# Statistics and Computing

*Series Editors:*
J. Chambers
W. Eddy
W. Härdle
S. Sheather
L. Tierney

Springer Science+Business Media, LLC

# Statistics and Computing

*Dalgaard:* Introductory Statistics with R.
*Gentle:* Elements of Computational Statistics.
*Gentle:* Numerical Linear Algebra for Applications in Statistics.
*Gentle:* Random Number Generation and Monte Carlo Methods, 2nd Edition.
*Härdle/Klinke/Turlach:* XploRe: An Interactive Statistical Computing Environment.
*Krause/Olson:* The Basics of S and S-PLUS, 3rd Edition.
*Lange:* Numerical Analysis for Statisticians.
*Loader:* Local Regression and Likelihood.
*Ó Ruanaidh/Fitzgerald:* Numerical Bayesian Methods Applied to Signal Processing.
*Pannatier:* VARIOWIN: Software for Spatial Data Analysis in 2D.
*Pinheiro/Bates:* Mixed-Effects Models in S and S-PLUS.
*Venables/Ripley:* Modern Applied Statistics with S, 4th Edition.
*Venables/Ripley:* S Programming.
*Wilkinson:* The Grammar of Graphics.

W.N. Venables
B.D. Ripley

# S Programming

With 10 Illustrations

Springer

W.N. Venables
CSIRO Marine Laboratories
P.O. Box 120
Cleveland, Queensland 4163
Australia
Bill.Venables@csiro.au

B.D. Ripley
University of Oxford
1 South Parks Road
Oxford OX1 3TG
United Kingdom
ripley@stats.ox.ac.uk

*Series Editors:*

J. Chambers
Bell Labs, Lucent
Technologies
600 Mountain Ave.
Murray Hill, NJ 07974
USA

W. Eddy
Department of Statistics
Carnegie Mellon University
Pittsburgh, PA 15213
USA

W. Härdle
Institut für Statistik und
Ökonometrie
Humboldt-Universität zu Berlin
Spandauer Str. 1
D-10178 Berlin
Germany

S. Sheather
Australian Graduate School
of Management
University of New South
Wales
Sydney, NSW 2052
Australia

L. Tierney
Department of Statistics
and Actuarial Science
The University of Iowa
Iowa City, IA 52242-1419
USA

Library of Congress Cataloging-in-Publication Data
Venables, W.N. (William N.)
S Programming / W.N. Venables, B.D. Ripley.
p. cm. — (Statistics and computing)
Includes bibliographical references and index.
ISBN 978-1-4419-3190-0 ISBN 978-0-387-21856-4 (eBook)
DOI 10.1007/978-0-387-21856-4

1. S (Computer program language). I. Ripley, Brian D., 1952- II. Title. III. Series.
QA76.73.S15 V46 2000
005.13'3—dc21 99-057466

Printed on acid-free paper.


Originally published by Springer-Verlag New York, Inc. in 2000
Softcover reprint of the hardcover 1st edition 2000



9 8 7 6 5 4

*springeronline.com*

# Preface

S is a high-level language for manipulating, analysing and displaying data. It forms the basis of two highly acclaimed and widely used data analysis software systems, the commercial S-PLUS and the Open Source R. This book provides an in-depth guide to writing software in the S language under either or both of those systems.

One of the great strengths of S is its extensibility, and how to exploit that strength is the theme of this book. The companion book, *Modern Applied Statistics with S* (Venables & Ripley, 2002), covers the use of S-PLUS in statistics and data analysis, including using extensions that we and others have written. This volume tells the reader how those extensions were produced and documented. It is aimed at users of S-PLUS or R who want to implement statistical methods or to manipulate data effectively, as well as at those who wish to develop vertical-market applications in S, for example a financial pricing or medical diagnosis system.

Several different implementations of S have appeared. There are currently two S 'engines' in use: S-PLUS 3.x, 4.x and 2000 are based on version 3 of the S language whereas S-PLUS 5.x is based on S version 4. There is also an Open Source system called R which is 'not unlike' version 3 of the S language. We consider all of these, in particular how to use the strengths of one engine to suggest how to develop better programs in another. To avoid endless circumlocutions we will refer to the S language unless we mean a specific dialect; this seems not unfair to R whose design pays homage to S.

A very good way to learn about S programming is to study example code: indeed that is how we had to learn ourselves. All the S code in the system can be listed and studied. We hope too that S programmers will find the extensive libraries we have written to accompany Venables & Ripley (2002) to be instructive; they are the result of years of honing the code to work well on many versions and in particular on all three engines.

Since 1997 the Windows version of S-PLUS has had a graphical user interface in the style of widespread Windows packages, and facilities to write 'user-friendly' graphical interfaces in that style. Since these can be crucial to the adoption of advanced methods written in S, we devote a chapter to the programming of such interfaces.

One very effective way to extend the S is to write interface functions to compiled code written in C, FORTRAN, C++, .... Such extensions are first-class objects in S (unlike some macro-based languages), and compiled code can be used both to interface to existing code, and to speed up methods initially written in S.

Indeed, we have often used the S language for rapid prototyping, transferring key computations to compiled code in C and eventually writing a pure C version. We devote a chapter to using compiled code.

The authors may be contacted by electronic mail as

`Bill.Venables@cmis.csiro.au`
`ripley@stats.ox.ac.uk`

and would appreciate being informed of errors and improvements to the contents of this book. Errata and updates will be made available on-line (see page 4).

To avoid any confusion, S-PLUS is a commercial product, details of which may be obtained from `http://www.insightful.com/`, and R is an Open Source project, with source and binaries available via a network of sites mirroring `http://www.r-project.org/`.

*Acknowledgements:*

This book would not be possible without the S environment which has been principally developed by Rick Becker, John Chambers and Allan Wilks; version 4 of the S language is the result of years of John Chambers' work. The S-PLUS code is the work of a much larger team acknowledged in the manuals for that system. The R project was the inspiration of Ross Ihaka and Robert Gentleman, and incorporates the work of many volunteers.

We are grateful to the many people who contributed to our understanding or have read and commented on draft material. In particular we would like to thank Rich Calaway, John Chambers, Bill Dunlap (the other authority on the S language, apart from its principal author), Nick Ellis, Stephen Kaluzny, José Pinheiro, Charles Roosen, Berwin Turlach and the R–core team (with special thanks to Kurt Hornik for help in implementing our code in R).

Bill Venables
Brian Ripley
December 1999

# Contents

## Appendices

## Typographical Conventions

Throughout this book S language constructs and commands to the operating system are set in a monospaced typewriter font `like this`. The character ~ may appear as ˜ on your keyboard, screen or printer.

We often use the prompts `$` for the operating system (it is the standard prompt for the Unix Bourne shell) and `>` for S. However, we do *not* use prompts for continuation lines, which are indicated by indentation. One reason for this is that the length of line available to use in a book column is less than that of a standard terminal window, so we have had to break lines which were not broken at the terminal.

Some of the S output has been edited. Where complete lines are omitted, these are usually indicated by

```
....
```

in listings; however most *blank* lines have been silently removed. Much of the output was generated with the options settings

```
options(width=65, digits=5)
```

in effect, whereas the defaults are `80` and `7`.

R R differences (at the time of writing) are often signalled by a marginal `R`, as here.