Contents

INTRODUCTION vi
  Technology Support vi
  Review Materials on the Web vii

HOW TO STUDY LINEAR ALGEBRA viii

CHAPTER 1 LINEAR EQUATIONS IN LINEAR ALGEBRA
  1.1 Systems of Linear Equations 1-1
  1.2 Row Reduction and Echelon Forms 1-7
  1.3 Vector Equations 1-10
  1.4 The Matrix Equation $Ax = b$ 1-15
  1.5 Solution Sets of Linear Systems 1-19
  1.6 Applications of Linear Systems 1-24
  1.7 Linear Independence 1-28
  1.8 Introduction to Linear Transformations 1-31
  1.9 Matrix of a Linear Transformation 1-35
  1.10 Linear Models in Business, Science, and Engineering 1-39
  Supplementary Exercises 1-43
  Glossary Checklist 1-44

CHAPTER 2 MATRIX ALGEBRA
  2.1 Matrix Operations 2-1
  2.2 The Inverse of a Matrix 2-5
  2.3 Characterization of Invertible Matrices 2-9
  2.4 Partitioned Matrices 2-14
  2.5 Matrix Factorizations 2-20
  2.6 The Leontief Input-Output Model 2-28
  2.7 Applications to Computer Graphics 2-30
  2.8 Subspaces of $\mathbb{R}^n$ 2-33
  2.9 Dimension and Rank 2-38
  Supplementary Exercises 2-42
  Glossary Checklist 2-42

CHAPTER 3 DETERMINANTS
  3.1 Introduction to Determinants 3-1
  3.2 Properties of Determinants 3-5
  3.3 Cramer’s Rule, Volume and Linear Transformations 3-8
  Glossary Checklist 3-13
CHAPTER 4 VECTOR SPACES
  4.1 Vector Spaces and Subspaces 4-1
  4.2 Null Spaces, Column Spaces, and Linear Transformations 4-4
  4.3 Linearly Independent Sets; Bases 4-7
  4.4 Coordinate Systems 4-10
  4.5 The Dimension of a Vector Space 4-16
  4.6 Rank 4-18
  4.7 Change of Basis 4-23
  4.8 Applications to Difference Equations 4-27
  4.9 Applications to Markov Chains 4-32
  Glossary Checklist 4-34

CHAPTER 5 EIGENVALUES AND EIGENVECTORS
  5.1 Eigenvectors and Eigenvalues 5-1
  5.2 The Characteristic Equation 5-5
  5.3 Diagonalization 5-10
  5.4 Eigenvectors and Linear Transformations 5-15
  5.5 Complex Eigenvalues 5-19
  5.6 Discrete Dynamical Systems 5-22
  5.7 Applications to Differential Equations 5-26
  5.8 Iterative Estimates for Eigenvalues 5-28
  Glossary Checklist 5-33

CHAPTER 6 ORTHOGONALITY AND LEAST SQUARES
  6.1 Inner Product, Length, and Orthogonality 6-1
  6.2 Orthogonal Sets 6-3
  6.3 Orthogonal Projections 6-5
  6.4 The Gram-Schmidt Process 6-9
  6.5 Least-Squares Problems 6-13
  6.6 Applications to Linear Models 6-16
  6.7 Inner Product Spaces 6-20
  6.8 Applications of Inner Product Spaces 6-23
  Glossary Checklist 6-26

CHAPTER 7 SYMMETRIC MATRICES AND QUADRATIC FORMS
  7.1 Diagonalization of Symmetric Matrices 7-1
  7.2 Quadratic Forms 7-5
  7.3 Constrained Optimization 7-8
  7.4 The Singular Value Decomposition 7-10
  7.5 Applications to Image Processing and Statistics 7-16
  Supplementary Exercises 7-19
  Glossary Checklist 7-19
CHAPTER 8 THE GEOMETRY OF VECTOR SPACES

8.1 Affine Combinations  8-1
8.2 Affine Independence  8-4
8.3 Convex Combinations  8-7
8.4 Hyperplanes  8-9
8.5 Polytopes  8-12
8.6 Curves and Surfaces  8-14
    Glossary Checklist  8-16

TECHNOLOGY INDEX OF PROCEDURES AND TERMS  TECH-1

APPENDICES

INTRODUCTION TO MATLAB
    Getting Started With Matlab  ML-1
    Script M-Files  ML-4
    Index of Matlab Commands  ML-6

NOTES FOR THE MAPLE COMPUTER ALGEBRA SYSTEM
    Getting Started With Maple  MP-1
    Study Guide Notes  MP-5
    Index of Maple Commands  MP-25

NOTES FOR THE MATHEMATICA COMPUTER ALGEBRA SYSTEM
    Getting Started With Mathematica  MM-1
    Study Guide Notes  MM-5
    Index of Mathematica Commands  MM-29

NOTES FOR THE TI-83+/84+/89 GRAPHIC CALCULATORS
    Getting Started with a TI-83+ Calculator  TI-1
    Getting Started with a TI-84+ Calculator  TI-2
    Getting Started with a TI-89 Calculator  TI-2
    Study Guide Notes  TI-3
    Index of TI-84 Family Commands  TI-22
    Index of TI-89 Commands  TI-23
INTRODUCTION

This Study Guide is designed to help you succeed in your linear algebra course. It shows you how to study mathematics, to learn new material, and to prepare effective review sheets for tests. Key Ideas and Study Notes guide you through each section, with summaries of important ideas and tables that connect related ideas. Detailed solutions to hundreds of exercises (usually every third odd exercise) allow you to check your work or help you get started on a difficult problem. Also, complete explanations are provided for each writing exercise whose answer in the text is only a “Hint”. Study Tips point out important exercises, give hints about what to study, and sometimes highlight potential exam questions. Frequent Warnings identify common student errors. Don’t ever take an exam without reviewing these warnings!

The most important material in this Study Guide is on pages vii through ix of this introduction. Students who follow the strategies in How to Study Linear Algebra invariably achieve remarkable results in this course. You can be one of those students.

TECHNOLOGY SUPPORT

If you are using technology with your course, you will need this Study Guide. Besides its valuable support for the course material, the Guide includes “Lab Manuals” for three computer programs and three graphic calculators. Everything you need to know about using this technology with your text is here. New commands are introduced gradually, and detailed instructions are given for their use. Also, data for more than 850 exercises from the text are stored in electronic files for each type of technology. You’ll save hours of time and avoid errors in typing. The files also contain special programs that reinforce basic concepts in the course. If your class is using MATLAB, Maple, or Mathematica, your files are probably already loaded on the school computer system or in specified labs. If you are using a TI-83+, TI-84+, or TI-89, your instructor may have plans to download the files to your calculator. In any case, you can always download the files yourself from the Web site:

www.pearsonhighered.com/lay

A ReadMe file with each data set describes how to incorporate the data into your software or load it into your calculator. The files on the Web will always reflect the latest versions available.

Special MATLAB boxes at the ends of many Study Guide sections explain how to use MATLAB for your homework, introducing simple commands as they are needed for the exercises. The first appendix at the back of the Guide contains a quick introduction, Getting Started with MATLAB, and an index of MATLAB commands. I encourage you to try MATLAB—it is easy to learn.

Notes for the Maple and Mathematica computer algebra systems and for the TI-83+/84+/89 graphic calculators are included in the last three appendices to this Study Guide. The notes correspond to the MATLAB boxes and translate MATLAB commands into syntax appropriate for the other technologies. Each set of notes has its own index. Whenever you see a MATLAB box in this Guide, turn to the appropriate appendix for help with your technology.
The following faculty members wrote the notes and developed the data sets:

- Maple: Douglas Meade, University of South Carolina, Columbia, SC
- Mathematica: Marie Vanisko, Carroll College, MT
- TI-83+/84+/89: Michael Miller, Corban University, Salem, OR
- MATLAB: Jeremy Case, Taylor University, Marion, IN

These colleagues have given good advice based on teaching with our text and Study Guide. I appreciate their contributions to this revision of the Study Guide.

**REVIEW MATERIALS ON THE WEB**

In addition to the help in this Study Guide, I have provided some material on the Web that my students really appreciate—review sheets and practice exams. Please heed the advice below, because using these study aids in the wrong way can lead to a disaster at exam time. I suggest four steps to prepare for each exam.

1. **Assemble a review sheet.** The Web review sheets reflect what I emphasize in my courses. If possible, you need to find out what your instructor expects of you. The three courses on the Web material vary somewhat in their content and approach, and they organize the material differently. To construct a review sheet for one of your exams, you may need to combine parts of two sheets from the Web.

2. **Try to complete an initial review for an exam a day or two early.** Hard to do, but worth the effort. Don’t read the sample exams! Studying from an old exam is a big mistake. Instead, study your lecture notes, looking for items that were emphasized in class. Read over your homework and old quizzes. I insist that my students learn key definitions, practically word for word. If they cannot write a definition properly, they don’t know what they are writing about.

3. **After the review, take the sample exam whose subject material most closely fits what your exam will cover.** Find a quiet place and time when you can work the entire exam without stopping and without looking at the text or your notes. (Of course, skip any questions that are inappropriate for your course.) Write your answers. Time yourself.

4. **Finally after you have completed the test, look at the solutions.** Identify the areas that need further review. If possible, find an example in the text related to an area in which you are weak. Cover up the solution to the example, and try to write out the solution yourself. Peek, if necessary. Don’t be reluctant to ask for help.
A first course in linear algebra is dramatically different from most mathematics courses that pre-
ceed it. The focus shifts from learning computational procedures to digesting and mastering basic
concepts that underlie the computations. To survive, you may need to learn a new way to study
mathematics. That’s why I wrote this Study Guide—to show you how to succeed in the course and
to give you tools to do this.

Because you are likely to use linear algebra later in your career, you need to learn the
material at a level that will carry you far beyond the final exam. I believe that the strategies
below are crucial to success.

Strategies for Success in Linear Algebra

1. **Study before you start to work on exercises.** Most students don’t do this in courses that
precede linear algebra. They survive by looking at the examples when they cannot solve an
exercise. That simply will not work in linear algebra. If you “copy” an example (with neces-
sary modifications), you may think you understand the problem, but very little true learning
has taken place. (You’ll find that out on your first exam.) For this course, in addition to
knowing how to carry out a certain procedure, you must learn when that procedure is appro-
priate and (most importantly) why it works.

   For success with homework, read the text section first, perhaps taking a few notes. Then,
   read the Key Ideas or Study Notes in the Study Guide for that section. Finally, start to work
   on the assigned exercises. In the long run, this approach will improve your performance and
   save you time. The preparation time spent here will greatly reduce your exam preparation
time.

2. **Prepare for each class period as you would for a language class.** Mastery of the subject
requires that you learn a rich vocabulary. Your goal now is to become so familiar with con-
cepts that you can use them easily (and correctly) in conversation and in writing. For home-
work, try to write complete sentences, such as you’ll find the Study Guide solutions. Pay
attention, too, to the warnings here about misuse of terminology.

   This course resembles a language course because of the preparation needed between class
meetings, to avoid falling behind. Most sections in the text build on preceding sections.
Once you are behind, catching up with the class is often difficult. The fact that concepts may
seem “simple” does not mean that you can afford to postpone your study until the weekend.
The homework may be harder than you expect. The most valuable advice I can give is to
keep up with the course.

3. **Concentrate more on learning definitions, facts, and concepts,** than on practicing routine
computations or algorithms. Pay attention to connections between concepts. Many theorems
and boxed “facts” describe such connections. For examples, see Theorem 2 in Section 1.2,
and Theorems 3 and 4 in Section 1.4. Your goal is to think in general terms, to imagine
typical computations without performing any arithmetic, and to focus on the principles
behind the computations.
4. **Review frequently.** Review and reflection are key ingredients for success in learning the material. At strategic points in this *Guide*, I have inserted special subsections labeled “Mastering Linear Algebra Concepts.” They provide specific help for your review of each main concept. I urge you to prepare the review sheets described as you reach each review point. Later, you may choose to add further notes. Of course, use the sheets to review for exams. A Glossary Checklist at the end of each chapter in the *Guide* may help you learn important definitions.

**CAUTION** Because you can find complete solutions here to many exercises, you will be tempted to read the explanations before you really try to write out the solutions yourself. Don’t do it! If you merely think a bit about a problem and then check to see if your idea is basically correct, you are likely to overestimate your understanding. Some of my students have done this and miserably failed the first exam. By then the damage was done, and they had great difficulty catching up with the class. Proper use of the *Study Guide*, however, will help you succeed and enjoy the course at the same time.

**A Personal Note**

Students who have used this material have told me how much it helped them learn linear algebra and prepare for tests. The first time my students used the *Study Guide* notes, they had already taken one exam. Grades on the next exam were substantially higher. For some students, the improvement was dramatic. I hope the *Study Guide* will encourage you to master linear algebra and to perform at a level higher than you ever dreamed possible.

David C. Lay
As you work through this chapter and the next, your experience may resemble several walks through a village at different seasons of the year. The surroundings will be familiar, but the landscape will change. You will examine various mathematical concepts from several points of view, and a major problem will be to learn all the new terminology and the many connections between the concepts. In Chapter 4, you will see these ideas in a more abstract setting. Diligent work now will make the trip through Chapter 4 just another walk through the same village.

1.1 SYSTEMS OF LINEAR EQUATIONS

The fundamental concepts presented in this section and the next must be mastered for they will be used throughout the course.

STUDY NOTES

Please read How to Study Linear Algebra, on the preceding two pages, before you continue.

The text uses boldface type to identify important terms the first time they appear. You need to learn them; some students write selected terms on 3 × 5 cards, for review. At the end of each chapter in this Study Guide, a glossary checklist may help you learn definitions.

The text defines the size of a matrix. Don’t use the term dimension, even though that appears in some computer programming languages, because in linear algebra, dimension refers to another concept (in Section 4.5).

The first few examples are so simple that they could be solved by a variety of techniques. But it is important to learn the systematic method presented here, because it easily handles more complicated linear systems, and it works in all cases.

The calculations in this section are based on the following important fact:

*When elementary row operations are applied to a linear system, the new system has exactly the same solution set.*
(See the text.) The steps in the summary below will be modified slightly in Section 1.2.

Summary of the Elimination Method (for This Section)

1. The first equation must contain an \( x_1 \). Interchange equations, if necessary. This will create a nonzero entry in the first row, first column, of the augmented matrix.
2. Eliminate \( x_1 \) terms in the other equations. That is, use replacement operations to create zeros in the first column of the matrix below the first row.
3. Obtain an \( x_2 \) term in the second equation. (Interchange the second equation with one below, if needed, but don’t touch the first equation.) You may scale the second equation, if desired, to create a 1 in the second column and second row of the matrix.
4. Eliminate \( x_2 \) terms in equations below the second equation, using replacement operations.
5. Continue with \( x_3 \) in the third equation, \( x_4 \) in the fourth equation, etc., eliminating these variables in the equations below. This will produce a “triangular” system (at least for systems in this section).
6. Check if the system in triangular form is consistent. If it is, a solution is found by starting with the last nonzero equation and working back up to the first equation. Each variable on the “diagonal” is used to eliminate the terms in that variable above it. The solution to the system becomes apparent when the system is finally transformed into “diagonal” form.
7. Check any solutions you find by substituting them into the original system.

The solution set of a system of linear equations either is empty, or contains one solution, or contains infinitely many solutions. When asked to “solve” a system, you may write “inconsistent” if the system has no solution.

As you will see later, determining the number of solutions in the solution set is sometimes more important than actually computing the solution or solutions. For that reason, pay close attention to the subsection on existence and uniqueness questions. Key Exercises: 19–22 and 25.

SOLUTIONS TO EXERCISES

Get into the habit now of working the Practice Problems before you start the exercises. Probably, you should attempt all the Practice Problems before checking the solutions at the end of the exercise set, because once you start reading the first solution, you might tend to read on through the other solutions and spoil your chance to benefit from those problems.

For brevity, the symbols R1, R2, . . ., stand for row 1 (or equation 1), row 2 (or equation 2), and so on.
1. \[x_1 + 5x_2 = 7\]
\[-2x_1 - 7x_2 = -5\]

\[
\begin{bmatrix}
1 & 5 & 7 \\
-2 & -7 & -5
\end{bmatrix}
\]

Replace R2 by R2 + (2)R1:

\[
\begin{bmatrix}
1 & 5 & 7 \\
3x_2 = 9 & 0 & 3 & 9
\end{bmatrix}
\]

Scale R2 by 1/3:

\[
\begin{bmatrix}
1 & 5 & 7 \\
x_2 = 3 & 0 & 1 & 3
\end{bmatrix}
\]

Replace R1 by R1 + (–5)R2:

\[
\begin{bmatrix}
x_1 = -8 \\
x_2 = 3
\end{bmatrix}
\]

The solution is \((x_1, x_2) = (-8, 3)\), or simply \((-8, 3)\). Check:

\[
\begin{align*}
-8 + 5(3) &= -8 + 15 = 7 \\
-2(-8) - 7(3) &= 16 - 21 = -5
\end{align*}
\]

7. \[
\begin{bmatrix}
1 & 7 & 3 & -4 \\
0 & 1 & -1 & 3 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & -2
\end{bmatrix}
\]

Ordinarily, the next step would be to interchange R3 and R4, to put a 1 in the third row and third column. But in this case, the third row of the augmented matrix corresponds to the equation \(0x_1 + 0x_2 + 0x_3 = 1\), or simply, \(0 = 1\). A system containing this condition has no solution. Further row operations are unnecessary once an equation such as \(0 = 1\) is evident.

The solution set is empty.

**Study Tip:** When writing a coefficient matrix or augmented matrix for a system of linear equations, be sure that the variables appear in the same order in each equation. Arrange the variables in columns, as in the text, placing zeros in the matrix whenever a variable is missing from an equation.

13. \[
\begin{align*}
\begin{bmatrix}
1 & 0 & -3 & 8 \\
2 & 2 & 9 & 7
\end{bmatrix} & \sim \begin{bmatrix}
1 & 0 & -3 & 8 \\
0 & 2 & 15 & -9
\end{bmatrix} & \sim \begin{bmatrix}
1 & 0 & -3 & 8 \\
0 & 1 & 5 & -2
\end{bmatrix} & \sim \begin{bmatrix}
1 & 0 & -3 & 8 \\
0 & 2 & 15 & -9
\end{bmatrix}
\end{align*}
\]

\[
\begin{align*}
\begin{bmatrix}
1 & 0 & -3 & 8 \\
0 & 1 & 5 & -2
\end{bmatrix} & \sim \begin{bmatrix}
1 & 0 & 0 & 5 \\
0 & 1 & 0 & 3
\end{bmatrix} & \sim \begin{bmatrix}
1 & 0 & 0 & 5 \\
0 & 1 & 0 & 3
\end{bmatrix}
\end{align*}
\]

The solution is \((5, 3, -1)\).

**Study Tip:** Pay attention to how a problem is worded. If you only need to determine the existence or uniqueness of a solution, you can stop row operations when you reach a “triangular” form. Exercises 15–18 do not require you to solve the systems of equation.
19. \[
\begin{bmatrix}
1 & h & 4 \\
3 & 6 & 8 \\
\end{bmatrix}
\sim
\begin{bmatrix}
1 & h & 4 \\
0 & 6-3h & -4 \\
\end{bmatrix}
\]
Think of \(6 - 3h\) as a constant, \(c\). When \(c\) is zero, that is, when \(h = 2\), the system has no solution, because \(0 x_2 = -4\) has no solution. Otherwise, when \(c\) is nonzero, that is, when \(h \neq 2\), the system has a solution.

23. My own students have recommended that I never give the complete answers to the true/false questions. They felt that the temptation to read the answers is too great. After working both with and without answers, they realized how much they benefited from doing the true/false work by themselves. So, all you will see here are the places where you can find the answers.

a. See the remarks following the box titled “Elementary Row Operations.”

b. The size of a matrix is defined just before the subsection titled “Solving a Linear System.”

c. The solution set of a linear system is the set of all solutions of the system. See page 3.

d. See the box before Example 2.

25. \[
\begin{bmatrix}
1 & -4 & 7 & g \\
0 & 3 & -5 & h \\
-2 & 5 & -9 & k \\
\end{bmatrix}
\sim
\begin{bmatrix}
1 & -4 & 7 & g \\
0 & 3 & -5 & h \\
0 & -3 & 5 & k + 2g \Rightarrow k + 2g + h \\
\end{bmatrix}
\]
Let \(b\) denote the number \(k + 2g + h\). Then the third equation represented by the augmented matrix above is \(0 x_3 = b\). If \(b\) is nonzero, this equation has no solution, so the system is inconsistent. The system is consistent if \(b\) is zero, that is, if \(k + 2g + h = 0\), then the system has a solution no matter what the values of \(g\) and \(h\). The text will explore this situation more in Section 1.2. Briefly, here is why this system, and hence the original system, is consistent. In this case, the third equation can be ignored, and the second equation, \(3x_2 - 5x_3 = h\) has many solutions. Imagine choosing any values for \(x_2\) and \(x_3\) that satisfy the second equation, and substituting those values for \(x_2\) and \(x_3\) in the first equation. The resulting first equation can be solved for \(x_1\). These values for \(x_1, x_2,\) and \(x_3\) will satisfy all three equations.

31. Look at the first column. The next row operation should replace the 4 in the third row by a 0. To do this, replace \(R3\) by \(R3 + (-4)R1\). To reverse the operation, replace \(R3\) by \(R3 + 4R1\).
A Mathematical Note: “If . . . , then . . . .”

Many important facts and theorems in the text are written as implication statements, in the form “If $P$, then $Q$”, where $P$ and $Q$ represent complete sentences. For instance, the statement in the box at the top of page 8 has the form

\[
\begin{align*}
\text{If } & \begin{cases} 
\text{the augmented matrices} \\
\text{of two linear systems} \\
\text{are row equivalent}
\end{cases} \text{, then } \begin{cases} 
\text{the two systems} \\
\text{have the same} \\
\text{solution set}
\end{cases} \\
\end{align*}
\]

(1)

An implication statement “If $P$, then $Q$” is itself true provided that statement $Q$ is true whenever statement $P$ is true. In mathematical terminology, we say that “$P$ implies $Q$”, and we write $P \implies Q$.

Be careful to distinguish between an implication statement “$P$ implies $Q$” and the converse or “opposite” implication, “$Q$ implies $P$”. The converse may or may not be true when the original implication is true. For instance, the converse of (1) above is not true, because there exist two linear systems with the same solution set but whose augmented matrices are not row equivalent. For example:

\[
\begin{align*}
x_1 + x_2 &= 1 \\
2x_1 + x_2 &= 2 \\
3x_1 + 3x_2 &= 3
\end{align*}
\]

\[
\begin{align*}
x_1 + x_2 &= 1 \\
2x_1 + 2x_2 &= 2
\end{align*}
\]

MATLAB Row Operations

To use MATLAB for your homework in this course, the MATLAB program must contain the data for the exercises in the fifth edition of the text. At some schools, the campus-wide version of MATLAB already has this data available on some or all computers. (The same may be true for Maple or Mathematica.) Ask your instructor. If you plan to run MATLAB at home, you will need to download the MATLAB Laydata5 Toolbox from the companion CD or from the website

www.pearsonhighered.com/lay

and follow the instructions there. Data files are also available at this site for Maple, Mathematica, and the graphic calculators TI-83+/84+/89. Basic instructions for using these matrix programs in this course are given in appendices at the end of this Study Guide. Specific commands for MATLAB will be introduced as needed at the end of some sections. Corresponding commands for other matrix programs can be found in the appendices.

