THE RANDOM HOUSE/BIRKHÄUSER MATHEMATICS SERIES

EXPERIMENTS IN COMPUTATIONAL MATRIX ALGEBRA

David R. Hill

Experiments in Computational Matrix Algebra

David R. Hill
Temple University

Cleve B. Moler
Consulting Editor
The MathWorks, Inc.

The Random House/Birkhäuser Mathematics Series

Random House RANDOM HOUSE New York

To the memory of

Emily and Ross

First Edition

9876543

Library of Congress Cataloging-in-Publication Data

Hill, David R. (David Ross), 1943–
Experiments in computational matrix algebra.

(The Random House/Birkhäuser mathematics series)
Bibliography: p.
Includes index.
1. Matrices—Data processing. 2. MATLAB (Computer program). I. Title. II. Series.
QA188.H55 1987 512.9'434'0285 87-3468
ISBN 0-394-35678-0

Preface

For the beginning student in linear algebra this book can be used in conjunction with traditional linear algebra books. For such students our presentation opens the way to using MATLAB for learning and experimenting with linear algebra. This is an opportunity for the students to build a foundation in computational matrix algebra that will be useful in later courses and in their professional careers. The amount of MATLAB involvement can vary depending upon the instructor, the students, and the goals of the course. The exercises are an integral part of this book, and care should be used to incorporate them into the instructional process.

In many cases students taking courses in linear algebra courses are technically oriented. They include students majoring in engineering, the physical sciences, mathematics, economics, biology, and computer science. In most cases such students will have had a year of calculus before taking linear algebra. The calculus background lays the foundation for a variety of applications, the need for numerical methods, and an appreciation for elementary theoretical topics. This book is designed with such students in mind. Text material and exercises are provided so that an instructor can choose applications, numerical involvement, and theoretical depth according to his or her tastes and the students' needs. Proofs are provided only when their exposition will add to the overall understanding of the topic under study.

Although this book is designed for students taking an introductory linear algebra course, there are a number of other ways it can be used. Some alternatives are:

- For students with previous exposure to linear algebra, our presentation provides a self-contained review of major topics and computational aspects.
- For numerical analysis students, we provide a substantial introduction to the solution of linear systems, eigenproblems, and least squares.
- For a short second course in linear algebra.
- For part of a numerical algorithms course which emphasizes computer implementation.
- For professionals desiring to acquire knowledge and skills in fundamentals of linear algebra, this book can be used as a self-paced text.

MATLAB

Experiments in Computational Matrix Algebra uses the powerful MATLAB software to aid in the development of the topics in linear algebra. Instead of using software as an add-on feature in the instructional process, this book uses MATLAP as a major contributor. Topics in linear algebra are developed in such a way that MATLAB promotes learning by doing. We present ideas using **experiments** within MATLAB so that a **conjecture** can be developed. (A conjecture is a guess made from incomplete evidence.) The conjecture can then be checked by further experimentation, hence providing additional experience and evidence. This approach is combined with material covering the core of linear algebra topics. This material is student-oriented and emphasizes understanding of both theory and computational strategies.

MATLAB is an easy-to-use interactive system for matrix algebra. The name MATLAB means "matrix laboratory." The algorithms used by MATLAB are derived from extensive research and represent the state of the art. This, combined with two- and three-dimensional graphics capability, provides a true mathematics laboratory for the instruction of linear algebra.

MATLAB was developed by a group of software professionals under the leadership of Cleve Moler and is available from

The MathWorks, Inc.
Suite 250
20 North Main St.
Sherborn, MA 01770
(617)653-1415

MATLAB must be purchased separately from this book.

Contents and Features

The **Preliminaries** give general information on notational conventions used, together with information about installing and using MATLAB. This material should be read by all users of this book.

Chapter 1 is divided into two parts, parts A and B. Part A introduces notation used in the book, together with basic MATLAB conventions and commands. This material is required for all users of this book. Part B introduces matrix algebra using MATLAB. We have integrated the use of MATLAB into the basic manipulations with entries of a matrix, arithmetic operators, submatrices, matrix products, matrix powers, transposes, matrices of special structure (diagonal, triangular, symmetric, etc.), inner products, vector norms, and orthogonal matrices. MATLAB commands are natural in the sense that they appear in a form analogous to the way matrix algebra is written. Complex matrices and vectors are introduced early because they are easily represented and manipulated by MATLAB. An introduction to complex arithmetic is given in Appendix II. Material on complex matrices may be omitted without loss of continuity, except in Chapter 3. However, we suggest that this not be done. The material presented on matrices is fundamental to the remainder of a course in linear algebra. Similarly, the MATLAB commands introduced are basic to the further use of MATLAB as a problem solving tool in linear algebra. Much of this chapter can be covered on a self-paced basis. We recommend a hands-on group introduction to MATLAB using aids such as the QUICK-Reference Sheet and Exercise Sheet given in Appendix III. Section 1.6 can be postponed until the vectors are covered in an accompanying linear algebra text. Exercises 1.4-32 through 1.4-39 are particularly important for later material. Exercise 1.5-7 is useful in several applications that appear later.

