EPSY 887: Computation Statistics

Introductions

Jason M. Bryer

http://github.com/jbryer/CompStats
jason@bryer.org

Week 1
January 28, 2013

Agenda

1. Introductions
2. Course Overview
3. Software
4. Data
5. Introduction to R

Outline

Introductions

- Your name and department.
- Experience with R.
- Research interests.
- What experiences have you had with larger datasets.

Outline

Why learn to program?

- *Independence*: otherwise, you rely on someone else always having made exactly the right tool for you, and giving it to you.

(Shalizi, 2012, `http://www.stat.cmu.edu/~cshalizi/statcomp/`)

Why learn to program?

- *Independence*: otherwise, you rely on someone else always having made exactly the right tool for you, and giving it to you.
- *Honesty*: otherwise, you end up distorting the problem to match the tools you happen to have.

(Shalizi, 2012, `http://www.stat.cmu.edu/~cshalizi/statcomp/`)

Why learn to program?

- *Independence*: otherwise, you rely on someone else always having made exactly the right tool for you, and giving it to you.
- *Honesty*: otherwise, you end up distorting the problem to match the tools you happen to have.
- *Clarity*: turning your method into something a machine can do forces you to discipline your thinking and make it communicable.

(Shalizi, 2012, `http://www.stat.cmu.edu/~cshalizi/statcomp/`)

Reproducible Research

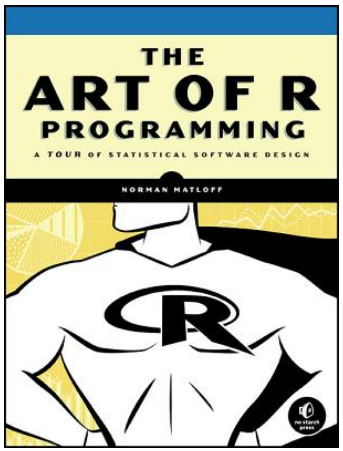
Let us change our traditional attitude to the construction of programs: Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to humans what we want the computer to do.

Knuth, D. E. (1984). Literate Programming. *The Computer Journal, 27*(2):97-111

Textbooks

- Kabacoff, R.J. (2011). *R in Action: Data Analysis and Graphics with R*. Shelter Island, NY: Manning.

- Matloff, N. (2011). *The Art of R Programming*. San Francisco, CA: No Starch Press.



Topics to Cover During the Semester

The following list of topics is subject to change based upon your interest.

- Using R as a programming language including control statements (including `if`, `for`, `while` statements) and functional programming.
- Advanced data visualizations.
- Object oriented programming (i.e. S3, S4, and Reference Classes)
- Missing data
- Analysis of complex survey designs
- LaTeX
- R package development

Topics to Cover During the Semester

The following list of topics is subject to change based upon your interest.

- Using R as a programming language including control statements (including `if`, `for`, `while` statements) and functional programming.
- Advanced data visualizations.
- Object oriented programming (i.e. S3, S4, and Reference Classes)
- Missing data
- Analysis of complex survey designs
- LaTeX
- R package development

Other statistical topics including:

- Propensity score analysis
- Multilevel modeling (HLM)
- Item response theory (IRT)

Outline

Software

We will utilize the following software:

- R (`www.r-project.org`)
- RStudio (`www.rstudio.com`)
- LaTeX (MacTeX or MiKTeX)
- Git vis-à-vis `www.github.com`
  - Github for Windows `http://windows.github.com/`
  - Github for Mac `http://mac.github.com/`
  - Tower for Mac `http://www.git-tower.com/` [1]

[1] Students get 50% off

Installing R

The latest version of R can be obtained from `http://cran.r-project.org`.
The current version of R is:

```
> R.version$version.string
```

```
[1] "R version 2.15.2 (2012-10-26)"
```

Installing R

The latest version of R can be obtained from `http://cran.r-project.org`.
The current version of R is:

```
> R.version$version.string
```

```
[1] "R version 2.15.2 (2012-10-26)"
```

For Windows the following should also be installed:

- RTools `http://www.murdoch-sutherland.com/Rtools/`
- ActivePerl

For Mac the following should also be installed which are available from `http://cran.r-project.org/bin/macosx/tools`

- gfortran-4.2.3
- tcl/tk 8.5.5

Detailed installation instructions are on the course website:
`https://github.com/jbryer/CompStats/blob/master/Installation/`

Outline

Data Sources

- Programme for International Student Assessment (PISA)
- DataFerrett (Federated Electronic Research, Review, Extraction, and Tabulation Tool)
- The World Bank
- Trends in International Mathematics and Science Study (TIMSS)
- Progress in International Reading Literacy Study (PIRLS)
- California Department of Education
- School Attendance Boundary Information System (SABINS)
- American Community Survey
- Integrated Postsecondary Education Data System (IPEDS) Data about higher education institutions.
- Google Public Data
- Zanran A search engine for data and statistics.
- The Washington Post The Washington Post has compiled a list of some data sources.
- Inter-university Consortium for Political and Social Research

Outline

What is 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...