While you are running MATLAB, type the command \texttt{c1s1} (which stands for chapter 1 section 1) at the MATLAB prompt. If the data are not installed, you will get a message such as “Undefined function”. Otherwise, you should see a list of exercises in Section 1.1 for which data are available. Type the number of the appropriate exercise and press <Enter>.
For Section 1.1, the MATLAB data for each exercise is stored in a matrix called $M$. You can perform row operations on $M$ with the following commands (which are in the Laydata5 Toolbox along with the data):

- `replace(M,r,m,s)` replaces row $r$ of matrix $M$ by row $r + m$-row $s$
- `swap(M,r,s)` interchanges rows $r$ and $s$ of $M$
- `scale(M,r,c)` multiplies row $r$ of $M$ by a nonzero scalar $c$

(Press <Enter> after each MATLAB command, displayed in boldface type.) The name of any matrix in your MATLAB workspace can be inserted in place of $M$; the letters $r$, $m$, $s$, and $c$ stand for any numbers you choose. Note that $r$ and $s$ should be whole numbers.

If you enter one of these commands, say `swap(M,1,3)`, then the new matrix, produced from $M$, is stored in the matrix “ans” (for “answer”). If, instead, you type $M1 = swap(M,1,3)$, then the answer is stored in a new matrix $M1$. If the next operation is $M2 = replace(M1,2,5,1)$, then the result of changing $M1$ is placed in $M2$, and so on.

The advantage of giving a new name to each new matrix is that you can easily go back a step if you don’t like what you just did to a matrix. If, instead, you type $M = replace(M,2,5,1)$, then the result is placed back in $M$ and the “old” $M$ is lost. Of course, the “reverse” operation, $M = replace(M,2,-5,1)$ will bring back the old $M$.

Note: For the simple problems in this section and the next, the multiple $m$ you need in the command `replace(M,r,m,s)` will usually be a small integer or fraction that you can compute in your head. In general, $m$ may not be so easy to compute mentally. The next two paragraphs describe how to handle such a case.

The entry in row $r$ and column $c$ of a matrix $M$ is denoted by $M(r,c)$. If the number stored in $M(r,c)$ is displayed with a decimal point, then the displayed value may be accurate to only about five digits. In this case, use the symbol $M(r,c)$ instead of the displayed value in calculations.

For instance, if you want to use the entry $M(s,c)$ to change $M(r,c)$ to 0, enter the commands

```
m = -M(r,c)/M(s,c)
M = replace(M,r,m,s)
```

Or, you can use just one command: $M = replace(M,r,-M(r,c)/M(s,c),s)$.

Finally, the command `format compact` will eliminate extra space between displays, so you can see more data on the screen. The command `format` will return the screen to the normal display.

**Warning:** Using a matrix program such as MATLAB is fun and will save you time, but make sure you can perform row operations rapidly and accurately with pencil and paper. Probably, you should work all the exercises in Section 1.1 by hand and use your matrix program only to check your work.
1.2 ROW REDUCTION AND ECHELON FORMS

Our interest in the row reduction algorithm lies mostly in the echelon forms that are created by the algorithm. For practical work, a computer should perform the calculations. However, you need to understand the algorithm so you can learn how to use it for various tasks. Also, unless you take your exams at a computer or with a matrix programmable calculator, you must be able to perform row reduction quickly and accurately by hand.

STUDY NOTES

The row reduction algorithm applies to any matrix, not just an augmented matrix for a linear system. In many cases, all you need is an echelon form. The reduced echelon form is mainly used when it comes from an augmented matrix and you have to find all the solutions of a linear system.

Strategies for faster and more accurate row reduction:

- Avoid subtraction in a row replacement. It leads to mistakes in arithmetic. Instead, add a negative multiple of one row to another.
- Always enclose each matrix with brackets or large parentheses.
- To save time, combine all row replacement operations that use the same pivot position, and write just one new matrix. Never “clean out” more than one column at a time. (You can combine several scaling operations, or combine several interchanges, if you are careful. But that seldom will be necessary.)
- Never combine an interchange with a replacement. In general, don’t combine different types of row operations. This will be particularly important when you evaluate determinants, in Chapters 3 and 5.

How to avoid copying errors:

- Practice neat writing, not too small. Develop proper habits in homework so your work on tests will be accurate, complete, and readable.
- Write a matrix row by row. Your eye may be less likely to read from the wrong row if you place the new matrix beside the old one. Arrange your sequence of matrices across the page, rather than down the page. (Some students prefer to place the matrices in columns. Use whichever method seems to work best for you.)
- Try not to let your work flow from one side of a paper to the reverse side.

**Study Tips:** Theorem 2 is a key result for future work. Also, study the procedure in the box following Theorem 2. Failure to write out the system of equations (step 4) is a common source of errors.
SOLUTIONS TO EXERCISES

1. To check whether a matrix is in echelon form, ask the questions:
   (i) Is every nonzero row above the all-zero rows (if any)?
   The matrix (c) fails this test, so it is not in echelon form.
   (ii) Are the leading entries in a stair-step pattern, with zeros below each leading entry?
   The matrices (a), (b), and (d) all pass tests (i) and (ii), so they are in echelon form.

To check whether a matrix in echelon form is actually in reduced echelon form, ask two more questions:
   (iii) Is there a 1 in every pivot position?
   Matrix (d) fails this test, so it is only in echelon form. Finally, ask:
   (iv) Is each leading 1 the only nonzero entry in its column?
   Matrices (a) and (b) pass all four tests, so they are in reduced echelon form.

Study Tip: Exercises 5 and 6 ask you to “visualize” echelon forms and write out matrices whose entries are just symbols. Example 2 suggests the form of your “answer,” but it does not show you how to find the answer. Later, other exercises will ask you to construct other types of examples. If you look at answers from the text, or the Study Guide (or another student), before you try to write your own answers, you will lose most of the value of such exercises. The process of trying to understand the question and writing an example is important.

7. \[
\begin{bmatrix}
1 & 3 & 4 & 7 \\
3 & 9 & 7 & 6
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 3 & 4 & 7 \\
0 & 0 & -5 & -15
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 3 & 4 & 7 \\
0 & 0 & 1 & 3
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 3 & 0 & -5 \\
0 & 0 & 1 & 3
\end{bmatrix}
\]

Corresponding system of equations:
\[x_1 + 3x_2 = -5\]
\[x_3 = 3\]

The basic variables (corresponding to the pivot positions) are \(x_1\) and \(x_3\). The remaining variable \(x_2\) is free. Solve for the basic variables in terms of the free variable. The general solution is
\[
\begin{align*}
\begin{cases}
x_1 = -5 - 3x_2 \\
x_2 \text{ is free} \\
x_3 = 3
\end{cases}
\end{align*}
\]

13. \[
\begin{bmatrix}
1 & -3 & 0 & -1 & 0 & -2 \\
0 & 1 & 0 & 0 & -4 & 1 \\
0 & 0 & 0 & 1 & 9 & 4 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & -3 & 0 & 0 & 9 & 2 \\
0 & 1 & 0 & 0 & -4 & 1 \\
0 & 0 & 0 & 1 & 9 & 4 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & -4 & 1 \\
0 & 0 & 0 & 1 & 9 & 4 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

Copyright © 2016 Pearson Education, Inc.
\[
\begin{align*}
    x_1 - 3x_3 &= 5 \\
    x_2 - 4x_5 &= 1 \\
    x_4 + 9x_5 &= 4 \\
    0 &= 0
\end{align*}
\]

Corresponding system:

\[
\begin{aligned}
    x_1 &= 5 + 3x_5 \\
    x_2 &= 1 + 4x_5 \\
    x_4 &= 4 - 9x_5 \\
    x_5 &\text{ is free}
\end{aligned}
\]

Basic variables: \(x_1, x_2, x_4\); free variables: \(x_3, x_5\). General solution:

Note: A common error in this exercise is to assume that \(x_3\) is zero. Another common error is to say nothing about \(x_3\) and write only \(x_1, x_2, x_4,\) and \(x_5,\) as above. To avoid these mistakes, identify the basic variables first. Any remaining variables are free. (This type of computation will arise in Chapter 5.) See also Exercise 8.

Study Tip: Be sure to work Exercises 17–20. The experience will help you later. These exercises make nice quiz questions, too.

19. \begin{bmatrix} 1 & h & 2 \\ 4 & 8 & k \end{bmatrix} \sim \begin{bmatrix} 1 & h & 2 \\ 0 & 8 - 4h & k - 8 \end{bmatrix}. \]

Look first at \(8 - 4h.\) If this number is not zero, then the system must be consistent. Also, the solution will be unique because there are no free variables. This is case (b), when \(h \neq 2.\) Now, if \(8 - 4h\) is zero, that is, if \(h = 2,\) there are two possibilities—either \(k\) equals 8 or \(k\) does not equal 8. If \(h = 2\) and \(k = 8,\) the second equation is \(0x_3 = 0.\) The system is consistent and has a free variable, so the system has infinitely many solutions. This is case (c). When \(h = 2\) and \(k \neq 8,\) the second equation is \(0x_2 = b,\) with \(b\) nonzero, and the system has no solution. This is case (a).


b. See the second paragraph of the section.

c. Basic variables are defined after equation (4).

d. See the beginning of the subsection, “Parametric Descriptions of Solution Sets.” Actually, this question does not consider the case of an inconsistent system. A better true/false statement would be: “If a linear system is consistent, then finding a parametric description of the solution set is the same as solving the system.

e. The row shown corresponds to the equation \(5x_4 = 0.\) Could there also be an equation of the form \(0x_4 = b,\) with \(b\) nonzero?

25. A full solution is in the text answer section.

Study Tip: Notice from Exercise 27 that the question of uniqueness of the solution of a linear system is not influenced by the numbers in the rightmost column of the augmented matrix.
31. Yes, a system of linear equations with more equations than unknowns can be consistent. The answer in the text includes an example.

34. The data for this exercise comes from one of my students who was working part time for a private wind tunnel company near the University of Maryland. You will need a matrix program to solve this problem. The basic instructions for MATLAB were given in the Study Guide notes for Section 1.1. For Maple, Mathematica, the TI-calculators, see the respective appendices at the end of this Study Guide.

A Mathematical Note: “If and only if”

You need to know what the phrase “if and only if” means. It was used above in Exercise 27, and you will see it again in theorems and in boxed facts. The phrase “if and only if” always appears between two complete statements. Look at Theorem 2, for instance:

\[
\begin{align*}
\{ & \text{A specific} \\
\{ & \text{linear system} \\
\} \text{ is consistent} & \text{ if and only if } \\
\{ & \text{the rightmost column of the} \\
\{ & \text{augmented matrix is not} \\
\} & \text{a pivot column.} \\
\end{align*}
\]

(1)

The entire sentence means that the two statements in parentheses are either both true or both false.

Sentence (1) has the general form

\[
P \text{ if and only if } Q
\]

(2)

where \( P \) denotes the first statement and \( Q \) denotes the second statement. This sentence says two things:

If statement \( P \) is true, then statement \( Q \) is also true.
If statement \( Q \) is true, then statement \( P \) is also true.

A mathematical shorthand for (2) is “\( P \iff Q \)”. 

1.3 VECTOR EQUATIONS

Do not be deceived by the rather simple beginning of Section 1.3. The important material on \( \text{Span}\{v_1, \ldots, v_p\} \) will take time to digest. Figures 8, 10, and 11 are important, along with Exercises 11–14, 17, 18, 25, and 26. Each of the exercises involves an existence question about whether a certain vector equation has a solution. (You don’t have to find the solution.) Notice how the same basic question can be asked in several different ways.
STUDY NOTES

Develop the habit of reading the section carefully once or twice before looking at the Study Guide and before starting the exercises. (Don’t just look at the pictures and examples! Important comments lurk in between.)

In nearly all of the text, a scalar is just a real number. By convention, scalars are usually written to the left of vectors, such as 5v or cv, rather than v5 or vc. To identify vectors in your lecture notes and homework, you should pay attention to context. (Some instructors may ask you to underline or put arrows over vectors, but many students find this a distraction.)

Vectors must be the same size to be added or used in a linear combination. For instance, a vector in $\mathbb{R}^3$ cannot be added to a vector in $\mathbb{R}^2$.

SOLUTIONS TO EXERCISES

1. $\mathbf{u} + \mathbf{v} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} + \begin{bmatrix} -3 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 + (-3) \\ 2 + (-1) \end{bmatrix} = \begin{bmatrix} -4 \\ 1 \end{bmatrix}$.

Using the definitions carefully,

$\mathbf{u} - 2\mathbf{v} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} + (-2) \begin{bmatrix} -3 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \end{bmatrix} + \begin{bmatrix} (-2)(-3) \\ (-2)(-1) \end{bmatrix} = \begin{bmatrix} -1 + 6 \\ 2 + 2 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}$, or, more quickly,

$\mathbf{u} - 2\mathbf{v} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} - 2 \begin{bmatrix} -3 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 + 6 \\ 2 + 2 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}$. The intermediate step is often not written.

7. See the figure below. Since the grid can be extended in every direction, the figure suggests that every vector in $\mathbb{R}^2$ can be written as a linear combination of $\mathbf{u}$ and $\mathbf{v}$. To write a vector $\mathbf{a}$ as a linear combination of $\mathbf{u}$ and $\mathbf{v}$, imagine walking from the origin to $\mathbf{a}$ along the grid “streets” and keep track of how many “blocks” you travel in the $\mathbf{u}$-direction and how many in the $\mathbf{v}$-direction

![Vector Diagram]

a. To reach $\mathbf{a}$ from the origin, you might travel 1 unit in the $\mathbf{u}$-direction and −2 units in the $\mathbf{v}$-direction (that is, 2 units in the negative $\mathbf{v}$-direction). Hence $\mathbf{a} = \mathbf{u} - 2\mathbf{v}$.

b. To reach $\mathbf{b}$ from the origin, travel 2 units in the $\mathbf{u}$-direction and −2 units in the $\mathbf{v}$-direction. So $\mathbf{b} = 2\mathbf{u} - 2\mathbf{v}$. Or, use the fact that $\mathbf{b}$ is 1 unit in the $\mathbf{u}$-direction from $\mathbf{a}$, so that
The vector \( c \) is 1.5 units from \( b \) in the \( v \)-direction, so

\[
\begin{align*}
\mathbf{c} &= \mathbf{b} - 1.5\mathbf{v} = (2\mathbf{u} - 2\mathbf{v}) - 1.5\mathbf{v} = 2\mathbf{u} - 3.5\mathbf{v}
\end{align*}
\]

d. The “map” suggests that you can reach \( d \) if you travel 3 units in the \( u \)-direction and –4 units in the \( v \)-direction. If you prefer to stay on the paths displayed on the map, you might travel from the origin to –3\( \mathbf{v} \), then move 3 units in the \( u \)-direction, and finally move –1 unit in the \( v \)-direction. So

\[
\mathbf{d} = -3\mathbf{v} + 3\mathbf{u} - \mathbf{v} = 3\mathbf{u} - 4\mathbf{v}
\]

Another solution is

\[
\mathbf{d} = \mathbf{b} - 2\mathbf{v} + \mathbf{u} = (2\mathbf{u} - 2\mathbf{v}) - 2\mathbf{v} + \mathbf{u} = 3\mathbf{u} - 4\mathbf{v}
\]

9. Here are the intermediate calculations, which usually are not displayed. Check with your instructor whether you need to “show work” on a problem such as this.

\[
\begin{align*}
\begin{bmatrix}
x_2 + 5x_3 \\
4x_1 + 6x_2 - x_3 \\
-x_1 + 3x_2 - 8x_3
\end{bmatrix}
&= 0, &
\begin{bmatrix}
x_2 + 5x_3 \\
4x_1 + 6x_2 - x_3 \\
-x_1 + 3x_2 - 8x_3
\end{bmatrix}
&= 0,
\end{align*}
\]

Helpful Hint: As you work Exercises 11–14, circle the pivots in an echelon form of an appropriate matrix. This will help you visualize the cases when a vector either is or is not a linear combination of other vectors.

13. Denote the columns of \( A \) by \( \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \). To determine if \( \mathbf{b} \) is a linear combination of these columns, use the boxed fact on page 36. Row reduce the augmented matrix until you reach echelon form:

\[
\begin{bmatrix}
1 & -4 & 2 & 3 \\
0 & 3 & 5 & -7 \\
-2 & 8 & -4 & -3
\end{bmatrix}
\sim
\begin{bmatrix}
1 & -4 & 2 & 3 \\
0 & 3 & 5 & -7 \\
0 & 0 & 0 & 3
\end{bmatrix}
\]

The system for this augmented matrix is inconsistent, so \( \mathbf{b} \) is not a linear combination of the columns of \( A \).

19. By inspection, \( \mathbf{v}_2 = (3/2)\mathbf{v}_1 \). Any linear combination of \( \mathbf{v}_1 \) and \( \mathbf{v}_2 \) is actually just a multiple of \( \mathbf{v}_1 \). For instance,

\[
\mathbf{a}\mathbf{v}_1 + \mathbf{b}\mathbf{v}_2 = \mathbf{a}\mathbf{v}_1 + \mathbf{b}(3/2)\mathbf{v}_1 = (\mathbf{a} + 3\mathbf{b}/2)\mathbf{v}_1
\]

So \( \text{Span}\{\mathbf{v}_1, \mathbf{v}_2\} \) is the set of points on the line through \( \mathbf{v}_1 \) and \( \mathbf{0} \).
Warning: Although Figures 8 and 11 provide the most common ways to view \( \text{Span}\{\mathbf{u}, \mathbf{v}\} \), don’t forget Exercise 19, which shows that in a special case, \( \text{Span}\{\mathbf{u}, \mathbf{v}\} \) can be just a line through the origin. In fact, \( \text{Span}\{\mathbf{u}, \mathbf{v}\} \) can also be just the origin itself. How?

21. Let \( y = \begin{bmatrix} h \\ k \end{bmatrix} \). Then \( [\mathbf{u} \ \mathbf{v} \ y] = \begin{bmatrix} 2 & 2 & h \\ -1 & 1 & k \end{bmatrix} \sim \begin{bmatrix} 2 & 2 & h \\ 0 & 2 & k + h / 2 \end{bmatrix} \). This augmented matrix corresponds to a consistent system for all \( h \) and \( k \). So \( y \) is in \( \text{Span}\{\mathbf{u}, \mathbf{v}\} \) for all \( h \) and \( k \).

23. a. The alternative notation for a (column) vector is discussed after Example 1.
   
   b. Plot the points to check the assertion. Or, see the statement preceding Example 3.
   
   c. See the line displayed just before Example 4.
   
   d. See the box that discusses the matrix in (5).
   
   e. Read the geometric description of \( \text{Span}\{\mathbf{u}, \mathbf{v}\} \) very carefully.

Study Tip: I urge my own students to work by themselves on the true/false questions and then meet together in groups of two or three, to compare and discuss their answers.

25. a. There are only three vectors in the set \( \{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\} \), and \( \mathbf{b} \) is not one of them.
   
   b. There are infinitely many vectors in \( W = \text{Span}\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\} \). To determine if \( \mathbf{b} \) is in \( W \), use the method of Exercise 13.

   \[
   \begin{bmatrix}
   1 & 0 & -4 & 4 \\
   0 & 3 & -2 & 1 \\
   -2 & 6 & 3 & -4 \\
   \end{bmatrix}
   \begin{bmatrix}
   \mathbf{a}_1 \\
   \mathbf{a}_2 \\
   \mathbf{a}_3 \\
   \mathbf{b}
   \end{bmatrix}
   \sim
   \begin{bmatrix}
   1 & 0 & -4 & 4 \\
   0 & 3 & -2 & 1 \\
   0 & 6 & -5 & 4 \\
   \end{bmatrix}
   \begin{bmatrix}
   \mathbf{a}_1 \\
   \mathbf{a}_2 \\
   \mathbf{a}_3 \\
   \mathbf{b}
   \end{bmatrix}
   \sim
   \begin{bmatrix}
   1 & 0 & -4 & 4 \\
   0 & 3 & -2 & 1 \\
   0 & 0 & -1 & 2 \\
   \end{bmatrix}
   \]

   The system for this augmented matrix is consistent, so \( \mathbf{b} \) is in \( W \).

   c. \( \mathbf{a}_1 = 1\mathbf{a}_1 + 0\mathbf{a}_2 + 0\mathbf{a}_3 \). See the discussion following the definition of \( \text{Span}\{\mathbf{v}_1, \ldots, \mathbf{v}_p\} \).

31. a. The center of mass is \( \frac{1}{3} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + 1 \begin{bmatrix} 8 \\ 1 \end{bmatrix} + 1 \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 10/3 \\ 2 \end{bmatrix} \).
   
   b. The total mass of the new system is 9 grams. The three masses added, \( w_1, w_2, \) and \( w_3 \), satisfy the equation

   \[
   \frac{1}{9} \left( (w_1 + 1) \begin{bmatrix} 0 \\ 1 \end{bmatrix} + (w_2 + 1) \begin{bmatrix} 8 \\ 1 \end{bmatrix} + (w_3 + 1) \begin{bmatrix} 2 \\ 4 \end{bmatrix} \right) = \begin{bmatrix} 2 \\ 2 \end{bmatrix}
   \]

   which can be rearranged to

   \[
   (w_1 + 1) \begin{bmatrix} 0 \\ 1 \end{bmatrix} + (w_2 + 1) \begin{bmatrix} 8 \\ 1 \end{bmatrix} + (w_3 + 1) \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 18 \\ 18 \end{bmatrix}
   \]

   and
\[
\begin{bmatrix}
0 \\
1
\end{bmatrix} + w_2 \begin{bmatrix}
8 \\
1
\end{bmatrix} + w_3 \begin{bmatrix}
2 \\
4
\end{bmatrix} = \begin{bmatrix}
8 \\
12
\end{bmatrix}
\]

The condition \(w_1 + w_2 + w_3 = 6\) and the vector equation above combine to produce a system of three equations whose augmented matrix is shown below, along with a sequence of row operations:

\[
\begin{bmatrix}
1 & 1 & 1 & 6 \\
0 & 8 & 2 & 8 \\
1 & 1 & 4 & 12
\end{bmatrix} \sim \begin{bmatrix}
1 & 1 & 1 & 6 \\
0 & 8 & 2 & 8 \\
0 & 0 & 3 & 6
\end{bmatrix} \sim \begin{bmatrix}
1 & 1 & 0 & 4 \\
0 & 8 & 0 & 4 \\
0 & 0 & 1 & 2
\end{bmatrix} \sim \begin{bmatrix}
1 & 0 & 0 & 3.5 \\
0 & 1 & 0 & .5 \\
0 & 0 & 1 & 2
\end{bmatrix}
\]

Answer: Add 3.5 g at (0, 1), add .5 g at (8, 1), and add 2 g at (2, 4).

33. a. For \(j = 1, \ldots, n\), the \(j\)th entry of \((u + v) + w\) is \((u_j + v_j) + w_j\). By associativity of addition in \(\mathbb{R}\), this entry equals \(u_j + (v_j + w_j)\), which is the \(j\)th entry of \(u + (v + w)\). By definition of equality of vectors, \((u + v) + w = u + (v + w)\).

b. For any scalar \(c\), the \(j\)th entry of \(c(u + v)\) is \(c(u_j + v_j)\), and the \(j\)th entry of \(cu + cv\) is \(cu_j + cv_j\) (by definition of scalar multiplication and vector addition). These entries are equal, by a distributive law in \(\mathbb{R}\). So \(c(u + v) = cu + cv\).

---

**MATLAB Constructing a Matrix**

To access the data for Section 1.3, give the command `c1s3`. The data for Exercise 25, for example, consists of a matrix \(A\), its columns \(a_1, a_2, a_3\), and the vector \(b\). The command \(M = [a_1 \ a_2 \ a_3 \ b]\) creates a matrix using the vectors as its columns. The same matrix is created by the command \(M = [A \ b]\).

Each time you want data for a new exercise in Section 1.3, you need the command `c1s3`. After the first exercise, you can use the up-arrow (↑). This will make MATLAB scroll back through your old commands. You may be able to find “c2s1” faster than you can retype it. Press <Enter> to reuse the command.