Chapter 2 discusses linear systems of equations. Covered in detail are row operations, Gaussian elimination, reduced row echelon form, pivoting strategies, LU-decomposition, rank, inverses, and conditioning of linear systems. Determinants are discussed within the context of Gaussian elimination, not Laplace expansion by cofactors. **The use of linear systems to determine linear independence and linear dependence is thoroughly discussed**. Appendix I contains material on linear combinations, linear independence and dependence, rank, inverse, determinants, vector spaces, and linear transformations which can be referred to as needed. A wide variety of applications are presented in the extensive sets of exercises.

Chapter 3 discusses eigenvalues and eigenvectors. The Introduction gives five examples where the solutions of the matrix equations $\mathbf{Ax} = \lambda\mathbf{x}$ and $\mathbf{AP} = \mathbf{PD}$ are required. This section provides motivation for studying eigenproblems and the formal solution techniques developed later in the chapter. We recommend that at least one of these examples be studied carefully before beginning Section 3.2. Section 3.2, Eigenproblem I, uses the standard characteristic polynomial approach. Hand calculation steps are matched with MATLAB procedures. The models presented in the Introduction are discussed further. In addition, theoretical concepts are studied. Section 3.3, Eigenproblem II, introduces the QR eigenmethod. This iterative procedure is developed using MATLAB commands and shows how theory and computational practice complement each other. The notion of origin shifts has been omitted, as have the classical algorithms based upon rotations. This section is optional for an introductory course in linear algebra. If Section 3.3 is to be covered, then Exercises 3.2-50 through 3.2-57 on the Gram-Schmidt process and Exercises 3.2-58 through 3.2-61 on the QR matrix factorization must be covered in detail. As background for Section 3.3, we recommend Appendix I.7 on linear transformations.

Chapter 4 discusses programming within MATLAB. Although MATLAB is technically not a programming language, it does support commands that permit repetitive execution of statements, testing of logical conditions, interactive data input and output, and the construction of algorithms. Topics covered include relational operators, if-then-else, while-loops, for-loops, text-strings, "instant subroutines," file manipulation commands, and extending MATLAB commands. Previous exposure to a high-level programming language is helpful, but not required. This chapter is optional for introductory linear algebra students, but is highly recommended for students in numerical analysis or computational linear algebra, and for those interested in numerical algorithms. The exercises provide an opportunity to use MATLAB for various types of computations.

Chapter 5 is a set of four applications modules. Section 5.1 discusses elementary graph theory and uses only matrix algebra. The notions of an adjacency matrix and dominance-directed graphs are presented with various applications. This module may be covered after Section 1.4 has been completed. MATLAB should be used to perform the calculations needed. Section 5.2 discusses Markov processes. The material begins with elementary notions and builds to using eigenvalues and eigenvectors. It is possible to use the elementary notions together with Example 3.1-3 and then finish the material after the completion of Section 3.2. We introduce an elementary model for Brownian motion. Hence this module is more sophisticated than that in Section 5.1. Section 5.3 discusses graphics and graphics commands in MATLAB. The easy availability of both two- and three-dimensional graphics gives MATLAB great versatility. We present a series of examples to demonstrate the commands and show how to combine previous topics with graphics. We demonstrate the power method and the sensitivity of eigenvalue computation using graphics. This section is optional for beginning linear algebra students, but could be used for independent study. At various places in Chapters 1 through 4 we have used graphics and have supplied the corresponding commands. This may be enough to encourage students to read this section. We recommend that serious users of MATLAB cover the material in this section. Section 5.4 presents the topic of least-squares solutions of linear systems using pseudoinverses. This section relies heavily on Chapter 2, which discusses the solution of systems of linear equations, and Appendix I.8, which discusses singular value decomposition. We have included this section for students who have mastered all of Chapter 2 and desire to study this important topic. This is the most ambitious of the modules in this chapter.

