



Computer Science and Data Analysis Series

Statistical Computing with R

Maria L. Rizzo

Bowling Green State University
Bowling Green, Ohio, U.S.A.



Chapman & Hall/CRC

Taylor & Francis Group

Boca Raton London New York

Chapman & Hall/CRC is an imprint of the
Taylor & Francis Group, an **informa** business

Chapman & Hall/CRC
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2008 by Taylor & Francis Group, LLC
Chapman & Hall/CRC is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works
Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number-13: 978-1-58488-545-0 (Hardcover)

This book contains information obtained from authentic and highly regarded sources. Reprinted material is quoted with permission, and sources are indicated. A wide variety of references are listed. Reasonable efforts have been made to publish reliable data and information, but the author and the publisher cannot assume responsibility for the validity of all materials or for the consequences of their use.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (<http://www.copyright.com/>) or contact the Copyright Clearance Center, Inc. (CCC) 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Library of Congress Cataloging-in-Publication Data

Rizzo, Maria L.
Statistical computing with R / Maria L. Rizzo.
p. cm. -- (Chapman & Hall/CRC computer science and data analysis series)
Includes bibliographical references and index.
ISBN-13: 978-1-58488-545-0 (alk. paper)
ISBN-10: 1-58488-545-9 (alk. paper)
1. Mathematical statistics--Data processing. 2. Statistics--Data processing. 3. R
(Computer program language) I. Title. II. Series.

QA276.45.R3R59 2007
519.50285'5133--dc22

2007034218

Visit the Taylor & Francis Web site at
<http://www.taylorandfrancis.com>

and the CRC Press Web site at
<http://www.crcpress.com>

Appendix B

Working with Data Frames and Arrays

B.1 Resampling and Data Partitioning

B.1.1 Using the `boot` function

Bootstrap is implemented in the `boot` function (`boot` package [34]), which provides functions and arguments for the book [63]. In ordinary bootstrap, the samples are selected with replacement. The basic syntax for ordinary bootstrap is

```
boot(data, statistic, R)
```

where `data` is the observed sample and `R` is the number of bootstrap replicates. The default is `sim = "ordinary"`, the ordinary bootstrap (sampling with replacement).

The second argument (`statistic`) is a function, or the name of a function, which calculates the statistic to be replicated. Suppose we call this function f . The `boot` function generates the random indices $i = (i_1, \dots, i_n)$ for each bootstrap replicate, and passes to the function f a copy of the `data` and the index vector i . The function f then computes the statistic $\hat{\theta}^{(b)}$ corresponding to the resampled observations. Example B.1 discusses how to extract the samples for the calculations inside f .

Example B.1 (Extracting a bootstrap sample using an index vector)

We have seen that the `sample` function can be used to sample from a vector with replacement. Equivalently, if x is a vector of length n , we can sample with replacement from the vector of indices `1:n`, and use the resulting value to extract the elements of `x`. Notice that the two methods below generate the same samples.

```

> set.seed(123)
> sample(letters[1:10], size = 10, replace = TRUE)
[1] "c" "h" "e" "i" "j" "a" "f" "i" "f" "e"

> set.seed(123)
> i <- sample(1:10, size = 10, replace = TRUE)
> letters[i]
[1] "c" "h" "e" "i" "j" "a" "f" "i" "f" "e"

```

Similarly, the `[]` operator can be used to extract bootstrap samples from data frames and matrices using `x[i,]`.

```

> x
  [,1] [,2] [,3] [,4]
[1,]  16  14  17  12
[2,]  14  13  16  14
[3,]  13  13  14  11
[4,]  19  11  15  11
[5,]  14  10   8  11

> i
[1] 1 3 3 2 1

> x[i, ]
  [,1] [,2] [,3] [,4]
[1,]  16  14  17  12
[2,]  13  13  14  11
[3,]  13  13  14  11
[4,]  14  13  16  14
[5,]  16  14  17  12

```

The `boot` function will pass a copy of the observed sample `x` and the b^{th} index vector `i`; the user's function `f` (`statistic`) should compute the test statistic on `x[i,]` or `x[i]`. For example, if `x` is a bivariate sample, and the statistic to replicate is correlation, then the function `f` can be written as follows.

```

f <- function(x, i) {
  cor(x[i, 1], x[i, 2])
}

```

For a resampling experiment, it is helpful to code the calculations for the statistic in a function like `f` above, whether or not the `boot` function will be used to run the bootstrap. \diamond

B.1.2 Sampling without replacement

The `boot` function can also be applied in situations where the resampling should be without replacement. For example, in permutation tests, the method of resampling should be `sim = "permutation"`.