Exercises 11–14, 25–28, and 31 can be solved using the commands `replace`, `swap`, and (occasionally) `scale`, described on page 1-6.
1.4 THE MATRIX EQUATION $Ax = b$

The ideas, boxed statements, and theorems in this section are absolutely fundamental for the rest of the text, so you should read the section extremely carefully.

---

**KEY IDEAS**

The definition of $Ax$ as a linear combination of the columns of $A$ will be used often. You should learn the definition in words as well as symbols. *Note:* It is not wrong to write a scalar on the *right* side of a vector and write $Ax$ as $a_1x_1 + \cdots + a_nx_n$, but the text follows the usual practice of writing a scalar on the *left* side of a vector.

You need to understand why Theorem 4 is true. That may take some time and effort. Example 3 should help, along with the proof. Theorem 4(d) can be restated as “The reduced echelon form of $A$ has no row of zeros.”

The phrase *logically equivalent* is explained in the statement of Theorem 4. This phrase is used with several statements in the same way that *if and only if* (or the symbol $\iff$) is used between two statements. (See the Mathematical Note at the end of Section 1.2 in this *Guide.*

Saying that statements (a), (b), (c), and (d) are logically equivalent means the same thing as saying that (a) $\iff$ (b), (b) $\iff$ (c), and (c) $\iff$ (d).

Key exercises are 1–20, 27, 28, 31 and 32. Think about 31 and 32, even if they are not assigned, because they introduce ideas you will need soon. (Don’t check the solution of Exercise 31 until you have written your own answer.)

**Checkpoint 1:** True or False? If an augmented matrix $[A \ b]$ has a pivot position in every row, then the equation $Ax = b$ is consistent.

*Note:* You should work a checkpoint problem when you first see it, provided that you have already read the text at least once. Always write your answer before comparing it with the one I have written. The checkpoint answer will be at the end of the solutions for this section.

---

**SOLUTIONS TO EXERCISES**

1. The text has the solution. Exercises 1–12 are designed to help you learn Theorem 3 and the definition of $Ax$. If a problem involves vectors—say, $v_1$, $v_2$, $v_3$—you can place the vectors into a matrix $[v_1 \ v_2 \ v_3]$, if that is helpful. If a problem involves a matrix $A$, you can give names to the columns of $A$—say, $a_1$, $a_2$, $a_3$—and reformulate a matrix equation as a vector equation. If a problem leads to a system of linear equations, you may regard it as either a vector equation or a matrix equation, whichever is most useful.

7. The left side of the equation is a linear combination of three vectors. Write the matrix $A$ whose columns are those three vectors, and create a variable vector $x$ with three entries:
Thus the equation \( A\mathbf{x} = \mathbf{b} \) is
\[
\begin{bmatrix}
4 & -5 & 7 \\
-1 & 3 & -8 \\
7 & -5 & 0 \\
-4 & 1 & 2
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix} =
\begin{bmatrix}
6 \\
-8 \\
0 \\
-7
\end{bmatrix}
\]

\textbf{Warning:} Be careful to distinguish between the \textit{matrix equation} \( A\mathbf{x} = \mathbf{b} \) and the \textit{augmented matrix} \( [a_1 \cdots a_p \mathbf{b}] \), which is used in Theorem 3 to refer to a system of linear equations having this augmented matrix. Thus, the answer to Exercise 7 is \textit{not} the augmented matrix at the right:
\[
\begin{bmatrix}
4 & -5 & 7 & 6 \\
-1 & 3 & -8 & -8 \\
7 & -5 & 0 & 0 \\
-4 & 1 & 2 & -7
\end{bmatrix}
\]

13. The vector \( \mathbf{u} \) is in the plane spanned by the columns of \( A \) if and only if \( \mathbf{u} \) is a linear combination of the columns of \( A \). This happens if and only if the equation \( A\mathbf{x} = \mathbf{u} \) has a solution. (See the box preceding Example 3 in Section 1.4.) To study this equation, reduce the augmented matrix \( [A \quad \mathbf{u}] \):
\[
\begin{bmatrix}
3 & -5 & 0 \\
-2 & 6 & 4 \\
1 & 1 & 4
\end{bmatrix}
\begin{bmatrix}
1 & 1 & 4 \\
-2 & 6 & 4 \\
1 & 3 & 5
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 1 & 4 \\
0 & 8 & 12 \\
0 & 0 & 0
\end{bmatrix}
\]

The equation \( A\mathbf{x} = \mathbf{u} \) has a solution, so \( \mathbf{u} \) is in the plane spanned by the columns of \( A \).

\textbf{Study Tip:} Exercises 17–20 require written explanations as well as calculations. For instance, your calculation for Exercise 17 might show the row reduction
\[
\begin{bmatrix}
1 & 3 & 0 & 3 \\
-1 & -1 & -1 & 1 \\
0 & -4 & 2 & -8 \\
2 & 0 & 3 & -1
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 3 & 0 & 3 \\
0 & 2 & -1 & 4 \\
0 & -4 & 2 & -8 \\
0 & 0 & -6 & 3 -7
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 3 & 0 & 3 \\
0 & 2 & -1 & 4 \\
0 & 0 & 0 & 5 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

After this, it is not enough to write “No, by Theorem 4.” Instead, you should show that you know \textit{why} Theorem 4 is relevant. For instance, you might write:

The matrix \( A \) does \textit{not} have a pivot in every row. By Theorem 4, the equation \( A\mathbf{x} = \mathbf{b} \) does \textit{not} have a solution for each \( \mathbf{b} \) in \( \mathbb{R}^4 \).

On a test, you probably would not have to know the theorem number. It might be enough to say “By a theorem,” instead of “By Theorem 4.” (Check with your instructor.)
19. The work in Exercise 17 shows that the equation \( Ax = b \) does not have a solution for each \( b \). That is, statement (d) in Theorem 4 is false. So all four statements in Theorem 4 are false. Since statement (b) is false, not all vectors in \( \mathbb{R}^4 \) can be written as a linear combination of the columns of \( A \). Since statement (c) is false, the columns of \( A \) do not span \( \mathbb{R}^4 \).

Checkpoint 2: Given \( v_1, v_2, v_3 \) as in Exercise 21, find a specific vector in \( \mathbb{R}^4 \) that is not in \( \text{Span}\{v_1, v_2, v_3\} \). (If necessary, reread Example 3.)

23. a. See the paragraph following equation (3). b. See the box before Example 3.
   c. See the warning following Theorem 4. d. See Example 4.
   e. See Theorem 4. f. See Theorem 4.

25. By definition, the matrix-vector product on the left is a linear combination of the columns of the matrix, in this case using weights \(-3, -1, \) and \(2\). So \( c_1 = -3, c_2 = -1, \) and \( c_3 = 2 \).

29. Start with any \( 3 \times 3 \) matrix \( B \) in echelon form that has three pivot positions. Perform a row operation (a row interchange or a row replacement) that creates a matrix \( A \) that is not in echelon form. Then \( A \) has the desired property. The justification is given by row reducing \( A \) to \( B \), in order to display the pivot positions. Since \( A \) has a pivot position in every row, the columns of \( A \) span \( \mathbb{R}^3 \), by Theorem 4.

31. A \( 3 \times 2 \) matrix has three rows and two columns. With only two columns, \( A \) can have at most two pivot columns, and so \( A \) has at most two pivot positions, which is not enough to fill all three rows. By Theorem 4, the equation \( Ax = b \) cannot be consistent for all \( b \) in \( \mathbb{R}^3 \). If \( A \) is any \( m \times n \) matrix with \( m>n \), the equation \( Ax = b \) be consistent for all \( b \) in \( \mathbb{R}^m \).

33. If the equation \( Ax = b \) has a unique solution, then the associated system of equations does not have any free variables. If every variable is a basic variable, then each column of \( A \) is a pivot column. So the reduced echelon form of \( A \) must be:

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0
\end{bmatrix}
\]

37. [M] The original matrix has no pivot in the fourth row, so its columns do not span \( \mathbb{R}^4 \), by Theorem 4.

**Helpful Hint:** For Exercises 41 and 42, use a matrix program to obtain an echelon form of the matrix. Try covering various columns of this matrix, one at a time, and ask yourself if the columns of the resulting matrix span \( \mathbb{R}^4 \). If you can delete one column, can you delete a second column? Why or why not?

The analysis here depends on the following idea, which is fairly obvious but is not explicitly mentioned in the text. When a row operation is performed on a matrix \( A \), the calculations for each new entry depend only on the other entries in the *same column*. If a column of \( A \) is removed, forming a new matrix, the absence of this column has no affect on any row-operation calculations for entries in the other columns of \( A \). (The absence of a column might affect the particular choice of row operations performed for some purpose, but that is not relevant.)
Answers to Checkpoints:

1. False. See the Warning after Theorem 4. If you missed this, you are not studying the text properly. You should read the text thoroughly before you look at the Study Guide and before you work on the exercises.

2. Let \( A = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & -1 & -1 \end{bmatrix} \) and \( \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \). Row reduce the augmented matrix for \( A \mathbf{x} = \mathbf{b} \) to determine values of \( b_1, \ldots, b_4 \) that make the equation inconsistent.

\[
\begin{bmatrix}
1 & 0 & 1 & b_1 \\
0 & 1 & 0 & b_2 \\
-1 & 0 & 0 & b_3 \\
0 & -1 & -1 & b_4
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 1 & b_1 \\
0 & 1 & 0 & b_2 \\
0 & 0 & 1 & b_3 + b_1 \\
0 & 0 & -1 & b_4 + b_2
\end{bmatrix}
\end{align*}
\]

\[
\begin{bmatrix}
1 & 0 & 1 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
b_1 \\
b_2 \\
b_3 + b_1 \\
b_4 + b_2 + b_3 + b_1
\end{bmatrix}
\]

Take \( \mathbf{b} = (1, 1, 0, 0) \), for example, or any other choice of \( b_1, \ldots, b_4 \) whose sum is not zero.

---

Mastering Linear Algebra Concepts: Span

Please begin by reviewing “How to Study Linear Algebra,” at the beginning of this Study Guide.

To really understand a key concept, you need to form an image in your mind that consists of the basic definition(s) together with many related ideas. Your goal at this point is to collect various ideas associated with the set Span\( \{v_1, \ldots, v_p\} \) and the concept of a set that “spans” \( \mathbb{R}^n \). Here are specific things to do now as you prepare a sheet (or sheets) for review and reference.

- Write the definition of Span\( \{v_1, \ldots, v_p\} \). (Learn it word for word.)

- Write the definition of the phrase: \( \{v_1, \ldots, v_p\} \) spans \( \mathbb{R}^n \). (See page 37.) Here span is a verb rather than a noun as in Span\( \{v_1, \ldots, v_p\} \).

- Add the equivalent description (not definition) of what is meant for a vector \( \mathbf{b} \) to be in Span\( \{v_1, \ldots, v_p\} \). (See page 30.)

- Copy Theorem 4 word for word. (If you try to rephrase or summarize it in your own words, you are likely to change the meaning.)
• Sketch some **geometric interpretations** of $\text{Span}\{v_1, \ldots, v_p\}$. (Select some of Figs. 8, 10, 11, and Exercises 19, 20 in Section 1.3.)

• Identify **special cases**. (Describe $\text{Span}\{u\}$ and $\text{Span}\{u, v\}$.)

• Summarize **algorithms** or **typical computations** (such as Example 6 and Exercises 11–14, 17, 18, 25, and 26 in Section 1.3, or Example 3 and Exercises 13–22 in Section 1.4).

• Describe connections with other concepts. (See pages 36–37).

Whenever you encounter new examples or situations that help you understand the concept of a spanning set, add them to this review sheet.

---

**MATLAB:** `gauss` and `bgauss`  
To solve $Ax = b$, row reduce the matrix $M = [A \ b]$. The command $x = [5; 3; -7]$ creates a column vector $x$ with entries 5, 3, −7. Matrix vector multiplication is $A\times x$.

To speed up row reduction of $M = [A \ b]$, the command `gauss(M,r)` will use the leading entry in row $r$ of $M$ as a pivot, and use row replacements to create zeros in the pivot column below this pivot entry. The result is stored in the default matrix “ans”, unless you assign the result to some other variable, such as $M$ itself.

For the backward phase of row reduction, use `bgauss(M,r)`, which selects the leading entry in row $r$ of $M$ as the pivot, and creates zeros in the column above the pivot. Use `scale` to create leading 1’s in the pivot positions. The commands `gauss`, `bgauss`, and `scale` are in the Laydata5 Toolbox, which you can download from the web.

---

### 1.5 SOLUTION SETS OF LINEAR SYSTEMS

Many of the concepts and computations in linear algebra involve sets of vectors which are visualized geometrically as lines and planes. The most important examples of such sets are the solution sets of linear systems.

**KEY IDEAS**

Visualize the solution set of a homogeneous equation $Ax = 0$ as:

- the single point $0$, when $Ax = 0$ has only the trivial solution,
- a line through $0$, when $Ax = 0$ has one free variable,
- a plane through $0$, when $Ax = 0$ has two free variables.

(For more than two free variables, also use a plane through $0$.)
For $\mathbf{b} \neq \mathbf{0}$, visualize the solution set of $A\mathbf{x} = \mathbf{b}$ as:

- empty, if $\mathbf{b}$ is not a linear combination of the columns of $A$,
- one nonzero point (vector), when $A\mathbf{x} = \mathbf{b}$ has a unique solution,
- a line not through $\mathbf{0}$, when $A\mathbf{x} = \mathbf{b}$ is consistent and has one free variable,
- a plane not through $\mathbf{0}$, when $A\mathbf{x} = \mathbf{b}$ is consistent and has two or more free variables.

The solution set of $A\mathbf{x} = \mathbf{b}$ is said to be described *implicitly*, because the equation is a condition an $\mathbf{x}$ must satisfy in order to be in the set, yet the equation does not show how to find such an $\mathbf{x}$. When the solution set of $A\mathbf{x} = \mathbf{0}$ is written as $\text{Span}\{\mathbf{v}_1, \ldots, \mathbf{v}_p\}$, the set is said to be described *explicitly*; each element in the set is produced by forming a linear combination of $\mathbf{v}_1, \ldots, \mathbf{v}_p$.

A common explicit description of a set is an equation in *parametric vector form*. Examples are:

\[
\mathbf{x} = t\mathbf{v}, \quad \text{a line through } \mathbf{0} \text{ in the direction of } \mathbf{v},
\]

\[
\mathbf{x} = \mathbf{p} + t\mathbf{v}, \quad \text{a line through } \mathbf{p} \text{ in the direction of } \mathbf{v},
\]

\[
\mathbf{x} = x_2\mathbf{u} + x_3\mathbf{v}, \quad \text{a plane through } \mathbf{0}, \mathbf{u}, \text{ and } \mathbf{v},
\]

\[
\mathbf{x} = \mathbf{p} + x_2\mathbf{u} + x_3\mathbf{v}, \quad \text{a plane through } \mathbf{p} \text{ parallel to the plane whose equation is } \mathbf{x} = x_2\mathbf{u} + x_3\mathbf{v}.
\]

An equation in parametric vector form describes a set explicitly because the equation shows how to produce each $\mathbf{x}$ in the set.

To *solve* an equation $A\mathbf{x} = \mathbf{b}$ means to find an explicit description of the solution set. If the system is inconsistent, the solution set is empty. Otherwise, the description of all solutions can be written in parametric vector form, in which the parameters are the free variables from the system. *Important:* The number of free variables in $A\mathbf{x} = \mathbf{b}$ depends only on $A$, not on $\mathbf{b}$.

Theorem 6 and the paragraph following it are important. They describe how the solutions of $A\mathbf{x} = \mathbf{0}$ and $A\mathbf{x} = \mathbf{b}$ are related when the solution set of $A\mathbf{x} = \mathbf{b}$ is nonempty. See Figs. 5 and 6.

**Key exercises:** 5–16, 29–32, 37.

**SOLUTIONS TO EXERCISES**

1. Reduce the augmented matrix to echelon form and circle the pivot positions. If a column of the *coefficient* matrix is not a pivot column, the corresponding variable is free and the system of equations has a nontrivial solution. Otherwise, the system has only the trivial solution.

\[
\begin{bmatrix}
2 & -5 & 8 & 0 \\
-2 & -7 & 1 & 0 \\
4 & 2 & 7 & 0
\end{bmatrix}
\]

The variable $x_3$ is free, so the system has a nontrivial solution.

7. Always use the *reduced* echelon form of an augmented matrix to find the solutions of a system. See the text’s discussion of back substitution on pages 19–20.
If you wrote something like the system above, then you made a common mistake. The matrix in the text problem is a coefficient matrix, not an augmented matrix. You should row reduce \([A \ 0]\). The correct system of equations is

\[
\begin{align*}
13x - 9x_3 &= 0 \\
23x - 4x_3 &= 0
\end{align*}
\]

The basic variables are \(x_1\) and \(x_2\), with \(x_3\) and \(x_4\) free. Next, \(x_1 = -9x_3 + 8x_4\), and \(x_2 = 4x_3 - 5x_4\).

The general solution is

\[
x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -9x_3 + 8x_4 \\ 4x_3 - 5x_4 \\ x_3 \\ 0 \end{bmatrix} = x_3 \begin{bmatrix} -9 \\ 4 \\ 1 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} 8 \\ -5 \\ 0 \\ 1 \end{bmatrix}
\]

The solution set is the same as \(\text{Span}\{\mathbf{u}, \mathbf{v}\}\), where \(\mathbf{u} = (-9, 4, 1, 0)\) and \(\mathbf{v} = (8, -5, 0, 1)\).

Originally, the solution set was described implicitly, by a set of equations. Now the solution set is described explicitly, in parametric vector form.

11.

\[
\begin{align*}
13x - 203 - 50 &= 0 \\
00 &= 0100 - 10 \\
00 &= 01 -40 \\
00 &= 00000
\end{align*}
\]

\[
\begin{align*}
x_1 - 4x_2 + 5x_6 &= 0 \\
x_3 &= 0 \\
x_3 - 4x_6 &= 0
\end{align*}
\]

Some students are not sure what to do with \(x_4\). Some ignore it; others set it equal to zero. In fact, \(x_4\) is free; there is no constraint on \(x_4\) at all. The basic variables are \(x_1, x_3,\) and \(x_5\). The remaining variables are free. So, \(x_1 = 4x_2 - 5x_6, x_3 = x_6,\) and \(x_5 = 4x_6,\) with \(x_2, x_4,\) and \(x_6\) free. In parametric vector form,
The solution set is the same as Span\{u, v, w\}.

**Study Tip:** When solving a system, identify (and perhaps circle) the basic variables. All other variables are free.

13. To write the general solution in parametric vector form, pull out the constant terms that do not involve the free variable:

\[
x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 5 + 4x_3 \\ -2 - 7x_3 \\ -2 \end{bmatrix} + \begin{bmatrix} 4x_3 \\ -7x_3 \\ 0 \end{bmatrix} = x_3 \begin{bmatrix} 4 \\ -7 \\ 1 \end{bmatrix} + \begin{bmatrix} 5 \\ -2 \\ 0 \end{bmatrix}
\]

Geometrically, the solution set is the line through \(\begin{bmatrix} -2 \\ 0 \\ 0 \end{bmatrix}\) parallel to \(\begin{bmatrix} 4 \\ -7 \\ 1 \end{bmatrix}\).

**Checkpoint:** Let \(A\) be a 2 \(\times\) 2 matrix. Answer True or False: If the solution set of \(Ax = 0\) is a line through the origin in \(\mathbb{R}^2\) and if \(b \neq 0\), then the solution set of \(Ax = b\) is a line not through the origin.

19. The line through \(a\) parallel to \(b\) can be written as \(x = a + tb\), where \(t\) represents a parameter:

\[
x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -2 \\ 0 \end{bmatrix} + t \begin{bmatrix} -5 \\ 3 \end{bmatrix}, \text{ or } \begin{cases} x_1 = -2 - 5t \\ x_2 = 3t \end{cases}
\]

23. a. See the first paragraph of the subsection titled “Homogeneous Linear Systems.”

b. See the first two sentences of the subsection titled “Parametric Vector Form.”

c. See the box before Example 1.

d. See the paragraph that precedes Fig. 5.

e. See Theorem 6.
25. Suppose \( p \) satisfies \( Ax = b \). Then \( Ap = b \). Theorem 6 says that the solution set of \( Ax = b \) equals the set \( S = \{ w : w = p + v_h \text{ for some } v_h \text{ such that } Av_h = 0 \} \). There are two things to prove: (a) every vector in \( S \) satisfies \( Ax = b \), (b) every vector that satisfies \( Ax = b \) is in \( S \).

a. Let \( w \) have the form \( w = p + v_h \), where \( Av_h = 0 \). Then
\[
Aw = A(p + v_h) = Ap + Av_h \\
= b + 0 = b
\]
So every vector of the form \( p + v_h \) satisfies \( Ax = b \).

b. Now let \( w \) be any solution of \( Ax = b \), and set \( v_h = w - p \). Then
\[
Av_h = A(w - p) = Aw - Ap = b - b = 0
\]
So \( v_h \) satisfies \( Ax = 0 \). Thus every solution of \( Ax = b \) has the form \( w = p + v_h \).

31. \( A \) is a \( 3 \times 2 \) matrix with two pivot positions.

a. Since \( A \) has a pivot position in each column, each variable in \( Ax = 0 \) is a basic variable. So the equation \( Ax = 0 \) has no free variables and hence no nontrivial solution.

b. With two pivot positions and three rows, \( A \) cannot have a pivot in every row. So the equation \( Ax = b \) cannot have a solution for every possible \( b \) (in \( \mathbb{R}^3 \)), by Theorem 4 in Section 1.4.

37. If you worked on the Checkpoint when you first saw it, you should be ready for this exercise. Since the solution set of \( Ax = 0 \) contains the point \((4, 1)\), the vector \( x = (4, 1) \) satisfies \( Ax = 0 \). Write this equation as a vector equation, using \( a_1 \) and \( a_2 \) for the columns of \( A \):
\[
4 \cdot a_1 + 1 \cdot a_2 = 0
\]
Then \( a_2 = -4a_1 \). So choose any nonzero vector for the first column of \( A \) and multiply that column by \(-4\) to get the second column of \( A \). For example, set \( A = \begin{bmatrix} 1 & -4 \\ 1 & -4 \end{bmatrix} \).

Finally, the only way the solution set of \( Ax = b \) could not be parallel to the line through \((4, 1)\) and the origin is for the solution set of \( Ax = b \) to be empty. (Theorem 6 applies only to the case when the equation \( Ax = b \) has a nonempty solution set.) For \( b \), take any vector that is not a multiple of the columns of \( A \).

Answer to Checkpoint: False. The solution set could be empty. In this case, the solution set of \( Ax = b \) is not produced by translating the (nonempty) solution set of \( Ax = 0 \). See the Warning after Theorem 6.
MATLAB Zero Matrices

The command `zeros(m,n)` creates an $m \times n$ matrix of zeros. When solving an equation $Ax = 0$, create an augmented matrix:

$$M = [A \ zeros(m, 1)]$$

$m$ is the number of rows in $A$.

Then use `gauss`, `bgauss`, and `scale` to row reduce $M$ completely.

1.6 APPLICATIONS OF LINEAR SYSTEMS

All of the examples and exercises in this system involve linear systems that have multiple solutions. In each case, make a note of why you should expect the system to have many solutions.

STUDY NOTES

The Leontief exchange model concerns the dollar value (called the price) of the annual output of each sector of a nation’s economy. An equilibrium price vector $p$ provides a list of prices, one for each section, such that each sector’s expenses and income are in balance. Example 1 shows that there are many equilibrium price vectors; each one is a multiple of a fixed equilibrium price vector. This means that once the prices are all in balance, multiplying all the prices by a fixed constant does not affect the balance. For instance, if all prices are doubled, then each sector’s expenses and income are doubled at the same time and hence they remain in balance.

