Statistical Programming & R Basics

Yihui Xie

School of Statistics, Renmin University of China

Oct. 21, 2006

Outline

1. The Meaning of Statistical Programming
2. Fundamentals for Programming
3. Whats R Language?
4. Syntax for R
5. Basic Applications of R in Statistics
6. R Graphics
7. Learn to learn!

1. The Meaning of Statistical Programming

- Why do we need statistical programming?
  - To understand the *essential* of statistical models, methods & algorithms; make clear those "*black boxes*"
  - Flexible control of statistical outputs
  - Conveniently develop corresponding programs (or packages) when *new* methods were born
- Programmings not everybodys business
  - if (you completely trust those existing stat. packages) {stop programming}
  - if (the outputs of current stat. software are enough for you) {stop programming}
  - if (you only use classical stat. models) {stop programming}

2. Fundamentals for Programming

- Differences between computers and human beings
  - Computers cannot comprehend *the significance of life* through a film named "LE PEUPLE MIGRATEUR"
  - Human beings cannot calculate the product 1*2*...*1000000 *by hand*
- Basic knowledge for a program
  - Algorithm (the soul of a program)
  - Data structure
  - Environment

2.1 Basic Program Control

- Two important control flow constructs
  - *Conditional execution*: if statements
    - if (cond) expr
    - if (cond) cons.expr else alt.expr
  - *Repetitive execution*: for loops, repeat and while
    - for (var in seq) expr
    - while (cond) expr

```
> a=1
> if (a==1) {print("Hello Yihui")}
[1] "Hello Yihui“
> a=0
> for (i in 1:10) {a=a+i}
> a
[1] 55
```

2.2 Data Structure

- Data types
  - Integer, String, Double, …
  - Array, (Vector, Matrix, …)
- Operators
  - +, -, *, /, …
  - <, >, ==, !=, &, |, …

2.3 Environments

- C/C++, Basic, Java, Fortran, ...
- .NET, ...
- R, S-Plus, SPSS, SAS, ...
- Different environments have different rules, e.g. *case sensitivity*, etc
  - There's no case sensitivity in Visual Basic, but in C and R this is not true, i.e. the variable *X* is NOT the same as *x*!

3. Whats R Language?

- When we start R, we may see
    - R version 2.4.0 (2006-10-03)
Copyright (C) 2006 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
    - R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type license() or licence() for distribution details.
    - R is a collaborative project with many contributors.
Type contributors() for more information and citation() on how to cite R or R packages in publications.

3.1 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 Foundations 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.

3.2 Contributors of R

- The current R is the result of a collaborative effort with contributions from all over the world. R was initially written by Robert Gentleman and Ross Ihaka—also known as "R & R" of the Statistics Department of the University of Auckland.
  - Douglas Bates
  - John Chambers
  - Peter Dalgaard
  - Robert Gentleman
  - Kurt Hornik
  - Stefano Iacus
  - Ross Ihaka
  - Friedrich Leisch
  - Thomas Lumley
  - Martin Maechler
  - Duncan Murdoch
  - Paul Murrell
  - Martyn Plummer
  - Brian Ripley
  - Duncan Temple Lang
  - Luke Tierney
  - Simon Urbanek

Ross Ihaka

Robert Gentleman

3.3 How to Get It?

- DO bear in mind this web site:
  - http://www.r-project.org
  - You'll find it a great help when you intend to study R language!
- CRAN – Comprehensive R Archive Network
  - http://cran.r-project.org/
  - Choose a mirror and download the install packages for your OS
- Base System
- Other Contributed Packages
  - quantreg, foreign, RODBC, cluster, …

3.4 Attention before Using R

- OMG, where has the lovely GUI gone? ...
- Perhaps youll get disappointed when you could find no menus for statistical analysis
- Before you get started, please be prepared for those "ugly" source codes...
- But "ugly" codes will pay you back with (probable) beautiful outputs including excellent graphics, etc

4. Syntax for R

- Data Structures
  - Modes and Types
    - "logical", "integer", "double", "complex", "raw", "character", "list", "expression", "name", "symbol" and "function"
    - Please check the help files to find the trivial difference between "mode" and "type"
  - Classes
    - "numeric", "logical", "character" or "list", but "matrix", "array", "factor" and "data.frame" are other possible values
- Operators
  - +, -, *, /, ^, %%, %/%, %*%

