of summary(lm): also r.squared
     - abline(lm) to add line
     - predict.lm for predictions
  2. details
     - lm relies on normally distributed data, especially for y
       * logarithms when data are skewed
       * for small integers or binomial data (0 vs. 1) there is an alternative, glm
     - when x is numeric, it is helpful to 'center' by substracting mean (especially if x is very far from zero)
  3. advanced
     glm
     lmer [lme4 package]
     coef
     summary
     fixef [arm package]
     ranef [arm package]
     display [arm package]
     dotplot [lattice package]
     xyplot [lattice package]