A solution of a chemical equation-balance problem is a list of coefficients that appear on the various terms in the chemical equation. When a chemical equation is balanced, the number of atoms of each type on the left side of the equation matches the number of corresponding atoms on the right side. If the coefficients in the equation are each multiplied by a fixed positive integer, the equation will remain balanced. So, there are many solutions to a chemical equation-balance problem.

The problems here in network flow have multiple solutions for the simple reason that there are more variables than there are constraint equations. The equations for network flow are mostly nonhomogeneous. In contrast, the Leontief model and the chemical equation-balance problem both lead to systems of homogeneous equations.

SOLUTIONS TO EXERCISES

1. Fill in the exchange table one column at a time. The entries in a column describe where a sector’s output goes. The decimal fractions in each column sum to 1.
Denote the total annual output (in dollars) of the sectors by \( p_G \) and \( p_S \). From the first row, the total input to the Goods sector is \( .2 \ p_G + .7 \ p_S \). The Goods sector must pay for that. So the equilibrium prices must satisfy

\[
\text{income} - \text{expenses} = \frac{.2 \ p_G + .7 \ p_S}{\ p_G}
\]

From the second row, the input (that is, the expense) of the Services sector is \( .8 \ p_G + .3 \ p_S \). The equilibrium equation for the Services sector is

\[
\text{income} - \text{expenses} = \frac{.8 \ p_G + .3 \ p_S}{\ p_S}
\]

Move all variables to the left side and combine like terms:

\[
.8 \ p_G - .7 \ p_S = 0
\]

\[
-.8 \ p_G + .7 \ p_S = 0
\]

Row reduce the augmented matrix:

\[
\begin{bmatrix}
.8 & -.7 & 0 \\
-.8 & .7 & 0 \\
\end{bmatrix}
\sim \begin{bmatrix}
.8 & -.7 & 0 \\
0 & 0 & 0 \\
\end{bmatrix}
\sim \begin{bmatrix}
1 & -8.75 & 0 \\
0 & 0 & 0 \\
\end{bmatrix}
\]

The general solution is \( p_G = .875 \ p_S \), with \( p_S \) free. One equilibrium solution is \( p_S = 1000 \) and \( p_G = 875 \). If one uses fractions instead of decimals in the calculations, the general solution would be written \( p_G = (7/8) \ p_S \), and a natural choice of prices might be \( p_S = 80 \) and \( p_G = 70 \). Only the ratio of the prices is important: \( p_G = .875 \ p_S \). The economic equilibrium is unaffected by a proportional change in prices.

### 7.

The following vectors list the numbers of atoms of sodium (Na), hydrogen (H), carbon (C), and oxygen (O):

- \( \text{NaHCO}_3 : \) \[ \begin{bmatrix} 1 \\ 1 \\ 3 \end{bmatrix} \]
- \( \text{H}_3\text{C}_6\text{H}_5\text{O}_7 : \) \[ \begin{bmatrix} 8 \\ 6 \\ 7 \end{bmatrix} \]
- \( \text{Na}_3\text{C}_6\text{H}_5\text{O}_7 : \) \[ \begin{bmatrix} 3 \\ 5 \\ 7 \end{bmatrix} \]
- \( \text{H}_2\text{O} : \) \[ \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} \]
- \( \text{CO}_2 : \) \[ \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \]

The order of the various atoms is not important. The list here was selected by writing the elements in the order in which they first appear in the chemical equation, reading left to right.
The coefficients \(x_1, \ldots, x_5\) satisfy the vector equation

\[
egin{bmatrix}
1 & 0 & -3 & 0 & 0 \\
1 & 8 & -5 & -2 & 0 \\
1 & 6 & -6 & 0 & -1 \\
3 & 7 & -7 & -1 & -2 \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
\end{bmatrix}
=
\begin{bmatrix}
x_3 \\
x_4 \\
x_5 \\
x_5 \\
\end{bmatrix}
\begin{bmatrix}
0 \\
2 \\
1 \\
1 \\
0 \\
\end{bmatrix}
\]

Move all terms to the left side (changing the sign of each entry in the third, fourth, and fifth vectors) and reduce the augmented matrix:

\[
\begin{bmatrix}
1 & 0 & -3 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & 1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\
\end{bmatrix}
\]

The general solution is \(x_1 = x_5\), \(x_2 = (1/3)x_5\), \(x_3 = (1/3)x_5\), \(x_4 = x_5\), and \(x_5\) is free. Take \(x_5 = 3\). Then \(x_1 = x_4 = 3\), and \(x_2 = x_3 = 1\). The balanced equation is

\[
3\text{NaHCO}_3 + \text{H}_3\text{C}_6\text{H}_5\text{O}_7 \rightarrow \text{Na}_3\text{C}_6\text{H}_5\text{O}_7 + 3\text{H}_2\text{O} + 3\text{CO}_2
\]

13. Write the equations for each intersection (see the diagram for the intersection labels):

<table>
<thead>
<tr>
<th>Intersection</th>
<th>Flow in</th>
<th>Flow out</th>
<th>Rearrange the equations:</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>(x_1 + 30)</td>
<td>(x_1 + 80)</td>
<td>(x_1 - x_2) = -50</td>
</tr>
<tr>
<td>B</td>
<td>(x_3 + x_5)</td>
<td>(x_2 + x_4)</td>
<td>(x_2 - x_1 + x_4 - x_5) = 0</td>
</tr>
<tr>
<td>C</td>
<td>(x_6 + 100)</td>
<td>(x_5 + 40)</td>
<td>(x_5 - x_6) = 60</td>
</tr>
<tr>
<td>D</td>
<td>(x_4 + 40)</td>
<td>(x_6 + 90)</td>
<td>(x_4 - x_6) = 50</td>
</tr>
<tr>
<td>E</td>
<td>(x_1 + 60)</td>
<td>(x_3 + 20)</td>
<td>(x_1 - x_3) = -40</td>
</tr>
</tbody>
</table>

Total flow: \(230 = 230\)
1.6  •  Applications of Linear Systems  1-27

Completely reduce the augmented matrix:

\[
\begin{bmatrix}
1 & -1 & 0 & 0 & 0 & 0 & -50 \\
0 & 1 & -1 & 1 & -1 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 60 \\
0 & 0 & 0 & 1 & 0 & -1 & 50 \\
1 & 0 & -1 & 0 & 0 & 0 & -40
\end{bmatrix}
\sim
\begin{bmatrix}
1 & -1 & 0 & 0 & 0 & 0 & -50 \\
0 & 1 & -1 & 1 & -1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & -1 & 50 \\
0 & 0 & 0 & 0 & 1 & -1 & 60 \\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & -1 & 0 & 0 & 0 & -40 \\
0 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & -1 & 50 \\
0 & 0 & 0 & 1 & -1 & 60 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

\[
x_1 = x_3 - 40 \\
x_2 = x_3 + 10 \\
x_3 \text{ is free} \\
x_4 = x_6 + 50 \\
x_5 = x_6 + 60 \\
x_6 \text{ is free}
\]

a. The general solution is

b. To find minimum flows, note that since \( x_1 \) cannot be negative, \( x_3 \geq 40 \). This implies that \( x_2 \geq 50 \). Also, since \( x_6 \) cannot be negative, \( x_4 \geq 50 \) and \( x_5 \geq 60 \). The minimum flows are \( x_2 = 50, x_3 = 40, x_4 = 50, x_5 = 60 \) (when \( x_1 = 0 \) and \( x_6 = 0 \)).

**MATLAB  Rational Format**

Chemical equation-balance problems are studied best using exact or symbolic arithmetic, because the balance variables must be whole numbers (with no round-off allowed). In MATLAB, a simple approach is to execute the command `format rat`, which will make MATLAB display matrix or vector entries as rational numbers. In general, the rational number displayed might be only an approximation for a floating-point number. But since the chemical equations studied here have integer coefficients, `format rat` will make MATLAB display the exact (rational) value of every entry during row reduction. Use `format` or `format short` to return to the standard MATLAB display of numbers.

Once you find a rational solution of a chemical equation-balance problem, you can multiply the entries in the solution vector by a suitable integer to produce a solution that involves only whole numbers.
1.7 LINEAR INDEPENDENCE

This section is as important as Section 1.4 and should be studied just as carefully. Full understanding of the concepts will take time, so get started on the section now.

KEY IDEAS

Figures 1 and 2, along with Theorem 7, will help you understand the nature of a linearly dependent set. (Fig. 2 applies only when \( u \) and \( v \) are independent.) But you must also learn the definitions of linear dependence and linear independence, word for word! Many theoretical problems involving a linearly dependent set are treated by the definition, because it provides an equation (the dependence equation) with which to work. (See the proof of Theorem 7.)

The box before Example 2 contains a very useful fact. Any time you need to study the linear independence of a set of \( p \) vectors in \( \mathbb{R}^n \), you can always form an \( n \times p \) matrix \( A \) with those vectors as columns and then study the matrix equation \( Ax = 0 \). This is not the only method, however. Stay alert for three special situations:

- A set of two vectors. Always check this by inspection; don’t waste time on row reduction of \([A \ 0]\). The set is linearly independent if neither of the vectors is a multiple of the other. (For brevity, I sometimes say that “the vectors are not multiples.”) See Example 3.
- A set that contains too many vectors, that is, more vectors than entries in the vectors; the columns of a short, fat matrix. Theorem 8.
- A set that contains the zero vector. Theorem 9.

The most common mistake students make when checking a set of three or more vectors for independence is to think they only have to verify that no vector is a multiple of one of the other vectors. Wrong! Study Example 5 and Figure 4.

Key exercises are 9–20 and 23–28, and 30. Try Exercise 35, even if it is not assigned. Think carefully, and write your answer before checking the answer section.

SOLUTIONS TO EXERCISES

1. Use an augmented matrix to study the solution set of \( x_1u + x_2v + x_3w = 0 \), where \( u, v, \) and \( w \) are the three given vectors. Since

\[
\begin{bmatrix}
5 & 7 & 9 & 0 \\
0 & 2 & 4 & 0 \\
0 & -6 & -8 & 0
\end{bmatrix}
\quad
\begin{bmatrix}
5 & 7 & 9 & 0 \\
0 & 2 & 4 & 0 \\
0 & 0 & 4 & 0
\end{bmatrix}
\]

are the three given vectors. Since there are no free variables. So the homogeneous equation has only the trivial solution. The vectors are linearly independent.

**Warning:** Whenever you study a homogeneous equation, you may be tempted to omit the augmented column of zeros because it never changes under row operations. I urge you to keep the zeros, to avoid possibly misinterpreting your own calculations. In Exercise 1, if you wrote
you might conclude that “the system is inconsistent” and then go on to make some crazy statement about linear dependence or independence. Don’t laugh. I have seen this happen on exams. A more common error occurs in a problem like Exercise 7. In that exercise, if you write

\[
\begin{bmatrix}
1 & 4 & -3 & 0 \\
-2 & -7 & 5 & 1 \\
-4 & -5 & 7 & -5
\end{bmatrix}
\sim
\begin{bmatrix}
0 & 1 & -1 & 1 \\
0 & 11 & -5 & 5 \\
0 & 0 & 6 & -6
\end{bmatrix}
\]

you might conclude that “the system has a unique solution” and the vectors are linearly independent. However, the four columns are actually linearly dependent. In both cases, the error is to misinterpret your matrix as an augmented matrix.

**7. Study the equation** \(Ax = 0\). Some people may start with the method of Example 2:

\[
\begin{bmatrix}
1 & 4 & -3 & 0 & 0 \\
-2 & -7 & 5 & 1 & 0 \\
-4 & -5 & 7 & -5 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 4 & -3 & 0 & 0 \\
0 & 1 & -1 & 1 & 0 \\
0 & 11 & -5 & 5 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 4 & -3 & 0 & 0 \\
0 & 1 & -1 & 1 & 0 \\
0 & 0 & 6 & -6 & 0
\end{bmatrix}
\]

But this is a waste of time. There are only 3 rows, so there are at most three pivot positions. Hence, at least one of the four variables must be free. So the equation \(Ax = 0\) has a nontrivial solution and the columns of \(A\) are linearly dependent.

**Warning:** Exercise 9 and Practice Problem 3 emphasize that to check whether a set such as \(\{v_1, v_2, v_3\}\) is linearly dependent, it is not wise to check instead whether \(v_3\) is a linear combination of \(v_1\) and \(v_2\).

**13. To study the linear dependence of three vectors, say** \(v_1, v_2, v_3\), row reduce the augmented matrix \([v_1 \quad v_2 \quad v_3 \quad 0]\\):  

\[
\begin{bmatrix}
1 & -2 & 3 & 0 \\
5 & -9 & h & 0 \\
-3 & 6 & -9 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & -2 & 3 & 0 \\
0 & 1 & h-15 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

The equation \(x_1v_1 + x_2v_2 + x_3v_3 = 0\) has a free variable and hence a nontrivial solution no matter what the value of \(h\). So the vectors are linearly dependent for all values of \(h\).

**Checkpoint:** What is wrong with the following statement?

The vectors \(\begin{bmatrix}3 \\ -1\end{bmatrix}, \begin{bmatrix}2 \\ 8\end{bmatrix}, \begin{bmatrix}-5 \\ 3\end{bmatrix}, \begin{bmatrix}7 \\ -4\end{bmatrix}\) are linearly dependent “because there is a free variable,” or “because there are more variables than equations.”
15. The set $\{ \begin{bmatrix} 5 \\ 1 \\ 8 \\ 3 \\ -1 \\ 7 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \\ -5 \\ -2 \\ -4 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 7 \\ -3 \\ 1 \\ 1 \\ 1 \end{bmatrix} \}$ is obviously linearly dependent, by Theorem 8, because there are more vectors (4) than entries in the vectors. On a test, you probably will not have to know the theorem number. Check with your instructor.

19. The set is linearly independent because neither vector is a multiple of the other vector. [Two of the entries in the first vector are –4 times the corresponding entry in the second vector. But this multiple does not work for the third entries.]

21. a. See the box before Example 2.
   b. See the warning after Theorem 7.
   c. See Fig. 3, after Theorem 8.
   d. See the remark following Example 4.

25. $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ and $\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$

31. Think of $A = [a_1 \ a_2 \ a_3]$. The text points out that $a_3 = a_1 + a_2$. Rewrite this as $a_1 + a_2 - a_3 = 0$. As a matrix equation, $Ax = 0$ for $x = (1, 1, -1)$.

33. The text uses Theorem 7 to conclude that $\{v_1, \ldots, v_4\}$ is linearly dependent. Another argument is to rewrite the equation $v_3 = 2v_1 + v_2$ as $2v_1 + v_2 + (-1)v_3 + 0v_4 = 0$. This is a linear dependence relation. Some students think of this argument rather than Theorem 7. Did you? (I hope you did not read the answer before trying this problem.)

37. True. The text gives a complete answer.

39. If for all $b$ the equation $Ax = b$ has at most one solution, then take $b = 0$, and conclude that the equation $Ax = 0$ has at most one solution. Then the trivial solution is the only solution, and so the columns of $A$ are linearly independent.

43. [M] Make $v$ any one of the columns of $A$ that is not in $B$ and row reduce the augmented matrix $[B \ v]$. The calculations will show that the equation $Bx = v$ is consistent, which means that $v$ is a linear combination of the columns of $B$. Thus, each column of $A$ that is not a column of $B$ is in the set spanned by the columns of $B$.

Answer to Checkpoint: The set of four vectors contains only vectors, no variables of any kind, and no equations. It makes no sense to talk about the variables in a set of vectors. Variables appear in an equation. One cannot assume that the writer of the statement has any idea of the appropriate equation. If you want to give an explanation involving variables, then you must specify the equation. One correct answer is: the vectors are linearly dependent because the equation $x_1 \begin{bmatrix} 3 \\ -1 \end{bmatrix} + x_2 \begin{bmatrix} 2 \\ 8 \end{bmatrix} + x_3 \begin{bmatrix} -5 \\ 3 \end{bmatrix} + x_4 \begin{bmatrix} -4 \\ 7 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ necessarily has a free variable.
Mastering Linear Algebra Concepts: Linear Independence

In Section 1.4 of this Guide, I described how to begin forming a mental image of the concept of a spanning set. The same technique works for linear independence. The goal is to merge all the ideas you find regarding linear independence into a single mental image, with each part immediately available in your mind for use as needed. Start now to organize on paper your understanding of linear independence/dependence, using the following list as a guide. In each case, write information that you think will be helpful. (Definitions and theorems should be copied word-for-word.)

- definitions of linear independence and dependence
- equivalent descriptions
- geometric interpretations
- special cases
- examples and “counterexamples”
- algorithms or typical computations
- connections with other concepts

As you work on your notes, be careful to use terminology correctly. For instance, the term “linearly independent” may be applied to a set of vectors, but it never is applied to a matrix or to an equation. The columns of a matrix may be linearly independent, but it is meaningless to refer to a linearly independent matrix. Similarly, solutions of a system of linear equations may be linearly independent, but the term “linearly independent equations” has never been defined. Finally, a set of vectors or a matrix cannot have a “nontrivial solution”. Only equations have solutions.

1.8 INTRODUCTION TO LINEAR TRANSFORMATIONS

Linear transformations are important for both the theory and the applications of linear algebra. You will see both uses in a variety of settings throughout the text. The graphical descriptions in this section will be augmented in Section 1.9 and in a later section on computer graphics.

STUDY NOTES

Viewing the correspondence from a vector \( x \) to a vector \( A x \) as a mapping provides a dynamic interpretation of matrix-vector multiplication and a new way to understand the equation \( A x = b \). Using the language of computer science, we can describe a matrix in two ways—as a data structure (a rectangular array of numbers) and as a program (a prescription for transforming vectors). Strictly speaking, however, the actual linear transformation is the function or mapping \( x \mapsto A x \) rather than just \( A \) itself.
Here is a way to visualize a matrix acting as a linear transformation. The entries in the input vector $\mathbf{x}$ are assigned as weights that multiply the corresponding columns of $A$, then the resulting weighted columns are added together to produce the output vector $\mathbf{b}$.

As you learn the definition of a linear transformation $T$, don’t forget the crucial phrases “for all $\mathbf{u}$ and $\mathbf{v}$ in the domain of $T$” and “for all $\mathbf{u}$ and all scalars $c$.” The mapping $T$ defined by $T(x_1, x_2) = (|x_2|, |x_1|)$ is not a linear mapping, and yet $T$ satisfies the linearity properties for some vectors in its domain and some scalars.

The key exercises are 17–20, 25 and 31.

**SOLUTIONS TO EXERCISES**

1. $T(\mathbf{u}) = A\mathbf{u} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ -3 \end{bmatrix} = \begin{bmatrix} 2 \\ -6 \end{bmatrix}$, $T(\mathbf{v}) = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 2a \\ 2b \end{bmatrix}$

5. $[A \quad b] = \begin{bmatrix} 1 & -5 & -7 & -2 \\ -3 & 7 & 5 & -2 \end{bmatrix} \sim \begin{bmatrix} 1 & -5 & -7 & -2 \\ 0 & 1 & 2 & 1 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 3 & 3 \\ 0 & 1 & 2 & 1 \end{bmatrix}$

Note that a solution is *not* $\begin{bmatrix} 3 \\ 1 \end{bmatrix}$. To avoid this common error, write the equations:

\[
\begin{align*}
x_1 + 3x_3 &= 3 \\
x_2 + 2x_3 &= 1
\end{align*}
\]

and solve for the basic variables: $\begin{align*}x_1 &= 3 - 3x_3 \\
x_2 &= 1 - 2x_3 \\
x_3 &= \text{free}\end{align*}$

General solution: $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 - 3x_3 \\ 1 - 2x_3 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ -2 \end{bmatrix} + x_3 \begin{bmatrix} -3 \\ 0 \\ 1 \end{bmatrix}$. For a particular solution, one might choose $x_3 = 0$ and $\mathbf{x} = \begin{bmatrix} 3 \\ 1 \\ 0 \end{bmatrix}$. 
7. \( a = 5; \) the domain of \( T \) is \( \mathbb{R}^5 \), because a \( 6 \times 5 \) matrix has 5 columns and for \( Ax \) to be defined, \( x \) must be in \( \mathbb{R}^5 \). \( b = 6; \) the codomain of \( T \) is \( \mathbb{R}^6 \), because \( Ax \) is a linear combination of the columns of \( A \), and each column of \( A \) is in \( \mathbb{R}^6 \).

13. The transformation may be described geometrically as a reflection through the origin. Two other correct descriptions are a rotation of \( \pi \) radians about the origin and a rotation of \( -\pi \) radians about the origin. See the figure.

18. **Additional Hint:** Draw a line through \( w \) parallel to \( v \), and draw a line through \( w \) parallel to \( u \). This will help you write \( w \) as a linear combination of \( u \) and \( v \).

19. All you know are the images of \( e_1 \) and \( e_2 \) and the fact that \( T \) is linear. The key idea is to write

\[
\mathbf{x} = \begin{bmatrix} 5 \\ -3 \end{bmatrix} = 5\begin{bmatrix} 1 \\ 0 \end{bmatrix} - 3\begin{bmatrix} 0 \\ 1 \end{bmatrix} = 5\mathbf{e}_1 - 3\mathbf{e}_2. \text{ Then, from the linearity of } T, \text{ write}
\]

\[
T(\mathbf{x}) = T(5\mathbf{e}_1 - 3\mathbf{e}_2) = 5T(\mathbf{e}_1) - 3T(\mathbf{e}_2) = 5\begin{bmatrix} 2 \\ 5 \end{bmatrix} - 3\begin{bmatrix} -1 \\ 6 \end{bmatrix} = \begin{bmatrix} 13 \\ 7 \end{bmatrix}
\]

To find the image of \( \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \), observe that \( \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = x_1\begin{bmatrix} 1 \\ 0 \end{bmatrix} + x_2\begin{bmatrix} 0 \\ 1 \end{bmatrix} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2. \text{ Then}
\]

\[
T(\mathbf{x}) = T(x_1\mathbf{e}_1 + x_2\mathbf{e}_2) = x_1T(\mathbf{e}_1) + x_2T(\mathbf{e}_2) = x_1\begin{bmatrix} 2 \\ 5 \end{bmatrix} + x_2\begin{bmatrix} -1 \\ 6 \end{bmatrix} = \begin{bmatrix} 2x_1 - x_2 \\ 5x_1 + 6x_2 \end{bmatrix}
\]

21. a. A function is another word for transformation or mapping.
   b. See the paragraph before Example 1.
   c. See Figure 2. Or, see the paragraph before Example 1.
   d. See the paragraph after the definition of a linear transformation.
   e. See the paragraph following the box that contains equation (4).

25. Any point \( \mathbf{x} \) on the line through \( \mathbf{p} \) in the direction of \( \mathbf{v} \) satisfies the parametric equation \( \mathbf{x} = \mathbf{p} + t\mathbf{v} \) for some value of \( t \). By linearity, the image \( T(\mathbf{x}) \) satisfies the parametric equation

\[
T(\mathbf{x}) = T(\mathbf{p} + t\mathbf{v}) = T(\mathbf{p}) + tT(\mathbf{v}) \quad (*)
\]
If \( T(v) = 0 \), then \( T(x) = T(p) \) for all values of \( t \), and the image of the original line is just a single point. Otherwise, (*) is the parametric equation of a line through \( T(p) \) in the direction of \( T(v) \).

**Study Tip:** Exercise 31 is important, because it will help you to connect the concepts of linear dependence and linear transformation. Be sure to try the exercise first, before looking in the answer section of the text. Don’t feel badly if you need to peek at the hint there. Only my best students can do this problem unaided. Once you have seen the hint, try hard to construct the desired explanation without consulting the solution I have written below. Don’t give up too soon. Reread the definitions of linear dependence and linear transformation, if necessary.

After you have written your best attempt at an explanation, check it against the *Study Guide* solution. Also, study the strategy there of how I found the solution. Even if your attempt is quite unsatisfactory, the time spent on this problem is worthwhile, because you will learn more from the solution here.

