Preface

This book is about matrix and linear algebra, and their applications. For many students the tools of matrix and linear algebra will be as fundamental in their professional work as the tools of calculus; thus it is important to ensure that students appreciate the utility and beauty of these subjects as well as the mechanics. To this end, applied mathematics and mathematical modeling ought to have an important role in an introductory treatment of linear algebra. In this way students see that concepts of matrix and linear algebra make concrete problems workable.

In this book we weave significant motivating examples into the fabric of the text. I hope that instructors will not omit this material; that would be a missed opportunity for linear algebra! The text has a strong orientation toward numerical computation and applied mathematics, which means that matrix analysis plays a central role. All three of the basic components of linear algebra — theory, computation, and applications — receive their due. The proper balance of these components gives students the tools they need as well as the motivation to acquire these tools. Another feature of this text is an emphasis on linear algebra as an experimental science; this emphasis is found in certain examples, computer exercises, and projects. Contemporary mathematical software make ideal "labs" for mathematical experimentation. Nonetheless, this text is independent of specific hardware and software platforms. Applications and ideas should take center stage, not software.

This book is designed for an introductory course in matrix and linear algebra. Here are some of its main goals:

- To provide a balanced blend of applications, theory, and computation that emphasizes their interdependence.
- To assist those who wish to incorporate mathematical experimentation through computer technology into the class. Each chapter has computer exercises sprinkled throughout and an optional section on computational notes. Students should use the locally available tools to carry out the

experiments suggested in the project and use the word processing capabilities of their computer system to create reports of results.
- To help students to express their thoughts clearly. Requiring written reports is one vehicle for teaching good expression of mathematical ideas.
- To encourage cooperative learning. Mathematics educators are becoming increasingly appreciative of this powerful mode of learning. Team projects and reports are excellent vehicles for cooperative learning.
- To promote individual learning by providing a complete and readable text. I hope that readers will find the text worthy of being a permanent part of their reference library, particularly for the basic linear algebra needed in the applied mathematical sciences.

An outline of the book is as follows: Chapter 1 contains a thorough development of Gaussian elimination. It would be nice to assume that the student is familiar with complex numbers, but experience has shown that this material is frequently long forgotten by many. Complex numbers and the basic language of sets are reviewed early on in Chapter 1. Basic properties of matrix and determinant algebra are developed in Chapter 2. Special types of matrices, such as elementary and symmetric, are also introduced. About determinants: some instructors prefer not to spend too much time on them, so I have divided the treatment into two sections, the second of which is marked as optional and not used in the rest of the text. Chapter 3 begins with the "standard" Euclidean vector spaces, both real and complex. These provide motivation for the more sophisticated ideas of abstract vector space, subspace, and basis, which are introduced largely in the context of the standard spaces. Chapter 4 introduces geometrical aspects of standard vector spaces such as norm, dot product, and angle. Chapter 5 introduces eigenvalues and eigenvectors. General norm and inner product concepts for abstract vector spaces are examined in Chapter 6. Each section concludes with a set of exercises and problems.

Each chapter contains a few more "optional" topics, which are independent of the nonoptional sections. Of course, one instructor's optional is another's mandatory. Optional sections cover tensor products, linear operators, operator norms, the Schur triangularization theorem, and the singular value decomposition. In addition, each chapter has an optional section of computational notes and projects. I employ the convention of marking sections and subsections that I consider optional with an asterisk.

There is more than enough material in this book for a one-semester course. Tastes vary, so there is ample material in the text to accommodate different interests. One could increase emphasis on any one of the theoretical, applied, or computational aspects of linear algebra by the appropriate selection of syllabus topics. The text is well suited to a course with a three-hour lecture and lab component, but computer-related material is not mandatory. Every instructor has her/his own idea about how much time to spend on proofs, how much on examples, which sections to skip, etc.; so the amount of material covered will vary considerably. Instructors may mix and match any of the optional sections according to their own interests, since these sections are largely independent of each other. While it would be very time-consuming to cover them all, every instructor ought to use some part of this material. The unstarred sections form the core of the book; most of this material should be covered. There are 27 unstarred sections and 10 optional sections. I hope the optional sections come in enough flavors to please any pure, applied, or computational palate.

