INTRODUCTION TO MATRIX COMPUTATIONS

G. W. Stewart

INTRODUCTION TO MATRIX COMPUTATIONS

G. W. Stewart

The University of Texas at Austin

Academic Press

New York and London

A Subsidiary of Harcourt Brace Jovanovich, Publishers

ACADEMIC PRESS, INC.
111 Fifth Avenue, New York, New York 10003

United Kingdom Edition published by
ACADEMIC PRESS, INC. (LONDON) LTD.
24/28 Oval Road, London NW1

LIBRARY OF CONGRESS CATALOG CARD NUMBER: 72-82636

AMS (MOS) 1970 Subject Classifications: 65F05, 65F10, 65F15, 65F20, 65F25, 65F30, 65F35

PRINTED IN THE UNITED STATES OF AMERICA

PREFACE

Speaking in 1966 before the Society of Industrial and Applied Mathematics, the late George E. Forsythe, then Chairman of the Computer Science Department at Stanford University, stated:

> It is safe to say that matrix computation has passed well beyond the stage where an amateur is likely to think of computing methods which can compete with the better-known methods. Certainly one cannot learn theoretical linear algebra and an algebraic programming language, and nothing else, and start writing programs which will perform acceptably by today's standards. There is simply too much hard-earned experience behind the better algorithms, and yet this experience is hardly mentioned in mathematical textbooks of linear algebra.

Professor Forsythe went on to point out that most of this hard-earned experience has been accumulated since 1953. The impetus for this extraordinary development of numerical linear algebra has been provided by the demand for efficient, self-contained matrix algorithms suitable for use on a high-speed digital computer.

Today linear algebra textbooks still do not mention this hard-earned experience. The gains of the past 20 years are, for the most part, contained in advanced treatises, in journal articles, and even in unpublished technical reports. The consequence of this is that many people whose daily business involves computations with matrices are unacquainted with the best algorithms and their properties. Moreover, specialists in other areas of numer ical analysis are frequently unaware of how the techniques of numerical linear algebra may be applied to their problems.

The purpose of this book is to provide a reasonably elementary introduction to some of the more important algorithms for matrix computations and to the techniques by which these algorithms may be analyzed. It is addressed to the beginner in numerical analysis as well as to the advanced student in the sciences who wishes to know more about the art of matrix computations. While the book is intended primarily as a text, it is hoped that its supplementary notes and bibliography will also make it a useful reference.

Numerical linear algebra is far too broad a subject to treat in a single introductory volume. I have chosen to treat algorithms for solving linear systems, linear least squares problems, and eigenvalue problems involving matrices whose elements can all be contained in the high-speed storage of a computer. By way of theory, I have chosen to discuss the theory of norms and perturbation theory for linear systems and for the algebraic eigenvalue problem. These choices exclude, among other things, the solution of large sparse linear systems by direct and iterative methods, linear programming, and the useful Perron–Frobenious theory and its extensions. However, a person who has fully mastered the material in this book should be well prepared for independent study in other areas of numerical linear algebra.

Since most of the algorithms discussed in this book have been published as ALGOL or FORTRAN programs, no program listings are given. However, it is a long step from a mathematical description of an algorithm to its efficient implementation on a computer. To illustrate the common techniques for conserving computer storage and operations, I have presented many of the algorithms in an informal algorithmic language, which is described in Chapter 2.

Some of the most useful results in numerical analysis consist of observations that cannot be proved in general but are nonetheless true most of the time. For example, the stability of Gaussian elimination with partial pivoting depends on the elements of the reduced matrices remaining of moderate size. Since matrices are known for which the elements become quite large, one cannot prove unconditionally that Gaussian elimination is stable. However, it has been observed that this growth does not occur with the matrices one usually encounters in practice. Similarly, error results phrased in terms of a vector norm give an imprecise idea of what is happening to the individual components of the vector; but in many applications such an imprecise idea is sufficient. It would be wrong to exclude such observations from the book because they are not mathematically rigorous. However, it is also important for the beginner to have a clear idea of what, on one hand, can be proved about an algorithm and what, on the other, is true of it most of the time. Accordingly, I have segregated the rigorous results into theorems and left discussions and empirical observations in the body of the text.