**31.** To help you use this *Study Guide* properly, I have hidden the solution at the end of the solutions for Section 1.9. *Do not look there until you have followed the instructions above.* (I may not “hide” a solution again, but I wanted this one time to emphasize the importance of working seriously on a problem before checking the solution.)

---

**Mastering Linear Algebra Concepts: Linear Transformation**

Start to form a robust mental image of a linear transformation by preparing a review sheet that covers the following categories:

- **definition**
- **equivalent descriptions**
- **geometric interpretations**
- **special cases**
- **examples and “counterexamples”**
- **connections with other concepts**

<table>
<thead>
<tr>
<th>Category</th>
<th>Page/Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>definition</td>
<td>Page 65</td>
</tr>
<tr>
<td>equivalent descriptions</td>
<td>Equations (4) and (5)</td>
</tr>
<tr>
<td>geometric interpretations</td>
<td>Figs. 1 or 2</td>
</tr>
<tr>
<td>special cases</td>
<td>Matrix transformation: page 64</td>
</tr>
<tr>
<td>examples and “counterexamples”</td>
<td>Superposition, Examples 2-6</td>
</tr>
<tr>
<td></td>
<td>Paragraph before Exercise 1, in this Guide</td>
</tr>
<tr>
<td></td>
<td>Exercises 29, 30, 33</td>
</tr>
<tr>
<td>connections with other concepts</td>
<td>Existence and uniqueness: page 64</td>
</tr>
<tr>
<td></td>
<td>Linear dependence: Exercise 31</td>
</tr>
</tbody>
</table>

**Note:** Exercise 31 should enrich your mental image of linear dependence, so add a note about it to your list for “linear independence”. If your course does not emphasize the next section, turn now to the end of the *Study Guide* material for Section 1.9 and read the box on *Existence and Uniqueness*. 
1.9 MATRIX OF A LINEAR TRANSFORMATION

Every matrix transformation is a linear transformation. This section shows that every linear transformation from \( \mathbb{R}^n \) to \( \mathbb{R}^m \) is a matrix transformation. Chapters 4 and 5 will discuss other examples of linear transformations.

**KEY IDEAS**

A linear transformation \( T: \mathbb{R}^n \rightarrow \mathbb{R}^m \) is completely determined by what it does to the columns of the identity matrix \( I_n \). The \( j \)th column of the standard matrix for \( T \) is \( T(e_j) \), where \( e_j \) is the \( j \)th column of \( I_n \).

There are two ways to compute the standard matrix \( A \). Either compute \( T(e_1), \ldots, T(e_n) \), which is easy to do when \( T \) is described geometrically, as in Exercises 1–14, or fill in the entries of \( A \) by inspection, which is easy to do when \( T \) is described by a formula, as in Exercises 15–22.

Existence and uniqueness questions about the mapping \( x \mapsto Ax \) are determined by properties of \( A \). You should know how this works. The proof of Theorem 11 also applies to linear transformations on the general vector spaces in Chapter 4. Here is another way to understand Theorem 11 (and Theorem 12(b)) using the language of matrix transformations:

Let \( A \) be the standard matrix of \( T \). Then \( T \) is one-to-one if and only if the equation \( Ax = b \) has at most one solution for each \( b \). This happens if and only if every column of \( A \) is a pivot column, which happens if and only if \( Ax = 0 \) has only the trivial solution.

The “if and only if” phrase in Theorem 11 (and in the proof above) was discussed in *A Mathematical Note*, in Section 1.2 of this Guide.

**SOLUTIONS TO EXERCISES**

1. The columns of the standard matrix \( A \) of \( T \) are the images of \( e_1 \) and \( e_2 \). Write these images

   \[
   \begin{bmatrix}
   3 \\
   1 \\
   3 \\
   1 \\
   \end{bmatrix}
   \quad \text{and} \quad
   \begin{bmatrix}
   -5 \\
   2 \\
   0 \\
   0 \\
   \end{bmatrix}
   \]

   vertically: \( T(e_1) = \begin{bmatrix} 3 \\ 1 \\ 3 \\ 1 \end{bmatrix} \) and \( T(e_2) = \begin{bmatrix} -5 \\ 2 \\ 0 \\ 0 \end{bmatrix} \). Then \( A = \begin{bmatrix} T(e_1) & T(e_2) \end{bmatrix} = \begin{bmatrix}
   3 & -5 \\
   1 & 2 \\
   3 & 0 \\
   1 & 0 \\
   \end{bmatrix} \).
7. Follow what happens to \( e_1 \) and \( e_2 \). Since \( e_1 \) is on the unit circle in the plane, it rotates through \(-3\pi/4\) radians into a point on the unit circle that lies in the third quadrant and on the line \( x_2 = x_1 \) (that is, \( y = x \) in more familiar notation). The point \((-1, -1)\) is on the line \( x_2 = x_1 \), but its distance from the origin is \( \sqrt{2} \). So the rotational image of \( e_1 \) is \((-1/\sqrt{2}, -1/\sqrt{2})\). Then this image reflects in the horizontal axis to \((1/\sqrt{2}, 1/\sqrt{2})\).

Similarly, \( e_2 \) rotates into a point on the unit circle that lies in the second quadrant and on the line \( x_2 = -x_1 \), namely, \((1/\sqrt{2}, -1/\sqrt{2})\). Then this image reflects in the horizontal axis to \((1/\sqrt{2}, 1/\sqrt{2})\).

When the two calculations described above are written in vertical vector notation, the transformation’s standard matrix \([T(e_1)] \quad [T(e_2)]\) is easily seen:

\[
e_1 \rightarrow \begin{bmatrix} -1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix} \quad \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix},
\quad e_2 \rightarrow \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix} \quad \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix},
\quad A = \begin{bmatrix} -1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}
\]

Checkpoint: Use an idea from this section to explain why the linear transformation \( T \) that reflects points through the origin, \( T(x_1, x_2) = (-x_1, -x_2) \), is the same as the linear transformation \( R \) that rotates points about the origin in \( \mathbb{R}^2 \) through \( \pi \) radians.

13. Since \((2, 1) = 2e_1 + e_2\), the image of \((2, 1)\) under \( T \) is \( 2T(e_1) + T(e_2) \), by linearity of \( T \). On the figure in the exercise, locate \( 2T(e_1) \) and use it with \( T(e_2) \) to form the parallelogram shown in the text’s answers.

19. The matrix \( A \) that changes \((x_1, x_2, x_3)\) into \((x_1 - 5x_2 + 4x_3, x_2 - 6x_3)\) can be found by inspection when vectors are written in column formation. Write a blank matrix \( A \) to the left of the column vector \( x \) and fill in the entries of \( A \). Since \( T(x) \) has 2 entries, \( A \) has 2 rows. Since \( x \) has 3 entries, \( A \) must have 3 columns.

\[
\begin{bmatrix}
x_1 - 5x_2 + 4x_3 \\
x_2 - 6x_3
\end{bmatrix} = \begin{bmatrix}
A
\end{bmatrix} \begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix} = \begin{bmatrix}
1 & -5 & 4 \\
0 & 1 & -6
\end{bmatrix} \begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix}
\]

21. \( T(x) = \begin{bmatrix}
x_1 + x_2 \\
4x_1 + 5x_2
\end{bmatrix} = \begin{bmatrix}
A
\end{bmatrix} \begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} = \begin{bmatrix}
1 & 1 \\
4 & 5
\end{bmatrix} \begin{bmatrix}
x_1 \\
x_2
\end{bmatrix}. \) To solve \( T(x) = \begin{bmatrix}
3 \\
8
\end{bmatrix} \), row reduce the augmented matrix:

\[
\begin{bmatrix}
1 & 1 & 3 \\
4 & 5 & 8
\end{bmatrix} \sim \begin{bmatrix}
1 & 1 & 3 \\
0 & 1 & -4
\end{bmatrix} \sim \begin{bmatrix}
1 & 0 & 7 \\
0 & 1 & -4
\end{bmatrix}
\]

\( x = \begin{bmatrix}
7 \\
-4
\end{bmatrix}. \)
**Study Tip:** When $T$ is described by a formula, as in Exercises 15–22, you can use the method of Exercise 19 to find an $A$ such that $T(x) = Ax$, provided that $T$ is a linear transformation. (Finding $A$ proves that $T$ is linear.) If you can’t find the matrix, $T$ is probably not a linear transformation. To show that such a $T$ is not linear, you have either to find two vectors $u$ and $v$ such that $T(u + v)$ is not equal to $T(u) + T(v)$ or to find a vector $u$ and scalar $c$ such that $T(cu) \neq cT(u)$.

The text does not give you practice determining whether a transformation is linear because the time needed to develop this skill would have to be taken away from some other topic. If you are expected to have this skill, you will need some exercises (besides Exercises 32 and 33 in Section 1.8). Check with your instructor.

23. a. See Theorem 10.
   b. See Example 3.
   c. See the paragraph before Table 1.
   d. See the definition of onto. Any function from $\mathbb{R}^n$ to $\mathbb{R}^m$ maps each vector onto another vector.
   e. See Example 5.

25. Row reduce the standard matrix $A$ of the transformation $T$ in Exercise 17 to produce

$$
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0
\end{bmatrix}
$$

This matrix shows that $A$ has only three pivot positions, so the equation $Ax = 0$ has a nontrivial solution. By Theorem 11, the transformation $T$ is not one-to-one. Also, since $A$ does not have a pivot in each row, the columns of $A$ do not span $\mathbb{R}^4$. By Theorem 12, $T$ does not map $\mathbb{R}^4$ onto $\mathbb{R}^4$.

31. $T$ is one-to-one if and only if $A$ has $n$ pivot columns. This statement follows by combining Theorem 12(b) with the statement in Exercise 30 of Section 1.7.

---

**A Mathematical Note: One-to-one**

Many students have difficulty with the concept of a one-to-one mapping. Figure 4 should help. The transformation $T$ on the left appears to map three (or even more) points to one image point. In contrast, the transformation $T$ on the right maps three points to one image point. In contrast, the transformation $T$ on the right maps three points to three points. You could say that $T$ is three-to-three (or six-to-six), but the standard terminology is one-to-one.

33. Define $T: \mathbb{R}^n \rightarrow \mathbb{R}^m$ by $T(x) = Bx$ for some $m \times n$ matrix $B$, and let $A$ be the standard matrix for $T$. By definition, $A = [T(e_1) \cdots T(e_n)]$, where $e_j$ is the $j$th column of $I_n$. However, by matrix-vector multiplication,

$$
T(e_j) = Be_j = b_j, \text{ the } j\text{th column of } B. \text{ So } A = [b_1 \cdots b_n] = B.
$$

Copyright © 2016 Pearson Education, Inc.
35. If \( T: \mathbb{R}^n \rightarrow \mathbb{R}^m \) maps \( \mathbb{R}^n \) onto \( \mathbb{R}^m \), then its standard matrix \( A \) has a pivot in each row, by Theorem 12 and by Theorem 4 in Section 1.4. So \( A \) must have at least as many columns as rows, so \( m \leq n \).

When \( T \) is one-to-one, \( A \) must have a pivot in each column, by Theorem 12, so \( m \geq n \).

37. [M] There is no pivot in the fourth column, so the columns of the matrix are not linearly independent and hence the linear transformation is not one-to-one (Theorem 12). (Or, use the result of Exercise 31.)

39. [M] Row reduction of the matrix shows that columns 1, 2, 3, and 5 contain pivots, but there is no pivot in the fifth row, so the columns of the matrix do not span \( \mathbb{R}^5 \). By Theorem 12, the linear transformation is not onto.

31. (This solution is for Section 1.8.) To construct the proof, first write in mathematical terms what is given.

Since \( \{v_1, v_2, v_3\} \) is linearly dependent, there exist scalars \( c_1, c_2, c_3 \), not all zero, such that
\[
c_1v_1 + c_2v_2 + c_3v_3 = 0 (*)
\]

Next, think about what you must prove. In this problem, to prove that the image points are linearly dependent, you need a dependence relation among \( T(v_1), T(v_2), \) and \( T(v_3) \). That fact suggests the next step.

Apply \( T \) to both sides of (*) and use linearity of \( T \), obtaining
\[
T(c_1v_1 + c_2v_2 + c_3v_3) = T(0)
\]

and
\[
c_1T(v_1) + c_2T(v_2) + c_3T(v_3) = 0
\]

Since not all the weights are zero, \( \{T(v_1), T(v_2), T(v_3)\} \) is a linearly dependent set. This completes the proof.

Study Tip: Analyze the strategy above for solving Exercise 31 (in Section 1.8). This approach will work later in a variety of situations.

Answer to Checkpoint: The reflection \( T \) has the property that \( T(e_1) = -e_1 \) and \( T(e_2) = -e_2 \), while the rotation \( R \) has the property that \( R(e_1) = -e_1 \) and \( R(e_2) = -e_2 \). Since a linear transformation is completely determined by what it does to the columns \( e_1 \) and \( e_2 \) of the identity matrix, \( T \) and \( R \) must be the same transformation. (You could also explain this by observing that \( T \) and \( R \) have the same standard matrix, namely, \([-e_1 -e_2]\).)
Mastering Linear Algebra Concepts: Existence and Uniqueness

It’s time to review and organize what you have learned about existence and uniqueness concepts, if you have not already done so. The review will help to prepare you for an exam on the chapter material.

Search through the chapter and collect all the various ways to express existence and uniqueness statements. Most of them can be found in boxes (and theorems) with an “if and only if” statement. Also, check the exercises. For existence, make two lists—one that concerns the equation $Ax = b$ for some fixed $b$ (but not always phrased as a matrix equation), and one that concerns the existence of solutions of $Ax = b$ for all $b$.

1.10 LINEAR MODELS IN BUSINESS, SCIENCE, AND ENGINEERING

This is the second of twelve sections devoted to uses of linear algebra. The applications in the text were selected to give you an impression of the power of linear algebra. You are likely to encounter some of these topics again—in school or in your career—and the discussions in your text will be valuable references.

The main point of this section is to present several interesting applications in which “linearity” arises naturally.

STUDY NOTES

Nutrition Problem: In some applied problems such as the nutrition problem considered here, the data are already organized naturally in a manner that leads to a vector equation of the type we have discussed. The steps to the solution in this case may be diagrammed as follows:

The nutrition model is linear because the nutrients supplied by each foodstuff are proportional to the amount of the foodstuff added to the diet mixture, and each nutrient in the mixture is the sum of the amounts from each foodstuff. Study equations (1) and (2) on page 81.

The nutrition problem leads naturally into linear programming, a topic covered in Chapter 9 that has applications in agriculture, business, engineering, and other areas. In the 1950’s and 1960’s, one of the most common applications of linear algebra (measured in millions of dollars per year for computer time) was to linear programming problems. Such problems are still of great importance in operations research and management science. The following reference gives an entertaining introduction to linear programming. Matrix notation is used in its appendix (pp. 127–152).
Electrical Networks: The linearity of this model, which is evident from the matrix equation $Ri = v$, comes from the linearity of Ohm’s law and Kirchhoff’s voltage law. (Kirchhoff’s current law, which is also linear, is needed when studying another model that involves branch currents.)

Population Movement: The entries in each column of the migration matrix must sum to one because the (decimal) fractions in a column account for the entire population in one region. A certain fraction of the population in a region remains in (or moves within) that region, and other fractions move elsewhere.

**SOLUTIONS TO EXERCISES**

1. **a.** If $x_1$ is the number of servings of Cheerios and $x_2$ is the number of servings of 100% Natural Cereal, then $x_1$ and $x_2$ should satisfy

   \[ \begin{bmatrix} \text{nutrients per serving of Cheerios} \\ \text{nutrients per serving of 100% Natural} \end{bmatrix} + \begin{bmatrix} \text{quantities of nutrients required} \end{bmatrix} = \begin{bmatrix} \text{nutrients quantities per serving} \end{bmatrix} \]

   That is,

   \[
   \begin{align*}
   x_1 \begin{bmatrix} 110 \\ 4 \\ 20 \\ 2 \end{bmatrix} + x_2 \begin{bmatrix} 130 \\ 3 \\ 18 \\ 5 \end{bmatrix} &= \begin{bmatrix} 295 \\ 9 \\ 48 \\ 8 \end{bmatrix} \\
   \begin{bmatrix} 110 \\ 4 \\ 20 \\ 2 \end{bmatrix} + \begin{bmatrix} 130 \\ 3 \\ 18 \\ 5 \end{bmatrix} &= \begin{bmatrix} 295 \\ 9 \\ 48 \\ 8 \end{bmatrix}
   \end{align*}
   \]

   **b.** The equivalent matrix equation is

   \[
   \begin{bmatrix} 110 & 130 \\ 4 & 3 \\ 20 & 18 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 295 \\ 9 \\ 48 \\ 8 \end{bmatrix}
   \]

   To solve this, row reduce the augmented matrix for this equation.

   \[
   \begin{bmatrix} 110 & 130 & 295 \\ 4 & 3 & 9 \\ 20 & 18 & 48 \\ 2 & 5 & 8 \end{bmatrix} \sim \begin{bmatrix} 2 & 5 & 8 \\ 4 & 3 & 9 \\ 20 & 18 & 48 \\ 110 & 130 & 295 \end{bmatrix} \sim \begin{bmatrix} 1 & 2.5 & 4 \\ 0 & 7 & 4 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}
   \]

   The desired nutrients are provided by 1.5 servings of Cheerios together with 1 serving of 100% Natural Cereal.
**Study Tip:** Be sure to distinguish between (i) the vector equation, (ii) the matrix equation (which has the form $A\mathbf{x} = \mathbf{b}$), and (iii) the augmented matrix (which has the form $[A \quad \mathbf{b}]$) that represents a system of linear equations.

7. Loop 1: The resistance vector is

$$r_1 = \begin{bmatrix} 12 \\ -7 \\ 0 \\ -4 \end{bmatrix}$$

- Total of three RI voltage drops for current $I_1$
- Voltage drop for $I_2$ is negative; $I_2$ flows in opposite direction
- Current $I_3$ does not flow in loop 1
- Voltage drop for $I_4$ is negative; $I_4$ flows in opposite direction

Loop 2: The resistance vector is

$$r_2 = \begin{bmatrix} -7 \\ 15 \\ -6 \\ 0 \end{bmatrix}$$

- Voltage drop for $I_1$ is negative; $I_1$ flows in opposite direction
- Total of three RI voltage drops for current $I_2$
- Voltage drop for $I_3$ is negative; $I_3$ flows in opposite direction
- Current $I_4$ does not flow in loop 2

Also, $r_3 = \begin{bmatrix} 0 \\ -6 \\ 14 \\ -5 \end{bmatrix}$, $r_4 = \begin{bmatrix} -4 \\ 0 \\ 13 \end{bmatrix}$, and

$$R = \begin{bmatrix} 12 & -7 & 0 & -4 \\ -7 & 15 & -6 & 0 \\ 0 & -6 & 14 & -5 \\ -4 & 0 & -5 & 13 \end{bmatrix}.$$  

Note that each off-diagonal entry of $R$ is negative (or zero). This happens because the loop current directions are all chosen in the same direction on the figure. (For each loop $j$, this choice forces the currents in other loops adjacent to loop $j$ to flow in the direction opposite to current $I_j$.)

Next, set $\mathbf{v} = \begin{bmatrix} 40 \\ 30 \\ 20 \\ -10 \end{bmatrix}$. Note the negative voltage in loop 4. The current direction chosen in loop 4 is opposed by the orientation of the voltage source in that loop. Thus $R\mathbf{i} = \mathbf{v}$ becomes

$$\begin{bmatrix} 12 & -7 & 0 & -4 \\ -7 & 15 & -6 & 0 \\ 0 & -6 & 14 & -5 \\ -4 & 0 & -5 & 13 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} 40 \\ 30 \\ 20 \\ -10 \end{bmatrix}.$$  

[$\mathbf{M}$]: The solution is $\mathbf{i} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} 11.43 \\ 10.55 \\ 8.04 \\ 5.84 \end{bmatrix}$. 

Copyright © 2016 Pearson Education, Inc.
13. [M] The order of entries in a column of a migration matrix must match the order of the columns. For instance, if the first column concerns the population in the city, then the first entry in each column of the matrix must be the fraction of the population that moves to (or remains in) the city. In this case, the data in the exercise leads to

\[ M = \begin{bmatrix} .95 & .03 \\ .05 & .97 \end{bmatrix} \text{ and } x_0 = \begin{bmatrix} 600,000 \\ 400,000 \end{bmatrix}. \]

a. Some of the population vectors are

\[ x_5 = \begin{bmatrix} 523,293 \\ 476,707 \end{bmatrix}, \quad x_{10} = \begin{bmatrix} 472,737 \\ 527,263 \end{bmatrix}, \quad x_{15} = \begin{bmatrix} 439,417 \\ 560,583 \end{bmatrix}, \quad x_{20} = \begin{bmatrix} 417,456 \\ 582,544 \end{bmatrix}. \]

The data here shows that the city population is declining and the suburban population is increasing, but the changes in population each year seem to grow smaller.

b. When \( x_0 = \begin{bmatrix} 350,000 \\ 650,000 \end{bmatrix} \), the situation is different. Now

\[ x_5 = \begin{bmatrix} 358,523 \\ 641,477 \end{bmatrix}, \quad x_{10} = \begin{bmatrix} 364,140 \\ 635,860 \end{bmatrix}, \quad x_{15} = \begin{bmatrix} 367,843 \\ 632,157 \end{bmatrix}, \quad x_{20} = \begin{bmatrix} 370,283 \\ 629,717 \end{bmatrix}. \]

The city population is increasing slowly and the suburban population is decreasing. No other conclusions are expected. (This example will be analyzed in greater detail later in the text.)

**MATLAB Generating a Sequence**

The m-file (in the Laydata5 Toolbox) for Exercises 9–13 in Section 1.10 stores initial vectors in \( x_0 \). Set \( x = x_0 \) to put the initial data into \( x \). Then use the command \( x = M \times x \) repeatedly to generate the sequence \( x_1, x_2, \ldots \). You only type the command once. After that, use the up-arrow (\( \uparrow \)) key to recall the command, and press <Enter>.

In Exercise 11, you need 6 decimal places to get four significant figures in \( M(1, 2) \). Use the command **format long** and then \( M \) to see more decimal places in \( M \). The command **format short** will return MATLAB to the standard four decimal place display. (The display format does not affect MATLAB’s accuracy in computations.)

Numbers are entered in MATLAB without commas. The number 600,000 in MATLAB scientific notation is 6e5. A small number such as .00000012 is 1.2e-7.
Chapter 1  SUPPLEMENTARY EXERCISES

The supplementary exercises at the end of each chapter review material from the chapter, synthesize concepts from several chapters, or supplement the chapter material in some way. The text has solutions for most of the odd-numbered exercises. The Study Guide provides solutions for selected odd-numbered exercises that have only an answer or a Hint.

In each chapter, Exercise 1 consists of many true/false questions, whose level of difficulty varies. Some are similar to the ones that appear in many sections of the text, in which a word or phrase is sometimes missing or slightly misstated. Some follow fairly easily from a theorem: others may need careful reasoning. A few may require an argument that uses several ideas. In each case, think carefully about the statement and attempt to write a solution. The text provides the true/false answer, but you must supply the justification or counterexample. Careful work on Exercise 1 will help you prepare for an exam over the chapter material.

7. a. Set \( \mathbf{v}_1 = \begin{bmatrix} 2 \\ -5 \\ 7 \end{bmatrix}, \mathbf{v}_2 = \begin{bmatrix} -4 \\ 1 \\ -5 \end{bmatrix}, \mathbf{v}_3 = \begin{bmatrix} -2 \\ 1 \\ -3 \end{bmatrix} \) and \( \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \). “Determine if \( \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \) span \( \mathbb{R}^3 \).”

