R Programming for Quantitative Finance

Guy Yollin
Applied Mathematics
University of Washington
1. R language overview and history
2. R language references
3. Short R Tutorial
4. The R help system
5. Web resources for R
6. IDE editors for R
J. Adler.
*R in a Nutshell: A Desktop Quick Reference.*
- Chapters 1-3

W. N. Venables and D. M. Smith.
*An Introduction to R.*
2013.
- Sections 1-3
The R programming language

- R is a *language* and *environment* for statistical computing and graphics.

- R is based on the *S language* originally developed by John Chambers and colleagues at AT&T Bell Labs in the late 1970s and early 1980s.

- R (sometimes called "GNU S") is free open source software licensed under the GNU general public license (GPL 2).

- R development was initiated by Robert Gentleman and Ross Ihaka at the University of Auckland, New Zealand in the 1990s.

- R is formally known as The R Project for Statistical Computing.
  - [www.r-project.org](http://www.r-project.org)
Strengths of the R programming language

- Data manipulation
- Data analysis
- Statistical modeling
- Data visualization

Plot from the PerformanceAnalytics package
S language implementations

R is the most recent and full-featured implementation of the S language

- Original S - AT & T Bell Labs
- S-PLUS (S plus a GUI)
  - Statistical Sciences, Inc.†
  - Mathsoft, Inc.
  - Insightful, Inc.
  - Tibco, Inc.
- R - The R Project for Statistical Computing

†Founded by UW Professor Doug Martin, CompFin Program Director

Figure from *The History of S and R, John Chambers, 2006*
The R Timeline

1976 2011
Work on S
Version 1
Exponential
growth of
R Users and
R packages
1993
StatSci Licenses S
1998
John Chambers
1998 ACM Software Award
1993
R on Statlib
1991
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1991
Statistical Models in S (white book)
1993
S3 methods
1988
The New S Language Written in C (Blue Book)
1999
1994
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
1997
R on CRAN GNU Project
1993
R on Statlib
1991
Statistical Models in S (white book)
1991
S3 methods
1988
The New S Language Written in C (Blue Book)
1984
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
1997
R on CRAN GNU Project
1993
R on Statlib
1991
Statistical Models in S (white book)
1991
S3 methods
1988
The New S Language Written in C (Blue Book)
1984
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
1997
R on CRAN GNU Project
1993
R on Statlib
1991
Statistical Models in S (white book)
1991
S3 methods
1988
The New S Language Written in C (Blue Book)
1984
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
1997
R on CRAN GNU Project
1993
R on Statlib
1991
Statistical Models in S (white book)
1991
S3 methods
1988
The New S Language Written in C (Blue Book)
1984
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
1997
R on CRAN GNU Project
1993
R on Statlib
1991
Statistical Models in S (white book)
1991
S3 methods
1988
The New S Language Written in C (Blue Book)
1984
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
1997
R on CRAN GNU Project
1993
R on Statlib
1991
Statistical Models in S (white book)
1991
S3 methods
1988
The New S Language Written in C (Blue Book)
1984
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
1997
R on CRAN GNU Project
1993
R on Statlib
1991
Statistical Models in S (white book) S3 methods
1988
The New S Language Written in C (Blue Book)
1984
S: An Interactive Environment for Data Analysis and Graphics (Brown Book)
1984
S-PLUS developed by Statistical Sciences, Inc. 1988
1988
Modern Applied Statistics with S-PLUS 1994
1997
R on CRAN GNU Project
2000
2002
Modern Applied Statistics with S 4th Edition (S+ 6.x, R 1.5.0)
2001
R 1.4.0 (S4)
2000
R 1.0.0 (S3)
1999
R on CRAN
1999
Recognition of software excellence

Association for Computing Machinery

John Chambers received the 1998 ACM Software System Award

Dr. Chambers’ work will forever alter the way people analyze, visualize, and manipulate data

American Statistical Association

Robert Gentleman and Ross Ihaka received the 2009 ASA Statistical Computing and Graphics Award