What is 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 provides a wide variety of statistical (linear and non linear modeling, classical statistical tests, time-series analysis, classifcation, 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. (R-project.org)

R's Roots... S

- S is a language that was developed by John Chambers and others at Bell Labs.
- S was initiated in 1976 as an internal statistical analysis environment - originally implemented as Fortran libraries.
- Early versions of the language did not contain functions for statistical modeling.
- In 1988 the system was rewritten in C and began to resemble the system that we have today (this was Version 3 of the language). The book Statistical Models in S by Chambers and Hastie (the blue book) documents the statistical analysis functionality.
- Version 4 of the S language was released in 1998 and is the version we use today. The book Programming with Data by John Chambers (the green book) documents this version of the language.

History of S

- In 1993 Bell Labs gave StatSci (now Insightful Corp.) an exclusive license to develop and sell the S language.
- In 2004 Insightful purchased the S language from Lucent for $2 million and is the current owner.
- In 2006, Alcatel purchased Lucent Technologies and is now called Alcatel-Lucent.
- Insightful sells its implementation of the S language under the product name S-PLUS and has built a number of fancy features (GUIs, mostly) on top of it-hence the "PLUS".
- In 2008 Insightful is acquired by TIBCO for $25 million; future of S-PLUS is uncertain.
- The S language itself has not changed dramatically since 1998.
- In 1998, S won the Association for Computing Machinery's Software System Award.

In "Stages in the Evolution of S", John Chambers writes:

> *"[W]e wanted users to be able to begin in an interactive environment, where they did not consciously think of themselves as programming. Then as their needs became clearer and their sophistication increased, they should be able to slide gradually into programming, when the language and system aspects would become more important."*

http://www.stat.bell-labs.com/S/history.html

History of R

- 1991: Created in New Zealand by Ross Ihaka and Robert Gentleman. Their experience developing R is documented in a 1996 JCGS paper.
- 1993: First announcement of R to the public.
- 1995: Martin Machler convinces Ross and Robert to use the GNU General Public License to make R free software.
- 1996: A public mailing list is created (R-help and R-devel)
- 1997: The R Core Group is formed (containing some people associated with S-PLUS). The core group controls the source code for R.
- 2000: R version 1.0.0 is released.
- 2012: R version 2.15.2 is released on October 31, 2012.
- There are now over 4,000 packages listed on CRAN.

R as a Big Calculator

```
> 2 + 2
[1] 4
```

R as a Big Calculator

```
> 2 + 2
[1] 4
> 1 + sin(9)
[1] 1.4
```

R as a Big Calculator

```
> 2 + 2
[1] 4
> 1 + sin(9)
[1] 1.4
> 23.76 * log(8)/(23+atan(9))
[1] 2
```

Installing Packages

Both Windows and Mac have a menu system for installing packages, however the `install.packages` function allows for the installation to be scriptable.

```
> install.packages(c("psych", "gdata", "foreign", "devtools",
    "roxygen"), dependencies = TRUE)
```

See the `Setup.r` script for more details, including some special details for installing some packages on Windows (e.g. `XML` package).

We will also install packages that are not yet on CRAN but are on Github. The `install_github` function in the `devtools` package allows us to install those packages:

```
> require(devtools)
> install_github("pisa","jbryer")
```

Loading Packages

The `require` command[2] will load a package into the current R session.

```
> require(psych)
> require(gdata)
> require(foreign)
```

[2] The `library` command will also load packages but the `require` is now preferred.

Loading Packages

The `require` command[2] will load a package into the current R session.

```
> require(psych)
> require(gdata)
> require(foreign)
```

For a list of packages that have been downloaded, but not necessarily attached, the `library()` function without any parameters will return that list.

```
> library()
```

[2] The `library` command will also load packages but the `require` is now preferred.