One of the major advances in numerical linear algebra has been the development of techniques for analyzing the effects of rounding error on matrix algorithms. However, even though the analyses are usually conceptually straightforward, they are often very tedious to present. Moreover, concentration on the details of a rounding-error analysis often obscures real purpose of the analysis, which is to demonstrate the stability of an algorithm or to expose the conditions under which it may become unstable. For this reason, although results from rounding-error analyses are frequently quoted in the text, only two representative analyses are given in Appendix 3.

The first chapter of the book contains a fairly complete review of elementary linear algebra, and a bright student might find it sufficient. Ordinarily, though, a course based on this book should require a year of calculus and a sophomore course in linear algebra or matrix theory, in which case the material in the first chapter can be used selectively to fill gaps in the background of the students. Particular attention should be paid to Sections 1.3 and 1.4, which treat material on matrix structure and matrix operations that is not usually stressed in linear algebra courses.

An instructor should be able to use the book at several levels, since really difficult material is introduced only in the later chapters. In particular, the content of the first three chapters supplemented by material from Sections 4.4 and 4.5 would comprise a respectable elementary course in the direct solution of linear systems. I have taught the contents of Chapters 1–5 in a one-semester introductory undergraduate course and the entire book in a one-semester graduate course. By emphasizing the derivations of the algorithms and the meanings of the theorems (as opposed to their proofs), it should be possible to present a large part of this book to a relatively unsophisticated audience. However, this can be carried too far. For example, a good grasp of the notion of orthogonality is required for the section on the linear least squares problem.

The nature of the material does not lend itself well to routine exercises, and the problems at the end of each section are relatively difficult. However, many of them have concise solutions that exploit the power of matrix methods, and the student will find it profitable to hunt for them. The instructor can of course supplement the problems with programming assignments.

ACKNOWLEDGMENTS

No book on numerical linear algebra can avoid drawing heavily on the pioneering work of J. H. Wilkinson, who, more than any one person, is responsible for the present high state of the art. I am happy to acknowledge my debt to him. I must also thank the many friends and colleagues who have helped and encouraged me in writing this book, especially those in the Center for Numerical Analysis at the University of Texas. I am specifically indebted to R. E. Funderlic, D. R. Kinkaid, W. R. Rheinboldt, and R. A. Tapia, who made detailed comments on the manuscript; to Linda Brothers who typed it; and to Margie Blevins, who has helped me at every stage and corrected my spelling errors.

More personally, I should like to thank Mrs. Ival Aslinger for introducing me to mathematics. I hope she will find this book by one of her grateful students a small reward for many years of inspired high school teaching.

I owe most to my friend and teacher A. S. Householder, to whom this book is dedicated.

CONTENTS

PRELIMINARIES

CHAPTER 1

The subject of this book is the description and analysis of computational methods involving vectors and matrices. In this chapter we shall develop the elementary theory which underlies our subject. This development has two aspects: first the definition of vectors, matrices, and their operations; second the abstract relationships between various concepts that grow out of the idea of a vector or a matrix, such as linear dependence, column spaces, and so on. Facility with matrix operations is required to understand the description of the algorithms to be presented later; insight into matrix theory is required to understand their analyses.

We shall be concerned with real n-space. When $n = 2$ this is, in effect, the Euclidean plane, and when $n = 3$, the three-dimensional space of our everyday experience. It follows that many general theorems can be visualized as geometric facts about the plane or three-dimensional space. Conversely, our geometric knowledge of two- or three-dimensional space can often be directly extended to general theorems about n-space. We shall develop this geometric point of view informally in this chapter.

Most of the algorithms dealing with the rectangular arrays of numbers called matrices proceed by a succession of transformations that introduce new zeros into the array, finally arriving at a conveniently simple form. It is not surprising then that there is an extensive terminology associated with the distribution of zeros in a matrix. Moreover, it is important to know what distributions are preserved by the standard matrix operations. These points are also treated in this chapter.