Of course, no one size fits all, so I will suggest two examples of how one might use this text for a three-hour one-semester course. Such a course will typically meet three times a week for fifteen weeks, for a total of 45 classes. The material of most of the unstarred sections can be covered at a rate of about one and one-half class periods per section. Thus, the core material could be covered in about 40 class periods. This leaves time for extra sections and in-class exams. In a two-semester course or a course of more than three hours, one could expect to cover most, if not all, of the text.

If the instructor prefers a course that emphasizes the standard Euclidean spaces, and moves at a more leisurely pace, then the core material of the first five chapters of the text are sufficient. This approach reduces the number of unstarred sections to be covered from 27 to 23.

I employ the following taxonomy for the reader tasks presented in this text. *Exercises* constitute the usual learning activities for basic skills; these come in pairs, and solutions to the odd-numbered exercises are given in an appendix. More advanced conceptual or computational exercises that ask for explanations or examples are termed *problems*, and solutions for problems are not given, but hints are supplied for those problems marked with an asterisk. Some of these exercises and problems are computer-related. As with pencil-and-paper exercises, these are learning activities for basic skills. The difference is that some computing equipment (ranging from a programmable scientific calculator to a workstation) is required to complete such exercises and problems. At the next level are *projects.* These assignments involve ideas that extend the standard text material, possibly some numerical experimentation and some written exposition in the form of brief project papers. These are analogous to lab projects in the physical sciences. Finally, at the top level are *reports.* These require a more detailed exposition of ideas, considerable experimentation — possibly open ended in scope — and a carefully written report document. Reports are comparable to "scientific term papers." They approximate the kind of activity that many students will be involved in throughout their professional lives. I have included some of my favorite examples of all of these activities in this textbook. Exercises that require computing tools contain a statement to that effect. Perhaps projects and reports I have included will provide templates for instructors who wish to build their own project/report materials. In my own classes I expect projects to be prepared with text processing software to which my students have access in a mathematics computer lab.

About numbering: exercises and problems are numbered consecutively in each section. All other numbered items (sections, theorems, definitions, etc.) are numbered consecutively in each chapter and are prefixed by the chapter number in which the item occurs.

Projects and reports are well suited for team efforts. Instructors should provide background materials to help the students through local system-dependent issues. When I assign a project, I usually make available a Maple, Matlab, or Mathematica notebook that amounts to a brief background lecture on the subject of the project and contains some of the key commands students will need to carry out the project. This helps students focus more on the mathematics of the project rather than computer issues. Most of the computational computer tools that would be helpful in this course fall into three categories and are available for many operating systems:

- Graphing calculators with built-in matrix algebra capabilities such as the HP 48, or the TI 89 and 92.
- Computer algebra systems (CAS) such as Maple, Mathematica, and Macsyma. These software products are fairly rich in linear algebra capabilities. They prefer symbolic calculations and exact arithmetic, but can be coerced to do floating-point calculations.
- Matrix algebra systems (MAS) such as Matlab, Octave, and Scilab. These software products are specifically designed to do matrix calculations in floating-point arithmetic and have the most complete set of matrix commands of all categories.

In a few cases I include in this text software-specific information for some projects for purpose of illustration. This is not to be construed as an endorsement or requirement of any particular software or computer. Projects may be carried out with different software tools and computer platforms. Each system has its own strengths. In various semesters I have obtained excellent results with all these platforms. Students are open to all sorts of technology in mathematics. This openness, together with the availability of inexpensive high-technology tools, has changed how and what we teach in linear algebra.

I would like to thank my colleagues whose encouragement has helped me complete this project, particularly David Logan. I would also like to thank my wife, Muriel Shores, for her valuable help in proofreading and editing the text, and Dr. David Taylor, whose careful reading resulted in many helpful comments and corrections. Finally, I would like to thank the outstanding staff at Springer, particularly Mark Spencer, Louise Farkas, and David Kramer, for their support in bringing this project to completion.

I continue to develop a linear algebra home page of material such as project notebooks, supplementary exercises, errata sheet, etc., for instructors and students using this text. This site can be reached at

`http://www.math.unl.edu/~tshores1/mylinalg.html`

Suggestions, corrections, or comments are welcome. These may be sent to me at `tshores1@math.unl.edu`.

http://www.springer.com/978-0-387-33194-2

Applied Linear Algebra and Matrix Analysis
Shores, Th.S.
2007, XII, 384 p. 27 illus., Hardcover
ISBN: 978-0-387-33194-2