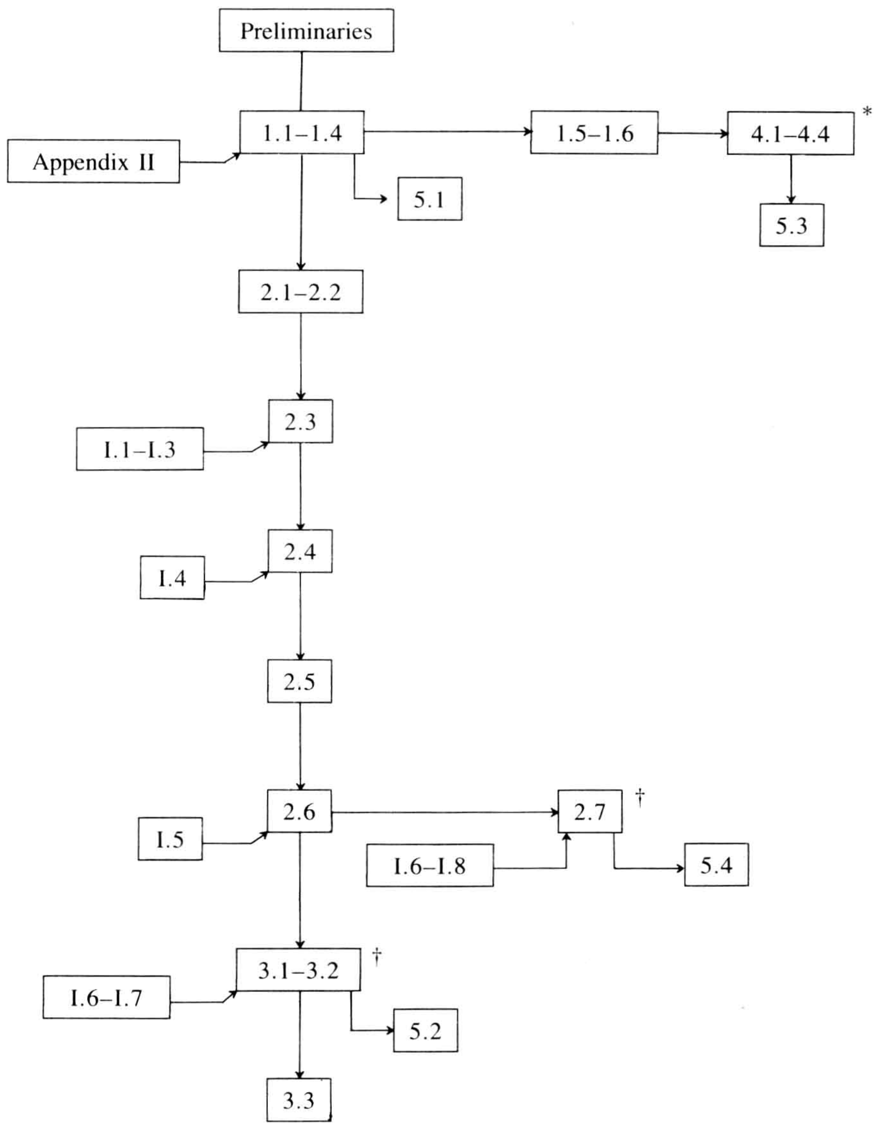
The **Appendices** have been included primarily for students needing a review of topics and for a quick reference check. Appendix I includes discussions of linear combinations, linear independence and dependence, rank of a matrix, inverse of a matrix, the determinant, vector spaces, linear transformations, and singular value decomposition. Appendix II reviews complex numbers and their properties. Appendix III includes several aids for getting students started using MATLAB. The QUICK-Reference Sheet is a succinct summary of elementary commands that incorporates a number of ideas from Chapter 1. In addition, the Exercise Sheet can be used as a set of examples to be done in MATLAB to illustrate the commands on the QUICK-Reference Sheet. Finally, a Command Reference Sheet for all MATLAB commands discussed in this book is provided. There is also a Command and Function Index which gives a brief description of the MATLAB instruction and a page reference for a more detailed explanation.

There is a table of contents for each chapter that lists the concepts discussed in the text or exercises and the MATLAB commands introduced in each section.

Solutions to selected exercises are included. At times there may be more than one way to formulate an answer to a problem. Also the design of experiments can vary.

