RICHARD M. HEIBERGER

COMPUTATION FOR THE ANALYSIS OF DESIGNED EXPERIMENTS

WILEY SERIES IN PROBABILITY AND MATHEMATICAL STATISTICS

Computation for the Analysis of Designed Experiments

RICHARD M. HEIBERGER

Temple University
Philadelphia, Pennsylvania

WILEY

JOHN WILEY & SONS

New York • Chichester • Brisbane • Toronto • Singapore

Copyright © 1989 by John Wiley & Sons, Inc.

All rights reserved. Published simultaneously in Canada.

Reproduction or translation of any part of this work beyond that permitted by Section 107 or 108 of the 1976 United States Copyright Act without the permission of the copyright owner is unlawful. Requests for permission or futher information should be addressed to the Permissions Department, John Wiley & Sons, Inc.

Library of Congress Cataloging in Publication Data:

Heiberger, Richard M., 1945–

Computation for the analysis of designed experiments / Richard M. Heiberger.

p. cm. — (Wiley series in probability and mathematical statistics. Applied probability and statistics)

Bibliography: p.

Includes index.

ISBN 0-471-82735-5

1. Mathematical statistics—Data processing. 2. Analysis of variance—Data processing. 3. Experimental design. I. Title. II. Series.

QA276.4.H44 1989

519.5′028′5—dc19 88-14896

CIP

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

Preface

What happens after an experiment has been designed and executed and the data are in hand? How is a computer program told the structure of the design and the tables needed for its analysis? How does the program calculate the tables? How is the program itself constructed?

Early applied design books assumed the desk calculator was the tool to be used for the analysis. More recently, the assumption has been the ANOVA program within a comprehensive statistics package. Current design texts and package manuals describe the use of ANOVA programs for the analysis of designed experiments. None of those books discusses how the ANOVA program and its components work.

This book is about the computations used in the analysis of designed experiments. Here we provide a synthesis of important concepts from modern computation theory as applied to the construction of program packages for the analysis of statistically designed experiments. The computational issues addressed are not just those from numerical analysis. There are already many books directed toward the statistician on general numerical issues (Kennedy and Gentle, 1980; Chambers, 1977; and Thisted, 1988) and on specific computational topics, such as regression (Maindonald, 1984).

The ideas presented here reflect the state of the computing art on the design of software systems. They are not elsewhere written in a form directed toward statisticians. This book's unique features include the following:

1. Its discussion of the design of software systems for statistical applications. This is the theme of Chapters 3–5 in Part II and in the program sections of the chapters describing algorithms.

2. Its emphasis on the construction and interpretation of sophisticated languages for specification of designs. The emphasis begins with the review of simple designed experiments in Chapter 2 and continues both in Part IV (Chapters 12–14), with discussions of the algebra, grammar, and interpretation of design statements, and in Chapter 15 with a discussion of specification of hypothesis tests. The techniques for interpretation discussed here also apply to the interpretation of familiar algebraic computing languages.

3. A translation of the notation and descriptions of the generally balanced designs to today's computational notation and computational tools in Part V (Chap-

ters 15–18). Several of the more esoteric and arcane calculational recipes from, for instance, Cochran and Cox (1957) appear much simpler when placed in this format.

4. Its presentation of the relationship between the structure of the analysis algorithms and the algebraic and geometric structure of the designs themselves. This material is spread throughout the book beginning in Part III, with the discussions of projections in Chapter 6 and the numerical algorithms in Chapters 7–11, and continuing in Part V with the discussion of the analysis of generally balanced designs in Chapters 15–18.

5. Compilable source code files for all programs in all four languages (Fortran, BASIC, APL, C) discussed in the book are included on the floppy disk packaged with the book. The text of the programs in the book itself was set directly from the files on the disk. Input files for all ANOVA examples discussed in the book are also included on the disk in three package languages (P-STAT, GENSTAT, and SAS). These files include both data and control statements to reproduce the analyses. The Appendix to the book includes a listing of the READ.ME file from the disk and an annotated directory of the files on the disk. The disk itself is formatted for the IBM PC or compatible computers.

This book is written to convey the computational information to the reader with a sound statistics background and fluency in at least one programming language. All programming examples are written in Fortran, BASIC, APL, and C. Familiarity with the use of a statistical package is assumed for Part II, Programming Systems, and to some extent for Part IV, Interpretation of Design Specifications. Familiarity with the material in a one year course in linear statistical methods (e.g., Neter, Wasserman, and Kutner, 1985) is assumed for Part III, Least Squares and Analysis of Variance. In addition, familiarity with the concept of confounding in design is needed for the last three chapters in Part V, Analysis of Designed Experiments.

Readers who desire additional information on the design issues and concepts discussed here are encouraged to seek out references on design. Box, Hunter, and Hunter (1978) provide an introduction to the process and context of design. Cochran and Cox (1957) describe many specific designs and outline the calculations needed for their analysis. In addition, there are many books on design directed toward specific areas of application. Winer (1972), for example, gives a very complete discussion of design with examples taken from the field of psychology.