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites
- The `help.search()` function will search the help file for a particular word or phrase. For example:

```
> help.search("cross tabs")
```

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites
- The `help.search()` function will search the help file for a particular word or phrase. For example:

  ```
  > help.search("cross tabs")
  ```

- To get documentation on a specific function, the `help()` function, or simply `?functionName` will open the documentation page in the web browser.

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites
- The `help.search()` function will search the help file for a particular word or phrase. For example:

  ```
  > help.search("cross tabs")
  ```

- To get documentation on a specific function, the `help()` function, or simply `?functionName` will open the documentation page in the web browser.
- Lastly, to search the R mailing lists, use the `RSiteSearch()` function.

NA vs. NULL

R is just as much a programming language as it is a statistical software package. As such it represents null differently for programming (using NULL) than for data (using NA).

NA vs. NULL

R is just as much a programming language as it is a statistical software package. As such it represents null differently for programming (using NULL) than for data (using NA).

> *NULL represents the null object in R: it is a reserved word. NULL is often returned by expressions and functions whose values are undefined.*

NA vs. NULL

R is just as much a programming language as it is a statistical software package. As such it represents null differently for programming (using NULL) than for data (using NA).

> *NULL represents the null object in R: it is a reserved word. NULL is often returned by expressions and functions whose values are undefined.*
>
> *NA is a logical constant of length 1 which contains a missing value indicator. NA can be freely coerced to any other vector type except raw. There are also constants NA_integer_, NA_real_, NA_complex_and NA_character_of the other atomic vector types which support missing values: all of these are reserved words in the R language.*

For more details, see `http://opendatagroup.com/2010/04/25/r-na-v-null/`

Atomic Vectors

R has six atomic vectors, they are:

- `character`
- `numeric`
- `integer`
- `logical`
- `complex`
- `raw`

Atomic Vectors

R has six atomic vectors, they are:

- `character`
- `numeric`
- `integer`
- `logical`
- `complex`
- `raw`

Methods useful for working with vectors:

`c`	Concatenate (i.e. combine values into a vector or list)
`str`	Provides the structure of any R object (perhaps the most useful function in R!)
`names`	Returns the names of an object
`dim`	Dimensions of the object
`dimnames`	Name of rows and columns of a matrix
`class`	Returns the class, or type, of an object

Lists

Lists are generic vectors where each element can be any R object, including other Lists!

```
> mylist <- list(letters=letters, numbers=1:10)
> class(mylist)
```

```
[1] "list"
```

```
> str(mylist)
```

```
List of 2
 $ letters: chr [1:26] "a" "b" "c" "d" ...
 $ numbers: int [1:10] 1 2 3 4 5 6 7 8 9 10
```

```
> length(mylist)
```

```
[1] 2
```

Subsetting Lists and Vectors

```
> mylist[1]

$letters
 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k"
[12] "l" "m" "n" "o" "p" "q" "r" "s" "t" "u" "v"
[23] "w" "x" "y" "z"

> mylist[[1]]

 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k"
[12] "l" "m" "n" "o" "p" "q" "r" "s" "t" "u" "v"
[23] "w" "x" "y" "z"

> mylist$letters

 [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k"
[12] "l" "m" "n" "o" "p" "q" "r" "s" "t" "u" "v"
[23] "w" "x" "y" "z"

> mylist$numbers

 [1]  1  2  3  4  5  6  7  8  9 10
```

Reading Data

read.table	Reads in a table where each line is a record. Lots of options to define the structure of the file.
read.csv	Comma deliminited files.
read.spss	In the `foreign` package, reads SPSS files.
read.xls	In the `gdata` package, reads Excel files.
RODBC	This package has functions to read data from most ODBC databases.
RMySQL	Package for reading data from MySQL databases.
RPoostgreSQL	Package for reading data from PostgreSQL databases.
load	Read in R data object files saved using the `save`. This is very useful for saving intermediate data files.

Descriptive Statistics

table	Crosstabs.
summary	Provides summary information relevant to the type.
describe	In the `psych`, provides many of the most common descriptives statistics (e.g. mean, median, standard deviation, range, etc.)
describeBy	Same as `describe` but will provide descriptive stats based upon grouping varaible(s).
fivenum	Returns Tukey's five number summary (minimum, lower-hinge, median, upper-hinge, maximum)
mean	Mean
median	Median
sd	Standard deviation
var	Variance