4. Syntax for R (cont.)

- Slicing and extracting data
  - Indexing lists
    - x[[n]] nth element of the list
    - x[["name"]] element of the list named "name"
    - x$name id.
  - Indexing vectors
    - x[n] nth element
    - x[-n] all but the nth element
    - x[1:n] first n elements
    - x[c(1,4,2)] specific elements
    - x["name"] element named "name"
    - x[x > 3] all elements greater than 3
  - Indexing matrices
    - x[i,j] element at row i, column j
    - x[i,] row i
    - x[,j] column j
    - x["name",] row named "name"
  - Indexing data frames (matrix indexing plus the following)
    - x[["name"]] column named "name"
    - x$name id.

4. Syntax for R (cont.)

- Some functions (might be useful)
  - c(), seq(), rep(), ... (please note the use of ":")
  - max(), min(), mean(), sd(), var(), cor(), sum(), median(), quantile(), summary(), ...
  - sort(), t(), solve(), eigen(), svd(), chol(), qr(), ...
  - pnorm(), dnorm(), qnorm(), rnorm(), ... (p, d, q, r-distribution function)
  - lm(), anova(), t.test(), ...
  - plot(), lines(), boxplot(), hist(), pairs(), stem(), pie()...
  - print(), read.table(), write.table(), ...

4. Syntax for R (cont.)

- Special hints on the use of brackets
  - () usually for parameters of functions; e.g. *mean(x)*
  - [] for indexing; e.g. *x[200]*
  - {} for an integrated statement; if you're not sure whether you need them, just use them. Lack of them might cause an error, but *redundant* use will never bring you errors.
    - if (a==3) {print("braces are good")}
    - if (a==3) print("braces are good")
  - [[]] for extracting data from list and data frames (refer to page 14)
- About the comments
  - #

5. Basic Applications of R in Statistics

- Descriptive Statistics
  - In most cases, the function *summary()* will be enough for text results, and *hist()* for graphical output. Besides, you may use special functions such as *skewness(e1071)* and *kurtosis(e1071)* to meet your special needs.
- Inferential Statistics
  - The most basic model in statistics – linear model: *lm()*
    - lm(y~x)
  - Other models
    - ANOVA, GLM, Mixed models, Robust regression, Tree-based models, Additive models (GAM), and several TS models, etc
  - Hypothesis test
    - t.test(), ks.test(), wilcox.test(), adf.test(), …

6. R Graphics

- plot() -- A generic function for plotting of R objects
  - plot(rnorm(300)): plot 300 random numbers (normally distributed) against their indices
  - plot(x, y): a scatter plot of x and y
  - plot(lm(y~x)): a sequence of plots will be generated
  - plot(a): if a is a data frame, a "matrix" of plots will be generated
- Other graphics
  - hist() for histograms; boxplot() for box-plots; pie() for pies; barplot() for bars; ...
  - Special graphs: plot.agnes(), acf(), pacf(), ...
  - Even maps! e.g. map("china"), map("world"), ...

7. Learn to learn!

- If you are able to make good use of "help" and "google", teachers will probably lose their jobs. –By Yihui
- Before you send a query to "help" or "google", please make sure that you know the correct English *terms*!
  - Do you know what are "positive definite", "orthogonal", "diagonal", "eigenvalue", "Augmented Dickey-Fuller Test", "unit root", "pareto distribution", "stepwise", "complex" and "modulus", etc?
- Sources I often use
  - ?
  - http://www.r-project.org (Search the site by: http://finzi.psych.upenn.edu/search.html)
  - Email to the author(s) directly

Some other words

- I'm planning to build an English website for COS, and R is just part of my plan, because up till now I haven't found a good forum for the discussion of R language; there're only mail lists. I believe this is a good opportunity to contact the authors of R language.
- There's a member of COS named "cran" from the University of Auckland, namely the birthplace of R. The other day I chatted with him on MSN and talked about the development of our domestic statistics as well as R.
- I hope some of you would be the pioneers of R language in China.
- I put a optimistic faith in open source software, R *ad hoc*

Thanks!

- To contact me:
  - The best bet would be via email; my address is:
    - paste("xieyihui", "@", "gmail.com", sep="") #run it in R ☺
  - Second choice: www.cos.name
    - browseURL("www.cos.name") #run it in R too!
  - Last choices:
    - Tel: (86)10-82509086
    - Fax: (86)10-82509086