Major Dependencies

The following chart illustrates the major dependencies of the material in this book. We indicate how Appendix I and Appendix II relate to the topics in Chapters 1, 2, and 3. In addition we show how the application modules in Chapter 5 can be integrated into the linear algebra topics. Chapter 4, or portions of it, can be used after the basic topics in Chapter 1.



*includes applications from Chapter 2 and Chapter 3

†requires Section 1.6

There are a number of alternatives for using this book. We present several suggestions for courses.

The material for a first course in linear algebra might include:

Chapter 1, all sections. (Section 1.6 can be postponed until vector norms are covered in an accompanying linear algebra text.)
Chapter 2, Sections 1–6.
Chapter 3, Sections 1–2.
Chapter 5, Sections 1–2.
Other topics and sections according to time available and instructor's preference. Appendices as needed.

Material for a second course in linear algebra might include (assuming no previous exposure to MATLAB):

Chapter 1, all sections.
Chapter 2. Review Sections 1–4, and cover Sections 5–7.
Section 5.4 and Appendix I.8.
Chapter 3, all sections. Review most of Section 3.2, develop the power method, and cover the exercises thoroughly. Concentrate on Section 3.3 and its exercises.
Selected topics and exercises from Chapter 4 and Sections 5.1–5.3. Appendices as needed.

Material for use as part of a numerical analysis course for the study of linear systems and eigenproblems might include:

Chapter 1. Primarily for self-study emphasizing MATLAB commands.
Chapter 2. Review Sections 1–4 and 6 covering the \ operator. Discuss Section 2.5 briefly, but concentrate on Section 2.7.
Chapter 3. Assign Section 3.1 for reading and treat the majority of Section 3.2 as review except for the power method. Discuss Section 3.3 thoroughly.
Selected topics from the exercises on iterative solution of linear systems.
Optionally Section 5.4 and Appendix I.8.

The author would be extremely grateful for suggestions regarding other ways to use this book.

Acknowledgments

Many people contributed to the final form of the book. First, I would like to thank the reviewers whose comments and suggestions helped clarify and focus the material presented:

George Davis, Georgia State University
Stephen Dragosh, Michigan State University

Roger Horn, The Johns Hopkins University
David Kraines, Duke University
Gary Kurowski, University of California, Davis
Robert Moore, University of Washington
Wlodek Proskurowski, University of Southern California
Louis Rall, University of Wisconsin–Madison
P. K. Wong, Michigan State University

Thanks also to the students in classes at Temple University who struggled with drafts of portions of this book. Their feedback and enthusiasm helped shape the text and exercises. A special thanks to John Ferraro who helped check the computations and displays and solve the exercises.

I would like to thank Cleve Moler and John Little of The MathWorks for their cooperation in obtaining and using MATLAB to develop this book. A number of stimulating discussions contributed to this work.

On the editorial side, I would like to thank John Martindale of Random House, who coordinated this project from its inception and offered constant encouragement. Thanks also to Edward Huddleston for his patience and expertise throughout the editing process.

Finally, I would like to thank my family for their patience, encouragement, and tolerance throughout this project.

Any errors that appear are the sole responsibility of the author.

Contents

Preliminaries

In this introductory section we describe notational conventions used throughout the book and present some general guidelines concerning the use of microcomputers with MATLAB. We have tried to maintain standard mathematical notation for linear algebra, though it is probably more accurate to say we have used the current notation, since symbolic conventions change. Specific symbols are defined as they are introduced and illustrative examples are given. In the numbering of equations, theorems, definitions, examples, figures, and tables, we use the form *xx.yy-zz,* which represents item number *zz* in Section *yy* of Chapter *xx*. For example, Theorem 3.2-4 is the fourth theorem in Section 2 of Chapter 3. Exercises are numbered in the same way. References appear in brackets and give the author's name and the year of publication. The only exception to this convention is reference to the *MATLAB User's Guide,* which appears as just displayed.

When MATLAB commands are displayed, they usually appear in a dot-matrix typeface, as do the input and output for MATLAB; otherwise, the commands are in boldface type. We have tried to present MATLAB, output in the form that appears on the screen. This will aid in learning to use MATLAB, since we recommend you work the examples as you read the book.

The examples and exercises in this book were developed on MS-DOS compatible personal computers. There may be some variation on other machines. Check the corresponding *MATLAB User's Guide* if variations occur. New versions of MATLAB may also affect the behavior of commands.

The remainder of this section discusses the installation of MATLAB and mechanics associated with using MATLAB. Before using MATLAB, read this material and compare it with any similar information furnished by your instructor or computer center. The information that follows should be used for reference as needed.