In recognition for their work in initiating the R Project for Statistical Computing
Essential web resources

- An Introduction to R
- W.N. Venables, D.M. Smith
- R Development Core Team

**R Reference Card**

- **R Basics**
- Baggott & Short

Definitely obtain these PDF files from the R homepage or a CRAN mirror
Introductory texts

- **R in a Nutshell: A Desktop Quick Reference**
  - Joseph Adler
  - O’Reilly Media, 2009

- **A Beginner’s Guide to R**
  - Zuur, Ieno, Meesters
  - Springer, 2009

Both texts available online through UW library
Introductory texts

- **R in Action**
  - Robert Kabacoff
  - Manning Publications, 2011

- **The Art of R Programming**
  - Norman Matloff
  - No Starch Press, 2011
- **Introductory Statistics with R**
  - P. Dalgaard
  - Springer, 2008

- **Modern Applied Statistics with S**
  - Venables and Ripley
  - Springer, 2002
Experience with other statistical computing languages

For those with experience in MATLAB, David Hiebeler has created a MATLAB/R cross reference document:


For those with experience in SAS, SPSS, or Stata, Robert Muenchen has written R books for this audience:

- http://r4stats.com
R is an *interpreted language*†

- An R interpreter must be running in order to evaluate R commands or execute R scripts
  - RGui which includes an R Console window
  - RStudio which includes an R Console window

†http://en.wikipedia.org/wiki/Interpreted_language
R expressions are processed via R’s Read-eval-print loop †:

The Read-Evaluate-Print Loop (REPL) for R

†http://en.wikipedia.org/wiki/Read-eval-print_loop
The RGui is an *interactive* command driven *environment*:

- Type R commands (expressions) into the R Console
- Copy/Paste multiple R commands into the R Console
- Source an R script
  - An R script is simply a text file of multiple R commands
Interacting with RStudio

The RStudio is an *Integrated Development Environment (IDE)* R:

- Embedded R Console
  - RStudio runs an R interpreter automatically
- Program editor for R
- Plot window
- File browser
- Integrated version control
- R debugger
R makes extensive use of functions†

- Functions can be defined to take zero or more arguments
- Functions typically return a value
  - A return value is not required
- Functions are called by name with any arguments enclosed in parentheses
  - Even if the function has no arguments the parentheses are required

†http://en.wikipedia.org/wiki/Functional_programming
Assigning values to variables

Like other programming languages, values can be stored in variables

- Variables are typically assigned in 1 of 3 ways:
  - assignment operator: <-
  - assignment function: assign
  - equal sign: =
    - must be used to assign arguments in a function call

```r
y <- 5
y
## [1] 5

assign("e", 2.7183)
e
## [1] 2.7183

s = sqrt(2)
s
## [1] 1.4142136

r <- rnorm(n=2)
r
## [1] -1.0067110533 -0.0020828847

s*e+y
## [1] 8.8442567
```
Object orientation in R

Everything in R is an Object†

- Use functions `ls` and `objects` to list all objects in the current workspace

```r
x <- c(3.1416, 2.7183)
m <- matrix(rnorm(9), nrow=3)
tab <- data.frame(store=c("downtown", "eastside", "airport"), sales=c(32, 17, 24))
cities <- c("Seattle", "Portland", "San Francisco")
ls()
```

---

†http://en.wikipedia.org/wiki/Object-oriented_programming
Object classes

All R objects have a *class*

The class of an object determines what it can do and what you can do with it

- Use function `class` to display an object’s class

- There are many R classes; basic classes are:
  - numeric
  - character
  - data.frame
  - matrix

```r
m
## [,1] [,2] [,3]
## [1,] 0.374352397 0.586864810 -0.73778598
## [2,] -0.071532765 -0.262264339 -0.19904931
## [3,] 0.790144078 0.012603635 1.96472235

class(m)
## [1] "matrix"

tab
## store sales
## 1 downtown 32
## 2 eastside 17
## 3 airport 24

class(tab)
## [1] "data.frame"
```
R is a vector/matrix programming language (also known as an array programming language†)

- vectors can easily be created with `c`, the combine function
- most places where a single value can be supplied, a vector can be supplied and R will perform a vectorized operation

```
my.vector <- c(2, 4, 3, 7, 10)
my.vector

## [1]  2  4  3  7 10

my.vector^2

## [1]  4 16  9 49 100

sqrt(my.vector)

## [1] 1.414214 2.000000 1.732051 2.645751 3.162278
```

The `[1]` in the above output is labeling the first element of the vector.