To do this, row reduce \([ \mathbf{v}_1 \hspace{1em} \mathbf{v}_2 \hspace{1em} \mathbf{v}_3 \]):

\[
\begin{bmatrix}
1 & -2 & -1 \\
-5 & 1 & 1 \\
7 & -5 & -3
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & -2 & -1 \\
0 & -9 & -4 \\
0 & 9 & 4
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & -2 & -1 \\
0 & -9 & -4 \\
0 & 0 & 0
\end{bmatrix}
\]

The matrix does not have a pivot in each row, so its columns do not span \( \mathbb{R}^3 \), by Theorem 4 in Section 1.4.

13. The reduced echelon form of \( \mathbf{A} \) looks like \( \mathbf{E} = \begin{bmatrix} 1 & 0 & * \\ 0 & 1 & * \\ 0 & 0 & 0 \end{bmatrix} \). Since \( \mathbf{E} \) is row equivalent to \( \mathbf{A} \),

\[
\begin{bmatrix}
1 & 0 & * \\
0 & 1 & * \\
0 & 0 & 0
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 0 & * \\
0 & 1 & -2 \\
0 & 0 & 0
\end{bmatrix}
\]

the equation \( \mathbf{Ex} = \mathbf{0} \) has the same solutions as \( \mathbf{Ax} = \mathbf{0} \). Thus

\[
\begin{bmatrix}
1 & 0 & * \\
0 & 1 & 2 \\
0 & 0 & 0
\end{bmatrix}
\]

inspection, the reduced echelon form of \( \mathbf{A} \) is \( \mathbf{E} = \begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix} \).

17. Here are two arguments. The first is a “direct” proof. The second is called a “proof by contradiction.”

i. Since \( \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\} \) is a linearly independent set, \( \mathbf{v}_1 \neq \mathbf{0} \). Also, Theorem 7 shows that \( \mathbf{v}_2 \) cannot be a multiple of \( \mathbf{v}_1 \), and \( \mathbf{v}_3 \) cannot be a linear combination of \( \mathbf{v}_1 \) and \( \mathbf{v}_2 \). By hypothesis, \( \mathbf{v}_4 \) is not a linear combination of \( \mathbf{v}_1, \mathbf{v}_2, \) and \( \mathbf{v}_3 \). Thus, by Theorem 7, \( \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4\} \) cannot be a linearly dependent set and so must be linearly independent.
ii. Since \( \{v_1, v_2, v_3\} \) is a linearly independent set, \( v_1 \neq 0 \). Suppose that \( \{v_1, v_2, v_3, v_4\} \) is linearly dependent. Then, by Theorem 7, one of the vectors in the set is a linear combination of the preceding vectors. This vector cannot be \( v_4 \) because \( v_4 \) is not in \( \text{Span}\{v_1, v_2, v_3\} \). Also, none of the vectors in \( \{v_1, v_2, v_3\} \) is a linear combination of the preceding vectors, by Theorem 7. So the linear dependence of \( \{v_1, v_2, v_3, v_4\} \) is impossible, and \( \{v_1, v_2, v_3, v_4\} \) is linearly independent.

Chapter 1  GLOSSARY CHECKLIST  

Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

affine transformation: A mapping \( T: \mathbb{R}^n \rightarrow \mathbb{R}^m \) of the form \( T(x) = \ldots \).

augmented matrix: A matrix made up of a \ldots.

back-substitution (with matrix notation): The \ldots phase of row reduction of an \ldots.

basic variable: A variable in a linear system that \ldots.

codomain (of \( T: \mathbb{R}^n \rightarrow \mathbb{R}^m \)): The set \ldots that contains \ldots.

coefficient matrix: A matrix whose entries are \ldots.

consistent linear system: A linear system with \ldots.

contraction: A mapping \( x \mapsto \ldots \).

difference equation (or linear recurrence relation): An equation of the form \ldots whose solution is \ldots.

dilation: A mapping \( x \mapsto \ldots \).

domain (of a transformation \( T \)): The set of \ldots.

echelon form (or row echelon form, of a matrix): An echelon matrix that \ldots.

echelon matrix (or row echelon matrix): A rectangular matrix that has three properties: \( (1) \ldots (2) \ldots (3) \ldots \).

elementary row operations: \( (1) \ldots (2) \ldots (3) \ldots \).

equal vectors: Vectors in \( \mathbb{R}^n \) whose \ldots.

equivalent (linear) systems: Linear systems with the \ldots.

existence question: Asks, “Does \ldots exist?” or “Is \ldots?” Also, “Does \ldots exist for \ldots?”

floating point arithmetic: Arithmetic with numbers represented as \ldots.

flop: One arithmetic operation \ldots.

free variable: Any variable in a linear system that \ldots.
Gaussian elimination: See row reduction algorithm.

general solution (of a linear system): A . . . description of a solution set that expresses . . .

homogenous equation: An equation of the . . .

identity matrix (denoted by \( I \) or \( I_n \)): A square matrix . . .

image (of a vector \( \mathbf{x} \) under a transformation \( T \)): The vector . . . (Use symbols)

inconsistent linear system: A linear system with . . .

leading entry: The . . . entry in a row of a matrix.

linear combination: A sum of . . .

linear dependence relation: A . . . equation where . . .

linear equation (in the variables \( x_1, \ldots, x_n \)): An equation that can be written in the form . . .

linearly dependent (vectors): An indexed set \{\mathbf{v}_1, \ldots, \mathbf{v}_p\} with the property that . . .

linearly independent (vectors): An indexed set \{\mathbf{v}_1, \ldots, \mathbf{v}_p\} with the property . . .

linear system: A collection of one or more . . . equations involving . . .

linear transformation: A transformation \( T : \mathbb{R}^n \rightarrow \mathbb{R}^m \) is linear if (i) . . . , and (ii) . . .

line through \( p \) parallel to \( \mathbf{v} \): The set . . . (Use symbols)

matrix: A rectangular . . .

matrix equation: An equation that . . .

matrix transformation: A mapping \( \mathbf{x} \mid\rightarrow . . .

migration matrix: A matrix that gives the . . . movement between different locations, from . . .

\( m \times n \) matrix: A matrix with . . .

nontrivial solution: A nonzero solution of . . .

one-to-one (mapping): A mapping \( T : \mathbb{R}^n \rightarrow \mathbb{R}^n \) such that . . .

onto (mapping): A mapping \( T : \mathbb{R}^n \rightarrow \mathbb{R}^m \) such that . . .

overdetermined system: A system of equations with . . .

parallelogram rule for addition: A geometric interpretation of . . .

parametric equation of a line: An equation of the form . . .

parametric equation of a plane: An equation of the form . . .

pivot: A . . . number that either is used . . . or is . . .

pivot column: A column that . . .

pivot position: A position in a matrix \( A \) that corresponds . . .

plane through \( u, v, \) and the origin: A set whose parametric equation is . . .

product \( Ax \): . . .
range (of a linear transformation $T$): The set of . . .
reduced echelon form (or reduced row echelon form, of a matrix): A rectangular matrix in echelon format that has these additional properties . . .
roundoff error: Error in floating point arithmetic caused when . . .
row-column rule for computing $Ax$: . . .
row equivalent (matrices): Two matrices for which there exists . . .
row reduction algorithm: A systematic method using . . .
row replacement: An elementary row operation that . . .
scalar: . . .
scalar multiple of $u$ by $c$: The vector . . .
set spanned by $\{v_1, \ldots, v_p\}$: . . .
size (of a matrix): Two numbers . . .
solution (of a linear system): . . .
solution set: The set of . . .
Span $\{v_1, \ldots, v_p\}$: The set . . .
standard matrix (for a linear transformation $T$): The matrix . . .
system of linear equations (or a linear system): A collection of . . .
transformation (or function or mapping) $T$ from $\mathbb{R}^n$ to $\mathbb{R}^m$: A rule that assigns to each vector $x$ in $\mathbb{R}^n$ a . . . Notation: $T : \mathbb{R}^n \rightarrow \mathbb{R}^m$.
translation (by a vector $p$): The operation of . . .
trivial solution: The solution . . . of a . . .
underdetermined system: A system of equations with . . .
uniqueness question: Asks, “If a solution of a system . . . ?”
vector: . . .
vector equation: An equation involving . . .
weights: . . .
2.1 MATRIX OPERATIONS

Most of this chapter is an outgrowth of the idea in Section 1.8 that a matrix can transform data. This dynamic role of matrices suggests that we study the combined effect of several matrices on data (that is, on a vector or a set of vectors). Sections 2.1 to 2.5 describe this matrix algebra.

---

**KEY IDEA**

Matrix multiplication corresponds to composition of linear transformations. The definition of $AB$, using the columns of $B$, is critical for the development of both the theory and some of the applications in the text.

---

**STUDY NOTES**

Double-subscript notation: The subscripts tell the location of an entry in the matrix—the first subscript identifies the row and the second subscript the column. (Remember: Row is shorter than column, so row goes first.) This convention is opposite to the way a spreadsheet identifies the location of an entry.

In the product $AB$, left-multiplication (that is, multiplication on the left) by $A$ acts on the columns of $B$, by definition, while right-multiplication by $B$ acts on the rows of $A$ (see page 97). That is,

$$
\text{column } j \text{ of } AB = A \text{ column } j \text{ of } B \quad \text{and} \quad \text{row } i \text{ of } AB = \text{row } i \text{ of } A \times B
$$
To compute a specific matrix product by hand, use the Row-Column Rule. If \( A \) is \( m \times n \), then the \((i,j)\)-entry of \( AB \) is written with sigma notation as

\[
(AB)_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}
\]

Remember that if you change the order (position) of the factors in a matrix product, the new product may be different, or it may not even be defined. For instance, \((A + C)B\) and \(AB + BC\) are probably not equal! Also, see the warning box on page 98.

Notes: Key exercises 13 and 17–22 emphasize the definition of a matrix product. Work at least five of these exercises, for practice.

SOLUTIONS TO EXERCISES

1. \(-2A = (-2) \begin{bmatrix} 2 & 0 & -1 \\ 4 & -5 & 2 \end{bmatrix} = \begin{bmatrix} -4 & 0 & 2 \\ -8 & 10 & -4 \end{bmatrix}\). Next, use \(B - 2A = B + (-2A)\):

\[
B - 2A = \begin{bmatrix} 7 & -5 & 1 \\ 1 & -4 & -3 \end{bmatrix} + \begin{bmatrix} -4 & 0 & 2 \\ -8 & 10 & -4 \end{bmatrix} = \begin{bmatrix} 3 & -5 & 3 \\ -7 & 6 & -7 \end{bmatrix}
\]

The product \(AC\) is not defined because the number of columns of \(A\) does not match the number of rows of \(C\).

\[
CD = \begin{bmatrix} 1 & 2 \\ -2 & 1 \end{bmatrix} \begin{bmatrix} 3 & 5 \\ -1 & 4 \end{bmatrix} = \begin{bmatrix} 1 \cdot 3 + 2(-1) & 1 \cdot 5 + 2 \cdot 4 \\ -2 \cdot 3 + 1(-1) & -2 \cdot 5 + 1 \cdot 4 \end{bmatrix} = \begin{bmatrix} 1 & 13 \\ -7 & -6 \end{bmatrix}
\]

For mental computation, the row-column rule is easier to use than the definition.

7. Since \(A\) has 3 columns, \(B\) must match with 3 rows. Otherwise, \(AB\) is undefined. Since \(AB\) has 7 columns, so does \(B\). Thus, \(B\) is \(3 \times 7\).

13. If you had difficulty with this problem, read the definition of \(AB\) from right to left. Here is the definition, written in reverse order:

\[
[Ab_1 \cdots Ab_p] = A[b_1 \cdots b_p] = AB, \text{ when } B = [b_1 \cdots b_p].
\]

Thus \([Qr_1 \cdots Qr_p] = QR, \text{ when } R = [r_1 \cdots r_p].

15. a. See the definition of \(AB\).  b. See the box after Example 3.

   c. Read Theorem 2(b) from right to left.  d. Read Theorem 3(b) from right to left.

   e. See the box after Theorem 3.

19. A solution is in the text. The main point is that the columns of \(AB\) are \(Ab_1, \ldots, Ab_p\).

Checkpoint: Show that if \(y\) is a linear combination of the columns of \(AB\), then \(y\) is a linear combination of the columns of \(A\).
21. Let \( b_p \) be the last column of \( B \). By hypothesis, the last column of \( AB \) is zero. Thus, \( Ab_p = 0 \). However, \( b_p \) is not the zero vector, because \( B \) has no column of zeros. Thus, the equation \( Ab_p = 0 \) is a linear dependence relation among the columns of \( A \), and so the columns of \( A \) are linearly dependent.

23. If \( x \) satisfies \( Ax = 0 \), then \( CAx = C0 = 0 \) and so \( Inx = 0 \) and \( x = 0 \). This shows that the equation \( Ax = 0 \) has no free variables. So every variable is a basic variable and every column of \( A \) is a pivot column. (A variation of this argument could be made using linear independence and Exercise 30 in Section 1.7.) Since each pivot is in a different row, \( A \) must have at least as many rows as columns.

25. By Exercise 23, the equation \( CA = In \) implies that \( (\text{number of rows in } A) > (\text{number of columns}) \), that is, \( m > n \). By Exercise 26, the equation \( AD = Im \) implies that \( (\text{number of rows in } A) < (\text{number of columns}) \), that is, \( m < n \). Thus \( m = n \). To prove the second statement, observe that \( CAD = C(AD) = CIn = C \) and also \( CAD = (CA)D = InD = D \). Thus \( C = D \). A shorter calculation is \( C = CIn = C(AD) = (CA)D = InD = D \).

29. The \((i,j)\)-entry of \( A(B + C) \) equals the \((i,j)\)-entry of \( AB + AC \), because

\[
\sum_{k=1}^{n} a_{ik} (b_{kj} + c_{kj}) = \sum_{k=1}^{n} a_{ik} b_{kj} + \sum_{k=1}^{n} a_{ik} c_{kj}
\]

The \((i,j)\)-entry of \((B + C)A\) equals the \((i,j)\)-entry of \( BA + CA \), because

\[
\sum_{k=1}^{n} (b_{ik} + c_{ik}) a_{kj} = \sum_{k=1}^{n} b_{ik} a_{kj} + \sum_{k=1}^{n} c_{ik} a_{kj}
\]

31. Use the definition of the product \( I_m A \) and the fact that \( I_m x = x \) for \( x \) in \( \mathbb{R}^m \).

\[
I_m A = I_m [a_1 \ldots a_n] = [I_m a_1 \ldots I_m a_n] = [a_1 \ldots a_n] = A
\]

33. The \((i,j)\)-entry of \( (AB)^T \) is the \((j,i)\)-entry of \( AB \), which is

\[
a_{ji} b_{i1} + \cdots + a_{ji} b_{ni}
\]

The entries in row \( i \) of \( B^T \) are \( b_{1i}, \ldots, b_{ni} \), because they come from column \( i \) of \( B \). Likewise, the entries in column \( j \) of \( A^T \) are \( a_{j1}, \ldots, a_{jn} \), because they come from row \( j \) of \( A \). Thus the \((i,j)\)-entry in \( B^T A^T \) is \( a_{ji} b_{i1} + \cdots + a_{ji} b_{ni} \), as above.

Answer to Checkpoint: If \( y \) is a linear combination of the columns of \( AB \), then there is a vector \( x \) such that \( y = (AB)x \). By definition of matrix multiplication, \( y = A(Bx) \). This expresses \( y \) as a linear combination of the columns of \( A \) using the entries in the vector \( Bx \) as weights.
MATLAB  Matrix Notation and Operations

To create a matrix, enter the data row-by-row, with a space between entries and a semicolon between rows. For instance, the command

\[ \mathbf{A} = [1 \ 2 \ 3; \ 4 \ 5 \ -6] \]

creates a $2 \times 3$ matrix $A$. If $A$ is $m \times n$, then $\text{size}(A)$ is the row vector $[m \ n]$. The $(i, j)$-entry in $A$ is $A(i, j)$. If $i$ or $j$ is replaced by a colon, the result is a column or row of $A$, respectively. Examples:

- $\mathbf{A}(:,3)$
- $\mathbf{A}(2,:)$

To specify columns 3, 4 and 5 of $A$, you can use

$\mathbf{A}(:,[3 \ 4 \ 5])$ or $\mathbf{A}[:,3:5]$

The symbols 3:5 (read “3 to 5”) stand for the vector $[3 \ 4 \ 5]$. Similar notation works for selected rows of $A$.

MATLAB uses $+$, $-$, and $*$ to denote matrix addition, subtraction, and multiplication, respectively. If $A$ is square and $k$ is a positive integer, $A^k$ denotes the $k$th power of $A$. The transpose of $A$ is $A'$ (with an apostrophe for the prime symbol). Note: when $A$ has complex entries, the $(i, j)$-entry of $A'$ is the complex conjugate of the $(j, i)$-entry of $A$.

Use a single column (or row) matrix for a vector. If $\mathbf{u}$ and $\mathbf{v}$ are column vectors of the same size, then $\mathbf{u}^*\mathbf{v}$ is their inner product, and $\mathbf{u}^*\mathbf{v}'$ is an outer product.

MATLAB has commands that construct many special matrices. For example,

- $\mathbf{M} = \text{zeros}(5,6)$  A $5 \times 6$ matrix of zeros
- $\mathbf{M} = \text{ones}(3,5)$  A $3 \times 5$ matrix of ones
- $\mathbf{M} = \text{eye}(6)$  The $6 \times 6$ identity matrix
- $\mathbf{M} = \text{diag}([3 \ 5 \ 7 \ 2 \ 4])$  A $5 \times 5$ diagonal matrix
- $\mathbf{M} = \text{rand}(6)$  A $6 \times 6$ matrix with random entries
- $\mathbf{M} = \text{randomint}(6,4)$  A $6 \times 4$ matrix with random integer entries

Place \texttt{help} in front of any command to learn all the features of the command. The former name for \texttt{randomint} was \texttt{randint}, but MATLAB now uses \texttt{randint} for a slightly different command in its Communications Toolbox.
Matrix inverses are essential for many discussions in linear algebra. This section and the next describe the main properties of invertible matrices.

**STUDY NOTES**

The inverse formula for a $2 \times 2$ matrix will be used frequently in exercises later in the text. (See Theorem 4.) To invert a $2 \times 2$ matrix, interchange the diagonal entries, reverse the signs of the off-diagonal entries, and divide each entry by the determinant (assuming $ad - bc \neq 0$).

Theorem 5 and its proof are important. The phrase “has a unique solution” includes the assertion that a solution exists, so the proof has two parts. The equation $AA^{-1} = I$ is used to prove that a solution exists, and the equation $A^{-1}A = I$ is used to show that the solution is unique.

Except when $A$ is $2 \times 2$, Theorem 5 is practically never used to solve $Ax = b$. Row reduction of $[A \ b]$ is faster. Actually, in practical work, you will seldom need to compute $A^{-1}$. (However, Example 3 illustrates a case in which the entries of $A^{-1}$ could be useful.)

When using an inverse in matrix algebra, remember that matrix multiplication is not commutative. The phrase “left-multiply $B$ by $A^{-1}$” means to multiply $B$ on its left side by $A^{-1}$.

Never write $B/A$ (or $B/A$) because it could stand for $A^{-1}B$ or $BA^{-1}$.

Elementary matrices are used in this text mainly to link row reduction to matrix multiplication. Each elementary row operation amounts to left-multiplication by an elementary matrix. So, if $A$ can be row reduced to $U$, then there is a product $F$ of elementary matrices such that $FA = U$.

Theorem 7 includes an if and only if statement, which was discussed in the Appendix to Section 1.2 in this Study Guide. The proof of this statement in Theorem 7 has two parts: (1) assume that $A$ is invertible and prove that $A \sim I_n$; and (2) assume that $A \sim I_n$ and prove that $A$ is invertible.

**SOLUTIONS TO EXERCISES**

1. \[
\begin{bmatrix}
8 & 6 \\
5 & 4
\end{bmatrix}^{-1} = \frac{1}{32 - 30} \begin{bmatrix} 4 & -6 \\ -5 & 8 \end{bmatrix} = \begin{bmatrix} 2 & -3 \\ -5/2 & 4 \end{bmatrix}
\]

7. a. \[
\begin{bmatrix} 1 & 2 \\ 5 & 12 \end{bmatrix}^{-1} = \frac{1}{1 \cdot 12 - 2 \cdot 5} \begin{bmatrix} 12 & -2 \\ -5 & 1 \end{bmatrix} = \begin{bmatrix} 12 & -2 \\ -5 & 1 \end{bmatrix} \text{ or } \begin{bmatrix} 6 & -1 \\ -2.5 & .5 \end{bmatrix}
\]

\[
x = A^{-1}b_1 = \frac{1}{2} \begin{bmatrix} 12 & -2 \\ -5 & 1 \end{bmatrix} = \begin{bmatrix} 6 & -1 \\ -2 & 0 \end{bmatrix} = \begin{bmatrix} -9 \\ 4 \end{bmatrix}. \text{ Similar calculations give}
\]

\[
A^{-1}b_2 = \begin{bmatrix} 11 & \end{bmatrix}, A^{-1}b_3 = \begin{bmatrix} 6 & \end{bmatrix}, A^{-1}b_4 = \begin{bmatrix} 13 & \end{bmatrix}.
\]

Copyright © 2016 Pearson Education, Inc.
2. b. \[
\begin{bmatrix}
A & b_1 & b_2 & b_3 & b_4
\end{bmatrix} = \begin{bmatrix}
1 & 2 & -1 & 1 & 2 & 3 \\
5 & 12 & 3 & -5 & 6 & 5
\end{bmatrix}
\]
\[
\begin{bmatrix}
1 & 2 & -1 & 1 & 2 & 3 \\
0 & 2 & 8 & -10 & -4 & -10
\end{bmatrix}
\begin{bmatrix}
1 & 2 & -1 & 1 & 2 & 3 \\
0 & 1 & 4 & -5 & -2 & -5
\end{bmatrix}
\begin{bmatrix}
1 & 0 & -9 & 11 & 6 & 13 \\
0 & 1 & 4 & -5 & -2 & -5
\end{bmatrix}
\]

The solutions are \[
\begin{bmatrix}
-9 \\
4
\end{bmatrix}, \begin{bmatrix}
11 \\
-5
\end{bmatrix}, \begin{bmatrix}
6 \\
-2
\end{bmatrix}, \begin{bmatrix}
13 \\
-5
\end{bmatrix}
\]

Note: This exercise was designed to make the arithmetic simple for both methods, but (a) requires more arithmetic than (b). In fact, (a) requires 22 multiplications or divisions and 9 additions or subtractions. In general, the arithmetic for method (b) can be unpleasant for hand calculation. However, when \(A\) is larger than 2×2, method (b) is much faster than (a).

Study Tip: Notice in Exercise 7(a) how the 1/2 in the formula for \(A^{-1}\) was kept outside the matrix \[
\begin{bmatrix}
12 & -2 \\
-5 & 1
\end{bmatrix}
\] when computing \(A^{-1}b\). This trick sometimes simplifies hand calculations (on exams!) by postponing the arithmetic with fractions (or decimals) until the end.

9. a. See the definition of invertible.
   b. See Theorem 6(b).
   c. See Theorem 4.
   d. See Theorem 5.
   e. See the box just before Example 6.

11. (See the proof of Theorem 5.) The \(n \times p\) matrix \(B\) is given (but is arbitrary). Since \(A\) is invertible, the matrix \(A^{-1}B\) satisfies \(AX = B\), because \(A(A^{-1}B) = AA^{-1}B = IB = B\). To show this solution is unique, let \(X\) be any solution of \(AX = B\). Then, left-multiplication of each side by \(A^{-1}\) shows that \(X\) must be \(A^{-1}B\): \(A^{-1}(AX) = A^{-1}B, \ IX = A^{-1}B, \ and \ X = A^{-1}B\).

Study Tip: Whenever you are told “\(A\) is invertible,” you know that \(A^{-1}\) exists, and you may use \(A^{-1}\) to solve an equation or to make appropriate calculations.