1. THE SPACE R^n

The idea of a vector in real n-dimensional space is a natural generalization of the representation of points in a plane by Cartesian coordinates. In this representation, a special point is distinguished as an origin and two perpendicular lines called coordinate axes are constructed through this point. Then each point p in the plane can be represented as an ordered pair (ξ_1, ξ_2) whose first and second elements are obtained by projecting p on the first and second coordinate axes, respectively (Fig. 1). Of course the point p and the ordered pair or vector (ξ_1, ξ_2) are different objects; however, their relation is so intimate that one often speaks of the point (ξ_1, ξ_2) and proves geometric theorems about the plane by manipulating ordered pairs of numbers rather than points.

Following this lead, we shall speak of ordered n-tuples as n-vectors. However, for reasons that will become clear later, it is convenient to arrange these numbers in a column rather than in a row.

DEFINITION 1.1. An *n-vector* x is a collection of n real numbers $\xi_1, \xi_2, \ldots, \xi_n$ arranged in order in a column:

$$x = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \cdot \\ \cdot \\ \cdot \\ \xi_n \end{pmatrix}$$

The numbers $\xi_1, \xi_2, \ldots, \xi_n$ are called the *components* of x.

When n is fixed or known from context, it is customary to refer to the vector x rather than the n-vector x. We shall denote the set of all n-vectors by R^n, which will be called the *vector space* R^n, or simply the *space* R^n. The field of real numbers will be denoted by R and will be referred to as *scalars*.

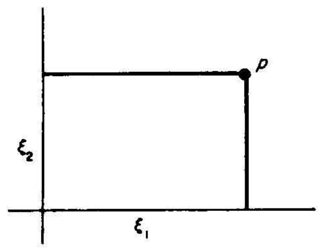

Fig. 1

EXAMPLE 1.2. The elements of the vector space R^1 may be placed in a natural correspondence with the elements of the field R as follows. Each vector $(\alpha) \in R^1$ is associated with the scalar $\alpha \in R$ and vice versa. Strictly speaking, the space R^1 and the field R are distinct mathematical objects. However, we shall often identify the two, using members of R^1 as scalars and conversely regarding scalars as members of R^1.

The following notational conventions will be used throughout the book. Lower case Greek letters will denote scalars; lower case Latin letters will denote vectors. Wherever possible we shall attempt to represent the components of a vector by the corresponding Greek letter. Thus, unless otherwise stated, the scalar α_i is to be taken as the ith component of the vector a. Since the correspondence between the Greek and Latin alphabets is not perfect, some of the associations, which are listed in Appendix 1, are artificial. Particular note should be made of the association of x with ξ and y with η. As an exception to the above conventions we shall often use lower case Latin letters as subscripts and summation indices.

As was noted above, a vector x in R^2 is associated with a point in the plane whose coordinates are ξ_1 and ξ_2. Similarly, a vector x in R^3 is associated with a point in three-dimensional space whose coordinates are ξ_1, ξ_2, and ξ_3. It is customary to represent a vector graphically by drawing an arrow from the origin to the point associated with the vector (Fig. 2).

Two n-vectors are equal if and only if their corresponding components are equal. Thus the vector equality

$$a = b$$

is equivalent to the set of scalar equalities

$$\alpha_i = \beta_i \qquad (i = 1, 2, \ldots, n).$$

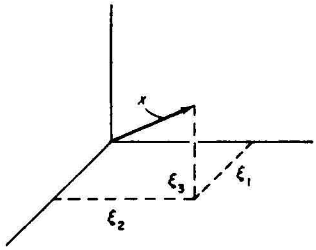

Fig. 2

In particular, to prove that two vectors are equal it is only necessary to show that their components are equal, and to define a new vector it is only necessary to specify how its components are formed.

We now turn to the first of the vector operations, the sum of two vectors.

DEFINITION 1.3. Let $a, b \in R^n$. The *sum* of a and b, written $a + b$, is the n-vector c whose components are given by

$$\gamma_i = \alpha_i + \beta_i.$$

The sum of two vectors has the following geometric interpretation. The vectors a and b form the sides of a parallelogram with one corner at the origin. The sum of a and b is then the diagonal of the parallelogram that proceeds from the origin (Fig. 3).