The `c` function can be used to create character vectors, numeric vectors, as well as other types of vectors.
Indexing vectors

Vectors are placed with square brackets: 

Vectors can be indexed in any of the following ways:

- vector of positive integers
- vector of negative integers
- vector of named items
- logical vector

```r
constants[c(1,3,4)]
##  pi  sqrt2  golden
## 3.1416 1.4142 1.6180

constants[c(-1,-2)]
##  sqrt2  golden
## 1.4142 1.6180

constants[c("pi","golden")]
##  pi  golden
## 3.1416 1.6180

constants > 2
##  pi  euler  sqrt2  golden
## TRUE  TRUE  FALSE  FALSE

constants[constants > 2]
##  pi  euler
## 3.1416 2.7183
```
Creating integer sequences with the \texttt{a:b} operator

The sequence operator will generate a vector of integers between \texttt{a} and \texttt{b}. Sequences of this type are particularly useful for indexing vectors, matrices, data.frames etc.

\begin{verbatim}
1:5
## [1] 1 2 3 4 5

-(1:4)
## [1] -1 -2 -3 -4

letters[1:15]
## [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n" "o"

letters[16:26]
## [1] "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z"

letters[-(1:15)]
## [1] "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z"
\end{verbatim}
Comparing vector and non-vector computing

# vectorized operation
# taking the log of each element in a vector
x <- c(97.87, 96.18, 95, 86.39, 88.18, 90.8, 86.06, 82.27, 83.32, 85.3, 83.25, 82.13, 78.54)
log(x)


# non-vectorized computation
# taking the log of each element in a vector
n <- length(x)
y <- rep(0,n)
for( i in 1:n )
  y[i] <- log(x[i])
y

Comparing vector and non-vector computing

# vectorized operation
# taking the log of each element in a matrix
x <- matrix(c(2,9,4,7,5,3,6,1,8),nrow=3)
x^2

## [,1] [,2] [,3]
## [1,] 4 49 36
## [2,] 81 25 1
## [3,] 16 9 64

# non-vectorized computation
# taking the log of each element in a matrix
y <- x
for( i in 1:nrow(x) )
  for( j in 1:ncol(x) )
    y[i,j] <- x[i,j]^2
y

## [,1] [,2] [,3]
## [1,] 4 49 36
## [2,] 81 25 1
## [3,] 16 9 64
Outline

1. R language overview and history
2. R language references
3. Short R Tutorial
4. The R help system
5. Web resources for R
6. IDE editors for R
The HTML help system

R has a comprehensive Html help facility

- Run the `help.start` function
- R GUI menu item
  Help|Html help

```r
help.start()
```

## If nothing happens, you should open

## 'http://127.0.0.1:28913/doc/html/index.html' yourself
The help function

Obtain help on a particular topic via the help function

- `help(topic)`
- `?topic`

`help(read.table)`
The help.search function

Search help for a particular topic via the help.search function

- `help.search(topic)`
- `??topic`

`??predict`
RStudio incorporates a dedicated help tab which facilitates accessing the R Html help system.
Outline

1. R language overview and history
2. R language references
3. Short R Tutorial
4. The R help system
5. Web resources for R
6. IDE editors for R
http://www.r-project.org