The MATLAB software is designed to run on various machines including MS-DOS compatible personal computers, Apple Macintosh, SUN Workstations, and VAX computers. The following information on operating instructions and hardcopy output is primarily directed towards MS-DOS compatible personal computers. Corresponding information for other machines is available in the appropriate *MATLAB User's Guide* for the particular machine.

The actual brand of computer you use is a personal choice, but there are certain general requirements:

- At least 320K of memory.
- Mathematics coprocessor chip.
- MS-DOS, Version 2.0 or higher.
- At least one double-sided, double-density floppy disk drive.

It is highly recommended that the user have access to more memory (up to 640K), graphics capability, a second floppy disk drive or hard disk, and a printer. Specific details concerning these options can be found in the *MATLAB User's Guide*. In addition, Section 5.3 gives some general information about graphics.

If a printer is available to you, you may follow these procedures to obtain hardcopy of text information:

1. Depressing the **shift PrtSc** keys prints the contents of the screen.
2. Depressing the **Ctrl PrtSc** keys sends each subsequent input and output line to the printer until **Ctrl PrtSc** is depressed again.

For information about obtaining graphics hardcopy, see the *MATLAB User's Guide* or Section 5.3 in this book.

When MATLAB diskettes are received from The MathWorks, Inc., an installation procedure must be followed before the software can be used. If you perform the installation yourself, be sure to follow the installation instructions in the accompanying *MATLAB User's Guide*. We will assume that MATLAB has successfully been installed either on diskettes or on a hard disk. If you are unfamiliar with procedures described in the *User's Guide,* get help from your local computer center. If difficulties arise, contact The MathWorks directly.

After the installation of the software onto **working diskettes** (or a hard disk) has been completed, you are ready to start up MATLAB. Specific startup instructions are available in the *MATLAB User's Guide*. For a two-floppy drive system the general startup steps are as follows. Assuming the machine is off and the **Working Program Diskette** is self-booting,

1. Insert the MATLAB **Working Program Diskette** into drive A: and the MATLAB **Working Utilities Diskette** into drive B:.
2. Turn on the machine.
3. After the system boots and the prompt > appears, type

   ```
   B:
   ```

   followed by **enter.** Then type command

   ```
   PATH=A:\;B:\;B:\MATLAB
   ```

   followed by **enter.** To execute, type

   ```
   MATLAB
   ```

   followed by **enter.**

MATLAB is ready for use when the MathWorks greeting comes up on the screen and the **MATLAB prompt** >> appears. A line-by-line explanation of the startup steps is given in the *MATLAB User's Guide*. For student use, it is recommended that the instructor or computer-center personnel place these commands in an

AUTOEXEC.BAT file so that they will be executed automatically at startup. (For details on constructing an AUTOEXEC.BAT file, see your MS-DOS user's manual.) If your machine has graphics capability and you have access to a graphics-compatible printer, we recommend that a graphics driver program be loaded as part of the AUTOEXEC.BAT file. (See Section 5.3 or the *MATLAB User's Guide* for details.) The following is an autoexec file that the author has used with student groups.

```
echo off
graphics
b:
path=a:\;b:\;b:\matlab
matlab
```

In this autoexec file, "graphics" is the graphics driver program that is on the MS-DOS system disk. It must be copied onto MATLAB's **Working Program Diskette** when using floppy disk drives.

Finally, here are some guidelines for handling diskettes:

- Grasp diskettes by the label. Avoid touching the recording surface of a diskette.
- To write on a diskette label use a felt-tip pen, never a pencil or ball-point pen.
- To insert a diskette into a floppy disk drive, hold the diskette by the label, with the label facing up and with the cut-out notch on your left. Gently slide the diskette into the drive and close the drive door or locking mechanism. To remove a diskette, reverse the procedure.
- When finished using diskettes, return them to their protective covers.
- Transport diskettes in a carrying case to avoid bending.
- Store diskettes away from sunlight, heat, magnets, and magnetic fields that may be present around motors and copy machines, chalk dust, and dirt.
- Keep a backup copy of important material and update it regularly.

Chapter 1
Beginning to Use MATLAB

Part A Introducing MATLAB

Section 1.1 Introduction to Matrices

Section 1.2 Matrices in MATLAB

Section 1.3 Basic Variables and Functions

Part B Matrix Algebra in MATLAB

Section 1.4 Building Expressions and Operations with Variables

Section 1.5 Basic Matrix Functions

Section 1.6 Data Manipulation Commands and Vector Norms