RICHARD M. HEIBERGER

Philadelphia, Pennsylvania
March, 1989

此为试读,需要完整PDF请访问: www.ertongbook.com

Acknowledgments

The material has been developed and used in advanced courses in design of experiments and in computer approaches to statistical problems, most recently at Temple University and previously at the University of Pennsylvania. Comments and reactions to classroom presentations of this material have been helpful in refining it to the present form. I am particularly indebted to Lilliam Kingsbury, Donna Shand, and Krishnendu Ghosh, who critically read through the complete manuscript. The editors at Wiley have provided me with a very thorough and detailed reading of the manuscript and made suggestions that have improved the structure of the book.

Many of the examples have been illustrated using the ANOVA program which I wrote for P-STAT. David Laurance and Lynne Firester of the P-STAT staff read through the detailed outline and working notes from which the manuscript was constructed. Much of the discussion of systems design issues in Part II of the book is based on discussions with Roald Buhler and David Laurance of P-STAT and my own experience with writing for the package. The parsing program described in Chapter 13 is based on a program developed with David Laurance.

I have also profited from discussions with the designers of GENSTAT during the summer I spent at Rothamsted Experimental Station.

Partial support for research that eventually led to the writing of this book was provided by the National Science Foundation through grants MCS-82-00398, MCS-79-11281, and MCS-75-13994. Computational support for preparation of many of the tables and figures was provided by the National Science Foundation through grant DMS 85-04988 to Temple University.

Portions of the material on the geometry of balanced and confounded designs and on the parsing of design specification statements were presented at the Forty-Second Annual Conference on Applied Statistics, held on December 10, 1986 at New Brunswick, New Jersey.

I am indebted to my wife Mary Morris Heiberger and daughter Sara Rebecca Heiberger for living with the manuscript as an additional family member for several years.

Contents

此为试读，需要完整PDF请访问：www.ertongbook.com

PART 1

Comparative Studies and Designed Experiments

CHAPTER 1

Introduction

The major tool in the analysis of designed experiments is the analysis of variance (ANOVA) procedure in a package of programs for statistical analysis. This book discusses statistical, mathematical, and computational aspects of the construction of packages and ANOVA programs. There are five basic themes:

1. The design of comparative experiments and their statistical analysis by least squares techniques.
2. The general problem of designing, writing, testing, and maintaining large program systems.
3. The construction of the numerical algorithms used to solve the linear equations and provide the least squares analysis.
4. The construction of a language for specifying designs to an ANOVA program, and the parsing programs needed to interpret specifications in the language.
5. The complications introduced into the analysis when the simplifying characteristics of balance and orthogonality are relaxed, and partially balanced and confounded designs are allowed.

Each of these themes is addressed primarily in its own part of the book.

1.1 COMPARATIVE STUDIES AND DESIGNED EXPERIMENTS

The analysis procedures are dictated by the underlying statistical processes. We therefore review in Chapter 2 the statistics of designing experiments for comparative studies. In a comparative study we observe the effect, on the value of a response variable, of different treatments applied to essentially identical objects.

The observed differences are compared in the analysis of variance (ANOVA) table to a measure of the underlying variability associated with multiple observations on identically treated objects. We discuss procedures for reducing the variability, hence increasing the precision, by means of replication and blocking.

We illustrate the differences in interpretation between designs, such as the one-way blocked design and the two-way design without interaction, whose arithmetic is identical. We distinguish the designs notationally with a formal design specification language that recognizes as fundamental the distinction between preexisting classifications of experimental material with blocking factors and experimentally imposed classifications by treatment factors.

1.2 PROGRAMMING SYSTEMS

An ANOVA program exists in a computational environment defined by the package of which it is a component and the host computer system on which it resides. We study the design and documentation features of individual programs needed to make them fit into an environment. We also study strategies for designing and testing entire systems of programs.

Chapter 3 presents and discusses those aspects of program system design that are apparent to all program users. These include the interactive behaviors of the individual program and of the operating system utilities with which the user accesses the program. A complete statistical system requires data input, data modification, and data file maintenance features; flexible output routines that can format arbitrarily scaled output tables to fit well within the confines of a screen or page; and the ability to route printed output to disk files as well as to the standard output unit.

Programs must be tested not only during construction by the programmer and system designer, but also by the end user before they are placed into production use. Correct arithmetic is but one feature to be checked. Boundary conditions (e.g., data sets with a single observation or all of whose observations have the same value) must be handled sensibly.

Chapters 4 and 5 discuss the programming strategies and tools needed to build an efficient, understandable, and easily maintained system. Chapter 4 focuses on the construction of individual subroutines and Chapter 5 on the procedures used to combine them into a system.