- List of CRAN mirror sites
- Manuals
- FAQs
- Site search
- Mailing lists
- Links
CRAN - Comprehensive R Archive Network

http://cran.fhcrc.org

- CRAN Mirrors
  - About 45 countries
  - About 100 sites worldwide
  - About 15 sites in US
- R Binaries
- R Packages
  - 5800+ packages
- R Sources
- Task Views

use your closest CRAN mirror site
CRAN Task Views

Organizes the 5800+ R packages by application

- Finance
- Time Series
- Econometrics
- Optimization
- Machine Learning
Stackoverflow has become the primary resource for help with R

http://stackoverflow.com/
Nerve center of the R finance community

Daily must read

Exclusively for Finance-specific questions, not general R questions

https://stat.ethz.ch/mailman/listinfo/r-sig-finance
Google’s R Style Guide

- Naming convention
- Coding Syntax
- Program Organization

http://google-styleguide.googlecode.com/svn/trunk/google-r-style.html
Quick R

http://www.statmethods.net

Introductory R Lessons

- R Interface
- Data Input
- Data Management
- Basic Statistics
- Advanced Statistics
- Basic Graphs
- Advanced Graphs

Site maintained by Robert Kabacoff, author of R in Action
R graphics details, colors, and other tech notes

Site of Earl Glynn of Stowers Institute for Medical Research

- R Graphics and other useful information
  - R Color Chart
  - Using Color in R (great presentation)
  - Plot area, margins, multiple figures
  - Mixture models
  - Distance measures and clustering
  - Using Windows Explorer to Start R with Specified Working Directory (under tech notes)

http://research.stowers-institute.org/efg/R/index.htm
Programming in R

Online R programming manual from UC Riverside:

- R Basics
- Finding Help
- Code Editors for R
- Control Structures
- Functions
- Object Oriented Programming
- Building R Packages

http://manuals.bioinformatics.ucr.edu/home/programming-in-r
## Other useful R sites

<table>
<thead>
<tr>
<th>Site</th>
<th>Description</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>R Bloggers</td>
<td>Aggregation of about 550 R blogs</td>
<td><a href="http://www.r-bloggers.com">http://www.r-bloggers.com</a></td>
</tr>
<tr>
<td>R Site Search</td>
<td>Search R function help, vignettes, R-help</td>
<td><a href="http://finzi.psych.upenn.edu/search.html">http://finzi.psych.upenn.edu/search.html</a></td>
</tr>
<tr>
<td>Inside-R</td>
<td>R community site by Revolution Analytics</td>
<td><a href="http://www.inside-r.org">http://www.inside-r.org</a></td>
</tr>
</tbody>
</table>
Outline

1. R language overview and history
2. R language references
3. Short R Tutorial
4. The R help system
5. Web resources for R
6. IDE editors for R
RStudio is a fully-featured open-source IDE for R

- R language highlighting
- Paste/Source to R console
- object explorer
- tabbed graphics window
- integrated version control
- 1-click knitr/Sweave compilation

RStudio also provides a server-based version (R running in the cloud)
Revolution Analytics is a company that sells a commercial distribution of R including a desktop IDE

Revolution R Enterprize is free to academic users

- R language highlighting
- Paste/Source code to R
- Source code debugger
- object explorer
- runs R in SDI mode

http://www.revolutionanalytics.com
WinEdt and R-Sweave

Based on WinEdt, an excellent shareware editor with support for \LaTeX{} and Sweave development

- R language highlighting
- Paste/Source code to R
- 1-click Sweave compilation
- Supports R in MDI mode
- Paste/Source code to S-PLUS

http://www.winedt.com
http://www.winedt.org/Config/modes/R-Sweave.php
StatET is a plug-in for the open-source Eclipse development environment

- R language highlighting
- Paste/Source code to R
- Source code debugger
- 1-click Sweave compilation
- Supports R in SDI mode
- Excellent documentation by Longhow Lam

http://www.walware.de/goto/statet
NpptoR is an automation widget (based on AutoHotkey) which allows the very useful program editor Notepad++ to interact with R

- R language highlighting
- Paste/Source code to R
- Supports R in SDI mode

http://notepad-plus-plus.org
http://sourceforge.net/projects/npptor