13. Left-multiply each side of the equation \(AB = AC\) by \(A^{-1}\) to obtain \(A^{-1}AB = A^{-1}AC, \ IB = IC, \ and \ B = C\).

This conclusion does not always follow when \(A\) is not invertible. The matrices in Exercise 10 of Section 2.1 provide a counterexample.

Warning: A common mistake in Exercise 16 is to try to use the formula \((AB)^{-1} = B^{-1}A^{-1}\). But, this formula can be used only when you know, in advance, that both \(A\) and \(B\) are invertible. In Exercise 16, you must prove that \(A\) is invertible.
19. Unlike Exercise 17, this exercise asks two things, “Does a solution exist?” and “What is the solution?” First, find what the solution must be, if it exists. That is, suppose \( X \) satisfies the equation \( C^{-1}(A + X)B^{-1} = I \). Left-multiply each side by \( C \), and then right-multiply each side by \( B \):

\[
CC^{-1}(A + X)B^{-1} = CI, \quad (A + X)B^{-1} = CB, \quad (A + X)I = CB
\]

Expand the left side and then subtract \( A \) from both sides:

\[
AI + XI = CB, \quad A + X = CB, \quad X = CB - A
\]

If a solution exists, it must be \( CB - A \). To show that \( CB - A \) really is a solution, substitute it for \( X \):

\[
C^{-1}[(A + (CB - A))B^{-1} - C^{-1}(CB)B^{-1} = C^{-1}CBB^{-1} = II = I]
\]

After this section, your instructor may permit you to include fewer details in your calculations. (Check on this.) For instance, after some practice with algebra, an expression such as \( CC^{-1}(A + X)B^{-1} \) could be simplified directly to \( (A + X)B^{-1} \) without first replacing \( CC^{-1} \) by \( I \).

21. Suppose \( A \) is invertible. By Theorem 5, the equation \( Ax = 0 \) has only one solution, namely, the zero solution. This means that the columns of \( A \) are linearly independent, by a remark in Section 1.7.

23. Suppose \( A \) is \( n \times n \) and the equation \( Ax = 0 \) has only the trivial solution. Then there are no free variables in this equation, and so \( A \) has \( n \) pivot columns. Since \( A \) is square and the \( n \) pivot positions must be in different rows, the pivots in an echelon form of \( A \) must be on the main diagonal. Hence \( A \) is row equivalent to the \( n \times n \) identity matrix.

25. Suppose \( A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \) and \( ad - bc = 0 \). If \( a = b = 0 \), then examine

\[
\begin{bmatrix} 0 & 0 \\ c & d \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\]

This has the solution \( x = \begin{bmatrix} d \\ -c \end{bmatrix} \). This solution is nonzero, except when \( c = d = 0 \). In that case, however, \( A \) is the zero matrix, and \( Ax = 0 \) for every vector \( x \). Finally, if \( a \) and \( b \) are not both zero, set \( u = \begin{bmatrix} -b \\ a \end{bmatrix} \). Then

\[
Au = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} -b \\ a \end{bmatrix} = \begin{bmatrix} -ab + ba \\ -cb + da \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \text{ because } -cb + da = 0.
\]

Thus, \( u \) is a nontrivial solution of \( Ax = 0 \). So, in all cases, the equation \( Ax = 0 \) has more than one solution. This is impossible when \( A \) is invertible (by Theorem 5), so \( A \) is not invertible.
27. a. Interchange \( A \) and \( B \) in equation (1) after Example 6 of Section 2.1: \( \text{row}_i(BA) = \text{row}_i(B) \cdot A \). Then replace \( B \) by the identity matrix: \( \text{row}_i(A) = \text{row}_i(IA) = \text{row}_i(I) \cdot A \).

b. Using part (a), when rows 1 and 2 of \( A \) are interchanged, write the result as
\[
\begin{bmatrix}
\text{row}_2(A) \\
\text{row}_1(A) \\
\text{row}_3(A)
\end{bmatrix} = \begin{bmatrix}
\text{row}_2(I) \\
\text{row}_1(I) \\
\text{row}_3(I)
\end{bmatrix} \cdot A = EA
\]

Here, \( E \) is obtained by interchanging rows 1 and 2 of \( I \). The second equality in (*) is a consequence of the fact that \( \text{row}_i(EA) = \text{row}_i(E) \cdot A \).

c. Using part (a), when row 3 of \( A \) is multiplied by 5, write the result as
\[
\begin{bmatrix}
\text{row}_1(A) \\
\text{row}_2(A) \\
5 \cdot \text{row}_3(A)
\end{bmatrix} = \begin{bmatrix}
\text{row}_1(I) \\
\text{row}_2(I) \\
5 \cdot \text{row}_3(I)
\end{bmatrix} \cdot A = EA
\]

Here, \( E \) is obtained by multiplying row 3 of \( I \) by 5.

31. \[
\begin{bmatrix}
A & I
\end{bmatrix} = \begin{bmatrix}
1 & 0 & -2 & 1 & 0 & 0 \\
-3 & 1 & 4 & 0 & 1 & 0 \\
2 & -3 & 4 & 0 & 0 & 1
\end{bmatrix} = \begin{bmatrix}
1 & 0 & -2 & 1 & 0 & 0 \\
0 & 1 & -2 & 3 & 1 & 0 \\
0 & -3 & 8 & -2 & 0 & 1
\end{bmatrix}
\]
\[
\begin{bmatrix}
1 & 0 & -2 & 1 & 0 & 0 \\
0 & 1 & -2 & 3 & 1 & 0 \\
0 & 0 & 2 & 7 & 3 & 1
\end{bmatrix} \sim \begin{bmatrix}
1 & 0 & 0 & 8 & 3 & 1 \\
0 & 1 & 0 & 10 & 4 & 1 \\
0 & 0 & 2 & 7 & 3 & 1
\end{bmatrix}
\]
\[
A^{-1} = \begin{bmatrix}
8 & 3 & 1 \\
10 & 4 & 1 \\
7/2 & 3/2 & 1/2
\end{bmatrix}
\]

33. Let \( B = \begin{bmatrix}
0 & -1 & 1 \\
\vdots & \ddots & \ddots \\
0 & 0 & \cdots & -1 & 1
\end{bmatrix} \), and for \( j = 1, \ldots, n \), let \( a_j, b_j, \) and \( e_j \) denote the \( j \)th columns of \( A, B, \) and \( I \), respectively. Note that for \( j = 1, \ldots, n-1 \), \( a_j - a_{j+1} = e_j \) (because \( a_j \) and \( a_{j+1} \) have the same entries except for the \( j \)th row), \( b_j = e_j - e_{j+1} \) and \( a_n = b_n = e_n \).
To show that $AB = I$, it suffices to show that $A\mathbf{b}_j = \mathbf{e}_j$ for each $j$. For $j = 1, \ldots, n-1$,

$$A\mathbf{b}_j = A(\mathbf{e}_j - \mathbf{e}_{j+1}) = A\mathbf{e}_j - A\mathbf{e}_{j+1} = \mathbf{a}_j - \mathbf{a}_{j+1} = \mathbf{e}_j$$

and $A\mathbf{b}_n = A\mathbf{e}_n = \mathbf{a}_n = \mathbf{e}_n$.

37. There are many possibilities for $C$, but $C = \begin{bmatrix} 1 & 1 & -1 \\ -1 & 1 & 0 \end{bmatrix}$ is the only one whose entries are $1, -1$, and $0$. With only three possibilities for each entry, the construction of $C$ can be done by trial and error. This is probably faster than setting up a system of 4 equations in 6 unknowns. The fact that $A$ cannot be invertible follows from Exercise 25 in Section 2.1, because $A$ is not square.

**MATLAB Constructing $A^{-1}$**

If $A$ is a $5 \times 5$ matrix, then the command $M = [A \text{ eye}(5)]$ creates the augmented matrix $[A \ I]$. Use $\text{gauss}$, $\text{swap}$, $\text{bgauss}$, and $\text{scale}$ to reduce $[A \ I]$. See page 1-14.

MATLAB has other commands that row reduce matrices, invert matrices, and solve equations $Ax = b$. They will be introduced later, after you have studied the concepts and algorithms in this section.

### 2.3 CHARACTERIZATION OF INVERTIBLE MATRICES

In many linear algebra texts, the equivalent of Chapter 4 is extremely difficult for students. But you won’t have problems if you prepare well now, because you are already learning basic ideas that will be presented again in Chapter 4. Review the major concepts from the previous sections, and plan for more study time here than you ordinarily spend on one section.

#### KEY IDEAS

The Invertible Matrix Theorem (IMT) only applies to square matrices. However, some groups of these statements in the IMT are also equivalent for rectangular matrices. The following table will help you remember other important theorems as well as the IMT. (See Theorem 4 in Section 1.4, Theorems 11 and 12 in Section 1.9, Theorem 5 in Section 2.2, and Theorem 9 in Section 2.3.) All of the statements in the table are equivalent when $A$ is square ($m = n = p$).
### Statements from the Invertible Matrix Theorem

<table>
<thead>
<tr>
<th>Equivalent statements for an $m \times n$ matrix $A$.</th>
<th>Equivalent statements for an $n \times n$ square matrix $A$.</th>
<th>Equivalent statements for an $n \times p$ matrix $A$.</th>
</tr>
</thead>
<tbody>
<tr>
<td>k. There is a matrix $D$ such that $AD = I$.</td>
<td>a. $A$ is an invertible matrix.</td>
<td>j. There is a matrix $C$ such that $CA = I$.</td>
</tr>
<tr>
<td>* A has a pivot position in every row.</td>
<td>c. $A$ has $n$ pivot positions.</td>
<td>* A has a pivot position in every column.</td>
</tr>
<tr>
<td>h. The columns of $A$ span $\mathbb{R}^m$.</td>
<td>b. $A$ is row equivalent to the identity matrix.</td>
<td>e. The columns of $A$ are linearly independent.</td>
</tr>
<tr>
<td>g. The equation $Ax = b$ has at least one solution for each $b$ in $\mathbb{R}^m$.</td>
<td>* The equation $Ax = b$ has a unique solution for each $b$ in $\mathbb{R}^n$.</td>
<td>d. The equation $Ax = 0$ has only the trivial solution.</td>
</tr>
<tr>
<td>i. The transformation $x \mapsto Ax$ maps $\mathbb{R}^n$ onto $\mathbb{R}^m$.</td>
<td>* The transformation $x \mapsto Ax$ is invertible.</td>
<td>f. The transformation $x \mapsto Ax$ is one-to-one.</td>
</tr>
</tbody>
</table>

The four statements denoted by (*) were not listed in the text as part of the IMT, mainly to avoid intimidating you with so many statements in one theorem. Note: the text did not actually prove that for a rectangular matrix, statements (j) and (k) are each equivalent to the other statements in their respective columns. (Exercises 23, 24, and 26 in Section 2.1 contain most of the facts needed to prove this.) A matrix $C$ such that $CA = I$ is called a **left-inverse** of $A$, and a matrix $D$ such that $AD = I$ is called a **right-inverse** of $A$.

**Checkpoint**: What can you say about the statements in the first column when $A$ has more rows than columns? (Why?) What about the statements in the third column when $A$ has more columns than rows? (Why?)

A question such as the one in the box below is one way I test whether my students know the IMT. Test yourself. Cover up the IMT, write your answers, and then check your work. The answers are given at the end of this section.

**Test Question:**

Let $A$ be an $n \times n$ matrix. Write 6 statements from the Invertible Matrix Theorem, each equivalent to the statement that $A$ is invertible. Use the following concepts, one in each statement: (i) row equivalent, (ii) the equation $AD = I$, (iii) columns, (iv) the equation $Ax = 0$, and (v) linear transformation.
SOLUTIONS TO EXERCISES

1. The columns of the matrix \[
\begin{pmatrix}
5 & 7 \\
-3 & -6
\end{pmatrix}
\] are not multiples, so they are linearly independent. By (e) in the IMT, the matrix is invertible. Also, the matrix is invertible by Theorem 4 in Section 2.2 because the determinant is nonzero.

\[
\begin{bmatrix}
-1 & -3 & 0 & 1 \\
3 & 5 & 8 & -3 \\
-2 & -6 & 3 & 2 \\
0 & -1 & 2 & 1
\end{bmatrix}
\]

The 4\times4 matrix has four pivot positions and so is invertible by (c) of the IMT.

11. Study the Invertible Matrix Theorem. The statements there are true only for an invertible matrix. Also, if one of the statements is true about a square matrix \(A\), then all statements in the theorem are true; if one of the statements is false, then all are false.

(a) See statements (d) and (b) of the IMT.

(b) See statements (h) and (e).

(c) See statement (g).

(d) See statements (d) and (e).

(e) See statement (1).

Study Tip: Learn how to recognize when a square matrix is not invertible. If \(A\) is an \(n\times n\) matrix, then each of the following statements is true if and only if \(A\) is not invertible.

- The matrix \(A\) has fewer than \(n\) pivot positions.
- The equation \(Ax = 0\) has a nontrivial (nonzero) solution.
- The columns of \(A\) are linearly dependent.
- The linear transformation \(x \mapsto Ax\) is not one-to-one.
- The equation \(Ax = b\) has no solution (is inconsistent) for some \(b\) in \(\mathbb{R}^n\).
- The equation \(Ax = b\) has more than one solution for some \(b\) in \(\mathbb{R}^n\).
- The columns of \(A\) do not span \(\mathbb{R}^n\).
- The linear transformation \(x \mapsto Ax\) does not map \(\mathbb{R}^n\) onto \(\mathbb{R}^n\).

13. If a square upper triangular \(n\times n\) matrix has nonzero diagonal entries, then because it is already in echelon form, the matrix is row equivalent to \(I_n\) and hence is invertible, by the IMT. Conversely, if the matrix is invertible, it has \(n\) pivots on the diagonal and hence the diagonal entries are nonzero.

Study Tip: If you check your answer for odd exercises between 13 and 33, be careful not to read any other answers or hints. You must try to write your own solutions first.
19. By (e) of the IMT, $D$ is invertible. Thus the equation $Dx = b$ has a solution for each $b$ in $\mathbb{R}^7$, by (g) of the IMT. Even better, the equation $Dx = b$ has a unique solution for each $b$ in $\mathbb{R}^7$, by Theorem 5 in Section 2.2. (See the paragraph following the proof of the IMT.)

25. Suppose that $A$ is square and $AB = I$. Then $A$ is invertible, by the (k) of the IMT. Left-multiplying each side of the equation $AB = I$ by $A^{-1}$, one has $A^{-1}AB = A^{-1}I$, $IB = A^{-1}$, and $B = A^{-1}$. By Theorem 6 in Section 2.2, the matrix $B$ (which is $A^{-1}$) is invertible, and its inverse is $(A^{-1})^{-1} = A$. Note: Exercise 25 makes a good test question.

27. Let $W$ be the inverse of $AB$. Then $ABW = I$ and $A(BW) = I$. This equation, by itself, does not prove that $A$ is invertible. However, since $A$ is square, the IMT does apply and by statement (k), $A$ is invertible.

Of course, in this exercise set there is an overall assumption that matrices in this section are square unless otherwise stated. So, with that given, you do not really have to mention here that $A$ is square. However, I put that question “Why not?” in the answer to make you think about this. Look back at Exercise 38 in Section 2.2. There, $AD = I$, which certainly makes $AD$ invertible, yet $A$ is not invertible.

31. Since the equation $Ax = b$ has a solution for each $b$, the matrix $A$ has a pivot in each row (Theorem 4 in Section 1.4). Since $A$ is square, $A$ has a pivot in each column, and so there are no free variables in the equation $Ax = b$, which shows that the solution is unique.

The preceding argument shows that the (square) shape of $A$ plays a crucial role. A less revealing proof is to use the “pivot in each row” and the IMT to conclude that $A$ is invertible. Then Theorem 5 in Section 2.2 shows that the solution of $Ax = b$ is unique.

33. The standard matrix of $T$ is $A = \begin{bmatrix} -5 & 9 \\ 4 & -7 \end{bmatrix}$, which is invertible because $\det A \neq 0$. By Theorem 9, the transformation $T$ is invertible and the standard matrix of $T^{-1}$ is $A^{-1}$. From the formula for a $2 \times 2$ inverse, $A^{-1} = \begin{bmatrix} 7 & 9 \\ 4 & 5 \end{bmatrix}$. So $T^{-1}(x_1, x_2) = \begin{bmatrix} 7 \\ 4 \\ 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = (7x_1 + 9x_2, 4x_1 + 5x_2)$

35. To show that $T$ is one-to-one, suppose that $T(u) = T(v)$ for some vectors $u$ and $v$ in $\mathbb{R}^n$. Then $S(T(u)) = S(T(v))$, where $S$ is the inverse of $T$. By Equation (1), $u = S(T(u))$ and $S(T(v)) = v$, so $u = v$. Thus $T$ is one-to-one. To show that $T$ is onto, suppose $y$ represents an arbitrary vector in $\mathbb{R}^p$ and define $x = S(y)$. Then, using Equation (2), $T(x) = T(S(y)) = y$, which shows that $T$ maps $\mathbb{R}^p$ onto $\mathbb{R}^q$.

Second proof: By Theorem 9, the standard matrix $A$ of $T$ is invertible. By the IMT, the columns of $A$ are linearly independent and span $\mathbb{R}^n$. By Theorem 12 in Section 1.9, $T$ is one-to-one and maps $\mathbb{R}^p$ onto $\mathbb{R}^q$. 

Copyright © 2016 Pearson Education, Inc.
37. Let \( A \) and \( B \) be the standard matrices of \( T \) and \( U \), respectively. Then \( AB \) is the standard matrix of the mapping \( x \mapsto T(U(x)) \), because of the way matrix multiplication is defined (in Section 2.1). By hypothesis, this mapping is the identity mapping, so \( AB = I \). Since \( A \) and \( B \) are square, they are invertible, by the IMT, and \( B = A^{-1} \). Thus, \( BA = I \). This means that the mapping \( x \mapsto U(T(x)) \) is the identity mapping, i.e., \( U(T(x)) = x \) for all \( x \) in \( \mathbb{R}^n \).

**Answers to Checkpoint:** If \( A \) has more rows than columns, then all statements in the first column of the table must be false, because they are equivalent and the statement about a pivot position in each row cannot be true. If \( A \) has more columns than rows, then all statements in the third column of the table must be false, because \( A \) cannot have a pivot in each of its columns.

**Answers to Test Question:** (i) \( A \) is row equivalent to \( I_n \). (ii) There exists an \( n \times n \) matrix \( D \) such that \( AD = I \). (iii) The columns of \( A \) span \( \mathbb{R}^n \). (iv) The equation \( Ax = 0 \) has only the trivial solution. (v) The linear transformation \( x \mapsto Ax \) maps \( \mathbb{R}^n \) onto \( \mathbb{R}^n \).

Another answer for (iii) is: The columns of \( A \) are linearly independent. Similarly, (v) has another answer. But the following statement is unacceptable as one of the answers to the test question:

\[ A \text{ is invertible if and only if the columns of } A \text{ span } \mathbb{R}^n. \]

This statement is itself a (true) theorem (assuming \( A \) is square), not a statement that is true precisely when \( A \) is invertible.

---

**Mastering Linear Algebra: Reviewing and Reflecting**

Two important steps to mastery of linear algebra are periodic review of earlier material and reflection on its relation to new material. When you reread the basic conceptual material from Chapter 1, you may be surprised to discover new insights that you missed earlier. Your broader experience now should give you a better framework within which to understand concepts such as spanning and linear independence.

Compare the review you conducted in Section 1.9 (see the Study Guide appendix to that section) with the three-part table at the beginning of this Study Guide section. (You did carry out that review, didn’t you?) The left and right columns of the table should match some of your “existence” and “uniqueness” statements, respectively.

If your review in Section 1.9 was thorough, you probably anticipated some of the content of the Invertible Matrix Theorem. Existence and uniqueness threads run through the fabric of linear algebra, and they intertwine when related to square matrices (the middle column of the table). A good review procedure now is to expand the table to include references to theorems, examples, and counterexamples. This will occupy several pages. The process of constructing this table is what will help you most.
MATLAB inv, cond, and hilb

Determining whether a matrix is invertible is not always a simple matter. A fast and fairly reliable method is to use the command `inv(A)`, which computes the inverse of \( A \). A warning is given if the matrix is singular (noninvertible) or nearly singular.

For Exercises 41–44, the command `cond(A)` computes the condition number of a matrix \( A \), using what are called the singular values of \( A \) (discussed in Section 7.4.) To perform the experiment described in Exercise 42, you can use the following MATLAB instructions

\[
x = \text{rand}(4,1); \quad b = A \cdot x; \quad x_1 = \text{inv}(A) \cdot b; \quad x - x_1
\]

Use `format long`. Displaying the value of \( x - x_1 \) is the best way to compare \( x \) and \( x_1 \). Press the up-arrow key (↑) to repeat this instruction line.

For Exercise 45, the commands `format rat; hilb(n)` produce the \( n \times n \) Hilbert matrix, with its entries displayed as rational numbers. Enter `format short` to return to the standard display of numbers as decimals.

2.4 PARTITIONED MATRICES

The ideas in this section are fairly simple. However, mark them for future reference, because you are likely to use this notation after you leave school. Partitioned matrices arise in theoretical discussions in essentially every field that makes use of matrices. Here are two examples.

1. The modern state space approach to control systems engineering depends on matrix calculations.\(^1\) The problem of determining whether a system is controllable amounts to calculating the number of pivot positions in a controllability matrix

\[
\begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}
\]

where \( A \) is \( n \times n \), \( B \) has \( n \) rows, and the matrices come from an equation of the form (8) in the discussion preceding Exercise 19.

2. Discussions of modern algorithms and computer software design for scientific computing naturally use the “language” of partitioned matrices. For instance, common techniques for parallel processing of large matrix calculations, such as slicing and crinkling, are described with partitioned matrices.\(^2\) Also, the standard computer science reference on matrix calculations relies heavily on partitioned matrices.\(^3\)

---

\(^1\) An understanding of control systems is important in the design of filtering circuitry, robots, process control systems, and spacecraft. Thus a control systems course is often part of the undergraduate curriculum for electrical, mechanical, chemical, and aerospace engineering. See *Control Systems Engineering*, 3rd ed., by Norman S. Nise, John Wiley & Sons, New York, 2000.


KEY IDEAS

The column-row evaluation of $AB$ is the last of five different “views” of matrix multiplication. All five are special cases of the block matrix version of the row-column rule for matrix multiplication. Here they are:

1. The definition of $Ax$ amounts to block multiplication of $AB$ where $B$ has only one column:

$$Ax = [a_1 | \cdots | a_n] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = [x_1 a_1 + \cdots + x_n a_n]$$

2. Partition $A$ as one row and one column. Then the definition of the usual product $AB$ is a row-column block product:

$$AB = A[b_1 | b_2 | \cdots | b_p] = [A b_1 | A b_2 | \cdots | A b_p]$$

3. Likewise, we observed in Section 2.1 that if $B$ is partitioned as one row and one column, then

$$AB = \begin{bmatrix} \text{row}_1(A) \\ \text{row}_2(A) \\ \vdots \\ \text{row}_n(A) \end{bmatrix} B = \begin{bmatrix} \text{row}_1(A)B \\ \text{row}_2(A)B \\ \vdots \\ \text{row}_n(A)B \end{bmatrix}$$

4. The next display can be viewed either as just the standard row-column rule in which each entry of $AB$ is computed as the product of a row of $A$ and a column of $B$, or as a multiplication of block matrices (with $A$ having only one column of blocks and $B$ having only one row of blocks):

$$AB = \begin{bmatrix} \text{row}_1(A) \\ \text{row}_2(A) \\ \vdots \\ \text{row}_n(A) \end{bmatrix} [\text{col}_1(B) \quad \text{col}_2(B) \quad \cdots \quad \text{col}_p(B)]$$

$$= \begin{bmatrix} \text{row}_1(A)\text{col}_1(B) & \cdots & \text{row}_1(A)\text{col}_p(B) \\ \text{row}_2(A)\text{col}_1(B) & \cdots & \text{row}_2(A)\text{col}_p(B) \\ \vdots & \vdots & \vdots \\ \text{row}_n(A)\text{col}_1(B) & \cdots & \text{row}_n(A)\text{col}_p(B) \end{bmatrix}$$

5. The final display is the column-row expansion of $AB$ (Theorem 10 in this section). In this view, $AB$ is expressed as a sum of outer products of the form $uv^T$, with $u$ a column of $A$ and $v^T$ a row of $B$. But the display can also be viewed as the block version of the row-column product in which $A$ has one row (of blocks) and $B$ has one column (of blocks):
Matrix Algebra

Copyright © 2016 Pearson Education, Inc.