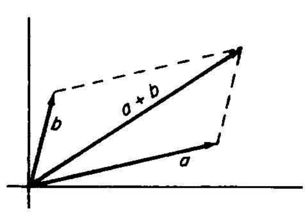

Fig. 3

The sum of two vectors has some of the properties of the usual sum of two scalars, as is shown in the following theorem.

THEOREM 1.4. Let $a, b, c \in R^n$. Then

1. $a + b = b + a$,
2. $(a + b) + c = a + (b + c)$.

PROOF. To prove part 1, let $x = a + b$ and $y = b + a$. Then by the properties of R,

$$\xi_i = \alpha_i + \beta_i = \beta_i + \alpha_i = \eta_i.$$

Hence by the above observations on the equality of vectors, $x = y$. To prove part 2, let $x = (a + b) + c$ and $y = a + (b + c)$. Then

$$\xi_i = (\alpha_i + \beta_i) + \gamma_i = \alpha_i + (\beta_i + \gamma_i) = \eta_i.$$

Hence $x = y$. ■

Property 1.4.2 says that the vector sum is an associative operation. More generally, if $a_1, a_2, \ldots, a_n \in R^n$, the sum $a_1 + a_2 + \cdots + a_n$ is the same, irrespective of the order in which sums are grouped. Likewise it follows from 1.4.1 that the sum $a_1 + a_2 + \cdots + a_n$ is unaltered when the order of the a_i's is changed. For example,

$$a_1 + a_2 + \cdots + a_k = a_k + a_{k-1} + \cdots + a_1.$$

DEFINITION 1.5. The *zero vector* in R^n is the vector whose n components are zero.

For all n we shall denote the zero vector in R^n by the same symbol "0", which is also used to denote the scalar zero. Where the meaning of the symbol "0" is not specified, it will be clear from the context what is meant.

The zero vector has some of the properties of the number zero.

THEOREM 1.6. Let $a \in R^n$. Then

1. $a + 0 = a$,
2. there is a vector, written $-a$, in R^n such that $a + (-a) = 0$.

PROOF. Let $b = a + 0$. Then

$$\beta_i = \alpha_i + 0 = \alpha_i$$

and $b = a$, which establishes part 1. For part 2, let

$$b = \begin{pmatrix} -\alpha_1 \\ -\alpha_2 \\ \vdots \\ -\alpha_n \end{pmatrix},$$

and $c = a + b$. Then

$$\gamma_i = \alpha_i + \beta_i = \alpha_i + (-\alpha_i) = 0.$$

Thus $c = 0$ and b is the vector $-a$ sought in part 2. ■

The vector $-a$ of the second part of Theorem 1.6 is simply the vector obtained by changing the signs of the components of a. Geometrically, this means that the vector a is reflected through the origin (Fig. 4).

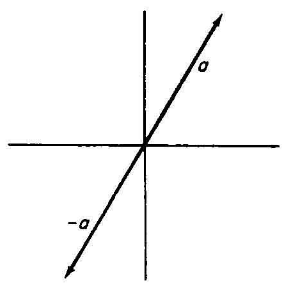

Fig. 4

The symbol "$-$" has been used for the unitary operation that, given the vector a, produces the vector $-a$ of Theorem 1.6. We shall also use the symbol to denoted the binary operation of subtraction. Specifically, given the vectors a and b, we defined their *difference* $a - b$ by

$$a - b = a + (-b).$$

It is easily verified that the difference operation satisfies the usual laws of subtraction among scalars; for example,

$$a - b = -(b - a).$$

The second important operation with vectors is the operation of scalar multiplication.

DEFINITION 1.7. Let $\lambda \in R$ and $a \in R^n$. The product of λ and a, written $\lambda \cdot a$ or λa, is the n-vector b whose components are given by

$$\beta_i = \lambda \alpha_i.$$

Geometrically, the operation of multiplying the vector a by the scalar λ

changes the length of a by a factor of $|\lambda|$. If λ is negative, a is also reflected through the origin.

THEOREM 1.8. Let $\lambda, \mu \in R$ and $a, b \in R^n$. Then

1. $(\lambda\mu)a = \lambda(\mu a)$,
2. $(\lambda + \mu)a = \lambda a + \mu a$,
3. $\lambda(a + b) = \lambda a + \lambda b$,
4. $1 \cdot a = a$.