Strategies include modular design so that each part can be tested on its own and proper sequencing of development so that the testing environment is in place for each subroutine as it is completed. Individual subroutines from well documented and tested program libraries should be used whenever possible. All programs should be well structured and well documented to make them easier both to write and to maintain.

Communication between subroutines uses the techniques of common storage, argument passing, external files, and scope of variable definition. The choice of overlay versus virtual memory techniques for fitting a large program into a small computer affects the portability and extensibility of the program. Both user and system documentation are needed and are best written simultaneously with program construction.

All programming discussions and examples are illustrated in four commonly used computer languages: Fortran, APL, C, and BASIC. Listings of programs in all four languages that implement the algorithms discussed in Chapters 5, 6, 8–12, 14, and 15 are included on the floppy disk accompanying this book. Printed listings appear for only the first three languages.

1.3 LEAST SQUARES AND ANALYSIS OF VARIANCE

The principal analytic technique used in the analysis of designed experiments is the analysis of variance (ANOVA), a least squares method. In Part III we develop the algebra, geometry, and computing of the least squares method in a general setting and then specialize it to ANOVA.

In Chapter 6 we introduce the idea of projection. The geometry of regression is presented for both the individual space and its dual the variable space. In the more familiar individual space the points for each individual are plotted on a set of axes determined by the variables. In the variable space the points for the observed and dummy variables are plotted on a set of axes determined by the individuals. The geometry of projection, although visible in both spaces, is clearer in the dual space, where sums of squares correspond to squared lengths of vectors and sums of cross products to angles between vectors. The relation between the two spaces is discussed. The matrix algebra of projection is introduced and then used in the construction of orthogonalization procedures and in the least squares process. We illustrate the algebra and geometry of projection operators with programs.

Chapter 7 discusses the relationship of the standard statistical assumption of normality to the usual procedures for testing hypotheses by comparing ratios of mean squares to the F-distribution. The algebra, geometry, and statistics of the t- and F-distributions are discussed and illustrated.

Several algorithms for the calculation of the sums of squares are described and used. Principally, the Wilkinson SWEEP algorithm, which operates directly on the levels of the classification factors, and the Beaton SWP algorithm, which operates on the cross-product matrix of a set of dummy variables generated from the classification factors, are described and illustrated with geometric, as well as algebraic and statistical, interpretations. Both SWEEP and SWP are explicitly projection techniques, yet they look very different. The algorithms are presented both in standard algebraic notation and as subroutines in several standard languages (Fortran, APL, C and BASIC).

Chapter 8 introduces the Wilkinson SWEEP algorithm for balanced designs. The algorithm generalizes the familiar row and column sums in the two-way layout to more complicated designs by using vectors of indices to specify which response values belong to which treatment combinations. Its primary arithmetic operation is the averaging of all response values with the same treatment combination. Its geometry is the sequential projection of the dependent variable Y, thought of as a vector of observations, onto subspaces determined by the index vectors. In Chapter 16 we continue the discussion of Wilkinson's algorithm by extending it to the set of generally balanced designs and then using it to detect confounding and to measure relative efficiency.

Chapter 9 presents the Beaton SWP algorithm, a method of organizing the calculations for multiple regression. In order to use the technique of regression, a method appropriate for continuous independent variables, with the discrete classification factors of analysis of variance, we must first generate dummy variables, based on the classification factors, to specify which response values belong to which treatment combinations. We discuss the problems of linear independence of the dummy variables and the determination of the number of degrees of freedom for effects.

Chapter 10 presents Yates' algorithm for 2^n and 3^m complete factorial designs. It develops the algorithm as a natural generalization from the two-sample t-test and extends it further to arbitrary dimensioned factorial designs by allowing multiplication by orthogonal matrices for factors with more than three treatment levels.

Chapter 11 is devoted to general algorithms for numerical linear algebra. We present the QR factorization, the Householder reflection, the Cholesky factorization, Gaussian elimination, and the LU factorization. We then show the relations among these algorithms and show that SWP is a variant of Gaussian elimination. The presentations of the algorithms include a discussion of important issues of numerical mathematics, distinguishing between the conceptual organization of the calculations and the practical requirements imposed by arithmetic on finite precision machines.

1.4 INTERPRETATION OF DESIGN SPECIFICATIONS

The principal language used throughout the book to describe the factors in a design, and their crossing and nesting relationships with each other, is the notation implemented in the ANOVA program in P-STAT (1986). It is based on Nelder's (1965) notation as implemented in GENSTAT (Alvey, et al., 1977) and GLIM (Baker and Nelder, 1978). The notations used for design specification by other major ANOVA programs [primarily SAS (1985a), SPSS^x (SPSS, Inc., 1983), and BMDP (Dixon 1983)] are also illustrated. All these formal notations are based on the traditional algebraic notations used in most design texts. All numerical examples are included on the floppy disk packaged with this book in the languages of P-STAT, GENSTAT, and SAS.

此为试读，需要完整PDF请访问：www.ertongbook.com