PROOF. We shall establish property 3, leaving the rest for Exercise 1.3. In stating the theorem we have followed the usual convention of allowing multiplication to take precedence over addition, so that $\lambda a + \mu b$ means $(\lambda a) + (\mu b)$. Let $x = \lambda(a + b)$ and $y = \lambda a + \lambda b$. Then

$$\xi_i = \lambda(\alpha_i + \beta_i) = (\lambda\alpha_i) + (\lambda\beta_i) = \eta_i,$$

which establishes part 3. ■

In the above development we have defined vectors and their operations in terms of the real numbers. The properties listed in Theorems 1.4, 1.6, 1.8 are then immediate consequences of the properties of real numbers. Alternatively, we could take the properties of the theorems as axioms describing the properties of a sum and product over some set of objects $\mathcal{V}$. Specifically we call a set $\mathcal{V}$ an *abstract vector space* (over the real numbers) if

1. there is a sum "+" defined among the elements of $\mathcal{V}$ that satisfies the properties listed in Theorem 1.4,
2. there is an element "0" of $\mathcal{V}$ that satisfies the properties listed in Theorem 1.6,
3. there is a product "·" defined between the real numbers and the elements of $\mathcal{V}$ that satisfies the properties listed in Theorem 1.8.

The elements of an abstract vector space need not be n-tuples of real numbers, as the following example shows.

EXAMPLE 1.9. Let $\mathcal{V}$ be the set of all real-valued functions defined on $[0, 1]$. If $f, g \in \mathcal{V}$, define $h = f + g$ as the function whose values are $h(\xi) = f(\xi) + g(\xi)$, $\xi \in [0, 1]$. If $f \in \mathcal{V}$ and $\lambda \in R$, define $h = \lambda f$ as the function whose values are $h(\xi) = \lambda f(\xi)$. Let "0" be the function that is identically zero on $[0, 1]$. Then with these definitions, $\mathcal{V}$ is a vector space.

The properties of an abstract vector space allow us to establish theorems that are not immediately obvious. However, the proofs may be tedious. For example, the following is a proof of the theorem that if $\mathcal{V}$ is an abstract vector space and $a \in \mathcal{V}$, then $0 \cdot a = 0$. The properties that ensure each equality are listed to the side. (Incidentally, note that in the equation $0 \cdot a = 0$, the symbol "0" is used in two ways; on the left it is the scalar zero, on the right the zero vector.)

$$\begin{aligned}
0 &= a + (-a), && 1.6.2,\\
&= 1 \cdot a + (-a), && 1.8.4,\\
&= (0 + 1) \cdot a + (-a), && 1 + 0 = 1,\\
&= (0 \cdot a + 1 \cdot a) + (-a), && 1.8.2,\\
&= 0 \cdot a + [1 \cdot a + (-a)], && 1.4.2,\\
&= 0 \cdot a + [a + (-a)], && 1.8.4,\\
&= 0 \cdot a + 0, && 1.6.2,\\
&= 0 \cdot a, && 1.6.1.
\end{aligned}$$

On the other hand, to verify this fact about R^n is easy. Let $b = 0 \cdot a$. Then $\beta_i = 0 \cdot \alpha_i = 0$; hence $b = 0$. It is often the case that theorems concerning abstract vector spaces are trivialities when stated about R^n. Since in this chapter we shall be concerned exclusively with R^n, we shall not develop the theory of abstract vector spaces. Whenever a fact can be easily demonstrated by appealing to the properties of real numbers, we will use it, leaving its verification as an exercise.

EXERCISES

1. Perform the indicated calculations.

(a) $\begin{pmatrix} 1 \\ -3 \end{pmatrix} + 2\begin{pmatrix} -1 \\ 1 \end{pmatrix}$, (b) $\begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} - \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix}$,

(c) $\beta\begin{pmatrix} 1 \\ \alpha \\ \alpha^2 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$, (d) $\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \tau\begin{pmatrix} 2 \\ 10 \\ 1 \end{pmatrix}$,

(e) $\xi_1\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \xi_2\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \xi_3\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$.