You might say that the row-column rule computes $AB$ as an array of inner products (view 4 above), while the column-row expansion displays $AB$ as a sum of arrays (view 5).

**SOLUTIONS TO EXERCISES**

1. Apply the row-column rule as if the matrix entries were numbers, but for each product (such as $EA$ below), always write the entry of the left block-matrix on the left.

$$
\begin{bmatrix}
I & 0 \\
E & I
\end{bmatrix}
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix}
= 
\begin{bmatrix}
IA + 0C & IB + 0D \\
EA + IC & EB + ID
\end{bmatrix}
= 
\begin{bmatrix}
A & B \\
EA + C & EB + D
\end{bmatrix}
$$

This must be $EA$, not $AE$.

**Checkpoint:** Notice in Exercises 1 and 3 that $\begin{bmatrix} I & 0 \\ E & I \end{bmatrix}$ and $\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$ act as block-matrix generalizations of elementary matrices. What sort of $2 \times 2$ block matrix is the appropriate generalization of an elementary matrix that acts as a scaling operation? (Answer this carefully.)

7. Compute the left side of the equation:

$$
\begin{bmatrix}
X & 0 & 0 \\
Y & 0 & I
\end{bmatrix}
\begin{bmatrix}
A & Z \\
B & I
\end{bmatrix}
= 
\begin{bmatrix}
XA + 0 + 0B & XZ + 0 + 0 \\
YA + 0 + IB & YZ + 0 + I
\end{bmatrix}
$$

Set this equal to the right side of the equation:

$$
\begin{bmatrix}
XA & XZ \\
YA + B & YZ + I
\end{bmatrix}
= 
\begin{bmatrix}
I & 0 \\
0 & I
\end{bmatrix}
$$

so that $XA = I$, $XZ = 0$, $YA + B = 0$, and $YZ + I = I$.

Since the $(1, 1)$-blocks are equal, $XA = I$. Since $X$ and $A$ are square, the IMT implies that $A$ and $X$ are invertible, and hence $X = A^{-1}$. From the $(1, 2)$-entries, $XZ = 0$. Since $X$ is invertible, $Z$ must be $0$. Therefore, the $(2, 2)$-entries give no new information. Finally, from the $(2, 1)$-entries, $YA + B = 0$ and $YA = -B$. Right-multiplication by $A^{-1}$ shows that $Y = -BA^{-1}$. The order of the factors for $Y$ is crucial.

**Study Tip:** Problems such as 5–10 make good exam questions. Remember to mention the IMT when appropriate, and remember that matrix multiplication is generally not commutative.
11. a. See the subsection Addition and Scalar Multiplication.
   b. See the paragraph before Example 3.

13. You are asked to establish an “if and only if” statement. First, suppose that $A$ is invertible, and let $A^{-1} = \begin{bmatrix} D & E \\ F & G \end{bmatrix}$. Then

$$\begin{bmatrix} B & 0 \\ 0 & C \end{bmatrix} \begin{bmatrix} D & E \\ F & G \end{bmatrix} = \begin{bmatrix} BD & BE \\ CF & CG \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

Since $B$ is square, the equation $BD = I$ implies that $B$ is invertible, by the IMT. Similarly, $CG = I$ implies that $C$ is invertible. Also, the equation $BE = 0$ implies that $E = B^{-1}0 = 0$. Similarly $F = 0$. Thus

$$A^{-1} = \begin{bmatrix} B & 0 \\ 0 & C \end{bmatrix}^{-1} = \begin{bmatrix} D & E \\ F & G \end{bmatrix} = \begin{bmatrix} B^{-1} & 0 \\ 0 & C^{-1} \end{bmatrix}$$ (*)&

This proves that $A$ is invertible only if $B$ and $C$ are invertible. For the “if” part of the statement, suppose that $B$ and $C$ are invertible. Then (*) provides a likely candidate for $A^{-1}$ which can be used to show that $A$ is invertible. Compute:

$$\begin{bmatrix} B & 0 \\ 0 & C \end{bmatrix} \begin{bmatrix} B^{-1} & 0 \\ 0 & C^{-1} \end{bmatrix} = \begin{bmatrix} BB^{-1} & 0 \\ 0 & CC^{-1} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

Since $A$ is square, this calculation and the IMT imply that $A$ is invertible. (Don’t forget this final sentence. Without it, the argument is incomplete.) Instead of that sentence, you could add the equation:

$$\begin{bmatrix} B^{-1} & 0 \\ 0 & C^{-1} \end{bmatrix} \begin{bmatrix} B & 0 \\ 0 & C \end{bmatrix} = \begin{bmatrix} B^{-1}B & 0 \\ 0 & C^{-1}C \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

19. The matrix equation (8) in the text is equivalent to

$$(A - sI_n)x + Bu = 0 \quad \text{and} \quad Cx + u = y$$

Rewrite the first equation as $(A - sI_n)x = -Bu$. When $A - sI_n$ is invertible,

$$x = (A - sI_n)^{-1}(-Bu) = -(A - sI_n)^{-1}Bu$$

Substitute this formula for $x$ into the second equation above:

$$C(- (A - sI_n)^{-1}Bu) + u = y, \quad \text{so that} \quad I_nu - C(A - sI_n)^{-1}Bu = y$$

Thus $y = (I_n - C(A - sI_n)^{-1}B)u$. If $W(s) = I_n - C(A - sI_n)^{-1}B$, then $y = W(s)u$. The matrix $W(s)$ is the Schur complement of the matrix $A - sI_n$ in the system matrix in equation (8).

23. To prove a statement by induction, a good first step is to write the statement that depends on $n$ but exclude the phrase “for all $n$,” and label the statement for reference:

The product of two $n \times n$ lower triangular matrices is lower triangular. (*).

Second, verify that the statement is true for $n = 1$. In this particular case, (*) is obviously true, because every $1 \times 1$ matrix is lower triangular. The “induction step” is next.
Suppose that (*) is true when \( n \) is some positive integer \( k \), and consider any \((k+1)\times(k+1)\) lower-triangular matrices \( A_1 \) and \( B_1 \). Partition these matrices as

\[
A_1 = \begin{bmatrix} a & 0^T \\ v & A \end{bmatrix}, \quad B_1 = \begin{bmatrix} b & 0^T \\ w & B \end{bmatrix}
\]

where \( A \) and \( B \) are \( k \times k \) matrices, \( v \) and \( w \) are in \( \mathbb{R}^k \), and \( a \) and \( b \) are scalars. Since \( A_1 \) and \( B_1 \) are lower triangular, so are \( A \) and \( B \). Now

\[
A_1B_1 = \begin{bmatrix} a & 0^T \\ v & A \end{bmatrix}\begin{bmatrix} b & 0^T \\ w & B \end{bmatrix} = \begin{bmatrix} ab + 0^T w & a0^T + 0^T B \\ vb + A w & v0^T + AB \end{bmatrix} = \begin{bmatrix} ab & 0^T \\ bv + A w & AB \end{bmatrix}
\]

Assuming (*) is true for \( n = k \), \( AB \) must be lower triangular. The form of \( A_1B_1 \) shows that it, too, is lower triangular. Thus the statement (*) about lower triangular matrices is true for \( n = k + 1 \) if it is true for \( n = k \). By the principle of induction, (*) is true for all \( n \geq 1 \).

25. First, visualize a partition of \( A \) as a \( 2 \times 2 \) block-diagonal matrix, as below, and then visualize the \((2,2)\) block-entry \( A_{22} \) itself as a block-diagonal matrix. That is,

\[
A = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 3 & 5 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 7 & 8 \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix}, \quad \text{where} \quad A_{22} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 7 & 8 \\ 0 & 5 & 6 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & B \end{bmatrix}
\]

Observe that \( B \) is invertible and \( B^{-1} = \frac{1}{2} \begin{bmatrix} 6 & -8 \\ -5 & 7 \end{bmatrix} = \begin{bmatrix} 3 & -4 \\ -2.5 & 3.5 \end{bmatrix} \). By Exercise 13, the block diagonal matrix \( A_{22} \) is invertible, and

\[
A_{22}^{-1} = \begin{bmatrix} .5 & 0 \\ 3 & -4 \\ -2.5 & 3.5 \end{bmatrix} = \begin{bmatrix} .5 & 0 & 0 \\ 0 & 3 & -4 \\ 0 & -2.5 & 3.5 \end{bmatrix}
\]

Next, observe that \( A_{11} \) is also invertible, with inverse \( \begin{bmatrix} -5 & 2 \\ 3 & -1 \end{bmatrix} \). By Exercise 13, \( A \) itself is invertible, and its inverse is block diagonal:

\[
A^{-1} = \begin{bmatrix} A_{11}^{-1} & 0 \\ 0 & A_{22}^{-1} \end{bmatrix} = \begin{bmatrix} -5 & 2 & 0 & 0 \\ 3 & -1 & 0 & 0 \\ .5 & 0 & 0 & 0 \\ 0 & 3 & -4 & 0 \\ 0 & 0 & 3 & -4 \\ 0 & 0 & 0 & -2.5 \end{bmatrix}
\]

Copyright © 2016 Pearson Education, Inc.
A somewhat less detailed solution would be to write (without formal proof) that the result of Exercise 13 seems to generalize to any block-diagonal matrix. Such a matrix \( A \) is invertible if and only if each of the diagonal blocks is invertible, and the inverse of \( A \) is the block-diagonal matrix formed from the inverses of the diagonal blocks. View the \( 5 \times 5 \) matrix in this exercise as a \( 3 \times 3 \) block matrix:

\[
A = \begin{bmatrix}
1 & 2 & 0 & 0 & 0 \\
3 & 5 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 0 \\
0 & 0 & 0 & 7 & 8 \\
0 & 0 & 0 & 5 & 6 \\
\end{bmatrix}
= \begin{bmatrix}
A_{11} & 0 & 0 \\
0 & A_{22} & 0 \\
0 & 0 & A_{33} \\
\end{bmatrix}
\]

Finish by inverting each of the diagonal blocks and use the results to assemble \( A^{-1} \), as above.

Answer to Checkpoint: The block diagonal matrices \( \begin{bmatrix} E & 0 \\ 0 & I \end{bmatrix} \) and \( \begin{bmatrix} I & 0 \\ 0 & E \end{bmatrix} \) are obvious choices. Less obvious is the requirement that \( E \) be invertible, in order to make these block matrices invertible. (Recall that the invertibility of elementary matrices was essential for the theory in Section 2.2.)

Appendix: The Principle of Induction

Consider a statement “(*)” that depends on a positive integer \( n \), as in Exercise 23. To prove “by induction” that (*) is true for all positive integers, you must prove two things:

(a) Statement (*) is true for \( n = 1 \).

(b) (The induction step) If (*) is true for any positive integer \( n = k \), then (*) is also true for the next integer \( n = k + 1 \).

A property or axiom of the real number system, called the principle of mathematical induction, says that if (a) and (b) are true, then (*) is true for all integers \( n \geq 1 \). This is reasonable, because if (*) is true for \( n = 1 \), then (b) shows that (*) is true for \( n = 2 \). Applying (b) again with \( n = 2 \), we see that (*) is true for \( n = 3 \). Applying (b) repeatedly, we see that (*) is true for 2, 3, 4, 5, . . . .
MATLAB Partitioned Matrices

MATLAB uses partitioned matrix notation. For example, if $A$, $B$, $C$, $D$, $E$, and $F$ are matrices of appropriate sizes, then the command

\[
M = \begin{bmatrix} A & B & C \\ D & E & F \end{bmatrix}
\]

creates a larger matrix of the form $M = \begin{bmatrix} A & B & C \\ D & E & F \end{bmatrix}$. Once $M$ is formed, there is no record of the partition that was used to create $M$. For instance, although $B$ was the $(1, 2)$-block used to form $M$, the number $M(1, 2)$ is the same as the $(1, 2)$-entry of $A$.

2.5 MATRIX FACTORIZATIONS

In a sense, Section 2.5 is the most up-to-date section in the text, because matrix factorizations lie at the heart of modern uses of matrix algebra. For instance, they are indispensable for the analysis of computational algorithms and research in parallel processing. The text focuses here on triangular factorizations, but the exercises introduce you to other important factorizations that you may encounter later.

KEY IDEAS

When a matrix $A$ is factored as $A = LU$, the data in $A$ are preprocessed in a way that makes the equation $Ax = b$ easier to solve. Write $LUx = b$, or $L(Ux) = b$, and let $y = Ux$. Solve $Ly = b$ for $y$ and then solve $Ux = y$ for $x$. The two-step process is fast when $L$ and $U$ are triangular.

Finding $L$ and $U$ requires the same number of multiplications and divisions as row reducing $A$ to an echelon form $U$ (about $n^3/3$ operations when $A$ is $n \times n$). After that, $L$ and $U$ are available for solving other equations involving $A$. The key to finding $L$ is to place entries in $L$ in such a way that the sequence of row operations reducing $A$ to $U$ also reduces $L$ to the identity. In this case, $LU$ must equal $A$. (See the top of page 126.)

The text discusses how to build $L$ when no row interchanges are needed to reduce $A$ to $U$. In this case, $L$ can be unit lower triangular. An appendix below describes how to build $L$ in permuted unit triangular form when row interchanges are needed (or desired, for numerical reasons).
SOLUTIONS TO EXERCISES

1. \[ L = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 2 & -5 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 3 & -7 & -2 \\ 0 & -2 & -1 \\ 0 & 0 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} -7 \\ 5 \end{bmatrix}. \] First, solve \( Ly = b. \)

\[
\begin{bmatrix} L & b \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -7 \\ -1 & 1 & 0 & 5 \\ 2 & -5 & 1 & 2 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 0 & -7 \\ 0 & 1 & 0 & -2 \\ 0 & -5 & 1 & 16 \end{bmatrix}
\]

The only arithmetic is in column 4. \( \sim 0 & 1 & 0 & -2, \) so \( y = \begin{bmatrix} -2 \\ 6 \end{bmatrix}. \)

Next, solve \( Ux = y, \) using back-substitution (with matrix notation).

\[
\begin{bmatrix} U & y \end{bmatrix} = \begin{bmatrix} 3 & -7 & -2 & -7 \\ 0 & -2 & -1 & -2 \\ 0 & 0 & -1 & 6 \end{bmatrix} \sim \begin{bmatrix} 3 & -7 & 0 & -19 \\ 0 & -2 & -1 & -2 \\ 0 & 0 & 1 & -6 \end{bmatrix} \sim \begin{bmatrix} 3 & 0 & 0 & 3 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 1 & -6 \end{bmatrix}
\]

So \( x = (3, 4, -6). \)

Checkpoint: Exercise 12 in Section 2.2 shows how to compute \( A^{-1}B \) by row reduction. Describe how you could speed up this calculation if you have an LU factorization of \( A \) available (and \( A \) is invertible).

7. Place the first pivot column of \( \begin{bmatrix} 2 \\ -3 \\ -4 \end{bmatrix} \) into \( L, \) after dividing the column by 2 (the pivot), then add \( 3/2 \) times row 1 to row 2, yielding \( U. \)
13. A good answer will require a written paragraph or two. If you have not tried to write your answer, do so now, without reading the solution below. Explain how you would row reduce \([A \quad I]\), knowing that \(A\) is lower triangular. Your answer to this question should contain some of the ideas shown below, although your wording might be quite different.

Let \(A\) be a lower-triangular \(n \times n\) matrix with nonzero entries on the diagonal, and consider the augmented matrix \([A \quad I]\).

**a.** The \((1, 1)\)-entry can be scaled to 1 and the entries below it can be changed to 0 by adding multiples of row 1 to the rows below. This affects only the first column of \(A\) and the first column of \(I\). So the \((2, 2)\)-entry in the new matrix is still nonzero and now is the only nonzero entry of row 2 in the first \(n\) columns (because \(A\) was lower triangular).
The (2, 2)-entry can be scaled to 1, and the entries below it can be changed to 0 by adding multiples of row 2 to the rows below. This affects only columns 2 and \( n + 2 \) of the augmented matrix. Now the (3, 3) entry in \( A \) is the only nonzero entry of the third row in the first \( n \) columns, so it can be scaled to 1 and then used as a pivot to zero out entries below it. Continuing in this way, \( A \) is eventually reduced to \( I \), by scaling each row with a pivot and then using only row operations that add multiples of the pivot row to rows below.

b. The row operations just described only add rows to rows below, so the \( I \) on the right in \( [A \ I] \) changes into a lower triangular matrix. By Theorem 7 in Section 2.2, that matrix is \( A^{-1} \).

21. Suppose \( A = BC \), with \( B \) invertible. Then there exist elementary matrices \( E_1, \ldots, E_p \) corresponding to row operations that reduce \( B \) to \( I \), in the sense that \( E_p \cdots E_1 B = I \). Applying the same sequence of row operations to \( A \) amounts to left-multiplying \( A \) by the product \( E_p \cdots E_1 \). By associativity of matrix multiplication,

\[
E_p \cdots E_1 A = E_p \cdots E_1 BC = IC = C
\]

so the same sequence of row operations reduces \( A \) to \( C \).

25. \( A = UDV^T \). Since \( U \) and \( V^T \) are square, the equations \( U^TU = I \) and \( V^TV = I \) imply that \( U \) and \( V^T \) are invertible, by the IMT, and hence \( U^{-1} = U^T \) and \( (V^T)^{-1} = V \). Since the diagonal entries \( \sigma_1, \ldots, \sigma_n \) in \( D \) are nonzero, \( D \) is invertible, with the inverse of \( D \) being the diagonal matrix with \( \sigma_1^{-1}, \ldots, \sigma_n^{-1} \) on the diagonal. Thus \( A \) is a product of invertible matrices. By Theorem 6, \( A \) is invertible and \( A^{-1} = (UDV^T)^{-1} = (V^T)^{-1}D^{-1}U^T = VD^{-1}U^T \).

Answer to Checkpoint: If \( A \) is an invertible \( n \times n \) matrix, with an LU factorization \( A = LU \), and if \( B \) is \( n \times p \), then \( A^{-1}B \) can be computed by first row reducing \( [L \ B] \) to a matrix \( [I \ Y] \) for some \( Y \) and then reducing \( [U \ Y] \) to \( [I \ A^{-1}B] \). One way to see that this algorithm works is to view \( A^{-1}B \) as \( [A^{-1}b_1 \cdots A^{-1}b_p] \) and use the LU algorithm to solve simultaneously the set of equations \( Ax = b_1, \ldots, Ax = b_p \). MATLAB uses this approach to compute \( A^{-1}B \) (after first finding \( L \) and \( U \)).

Appendix: Permutated LU Factorizations

Any \( m \times n \) matrix \( A \) admits a factorization \( A = LU \), with \( U \) in echelon form and \( L \) a permutated unit lower triangular matrix. That is, \( L \) is a matrix such that a permutation (rearrangement) of its rows (using row interchanges) will produce a lower triangular matrix with 1’s on the diagonal.

The construction of \( L \) and \( U \), illustrated below, depends on first using row replacements to reduce \( A \) to a permuted echelon form \( V \) and then using row interchanges to reduce \( V \) to an echelon form \( U \). By watching the reduction of \( A \) to \( V \), we can easily construct a permuted unit lower triangular matrix \( L \) with the property that the sequence of operations changing \( A \) into \( U \) also changes \( L \) into \( I \). This property will guarantee that \( A = LU \). (See the paragraph before Example 2 in the text.)
The following algorithm reduces any matrix to a permuted echelon form. In the algorithm when a row is covered, we ignore it in later calculations.

1. Begin with the leftmost nonzero column. Choose any nonzero entry as the pivot. Designate the corresponding row as a pivot row.

2. Use row replacements to create zeros above and below the pivot (in all uncovered rows). Then cover that pivot row.

3. Repeat steps 1 and 2 on the uncovered submatrix, if any, until all nonzero entries are covered.

This algorithm forces each pivot to be to the right of the preceding pivots; when the rows are rearranged with the pivots in stair-step fashion, all entries below each pivot will be zero. Thus, the algorithm produces a permuted echelon matrix. Whenever a pivot is selected, the column containing the pivot will be used to construct a column of $L$, as we shall see.

As an example, choose any entry in the first column of the following matrix as the first pivot, and use the pivot to create zeros in the rest of column 1. We choose the $(3, 1)$-entry.

\[
A = \begin{bmatrix}
1 & -1 & 5 & -8 & -7 \\
-2 & -1 & -4 & 9 & 1 \\
4 & 8 & -4 & 0 & -8 \\
2 & 3 & 0 & -5 & 3 \\
\end{bmatrix}
\]

Row 3 is the first pivot row. Choose the $(2, 2)$-entry as the second pivot, and create zeros in the rest of column 2, excluding the first pivot row.
Cover row 2 and choose the (4, 4)-entry as the pivot. (The row index of the pivot is relative to the original matrix.) Create zeros in the other rows (in the pivot column), excluding the first two pivot rows.

Let \( V \) denote this permuted echelon form, and permute the rows of \( V \) to create an echelon form. The first pivot row goes to the top, the second pivot row goes next, and so on. The resulting echelon matrix \( U \) is

\[
U = \begin{bmatrix}
4 & 8 & -4 & 0 & -8 \\
0 & 3 & -6 & 9 & -3 \\
0 & 0 & 0 & -2 & 6 \\
0 & 0 & 0 & 5 & \\
\end{bmatrix}
\]

The last step is to create \( L \). Go back and watch the reduction of \( A \) to \( V \). As each pivot is selected, take the pivot column, and divide the pivot into each entry in the column that is not yet in a pivot row. Place the resulting column into \( L \). At the end, fill the holes in \( L \) with zeros.

\[
\begin{pmatrix}
1 \\
-2 \\
4 \\
2 \\
\end{pmatrix}
\begin{pmatrix}
-3 \\
3 \\
-1 \\
-2 \\
\end{pmatrix}
\begin{pmatrix}
1 \\
3 \\
1 \\
1 \\
\end{pmatrix}
\begin{pmatrix}
-5 \\
-6 \\
-4 \\
-8 \\
\end{pmatrix}
\]

\[\begin{align*}
4 & +3 & +2 & +5 \\
1/4 & -1 & -1/2 & 1 \\
-1/2 & 1 & \\
1/2 & -1/3 & 1 & \\
\end{align*}\]

\[L = \begin{bmatrix}
1/4 & -1 & -1/2 & 1 \\
-1/2 & 1 & 0 & 0 \\
1 & 0 & 0 \\
1/2 & -1/3 & 1 & 0
\end{bmatrix}\]

You can check that \( LU = A \). To see why this is so, observe that \( L \) is constructed so the operations that reduce \( A \) to \( V \) also reduce \( L \) to a permuted identity matrix. Since the pivots in \( L \) are in exactly the same rows as in \( V \), the sequence of row interchanges that reduces \( V \) to \( U \) also reduces the permuted identity matrix to \( I \). Thus, the full sequence of operations that
reduces $A$ to $U$ also reduces $L$ to $I$, so that $A = LU$. (See argument at the bottom of page 125 of the text.)

The next example illustrates what to do when $V$ has one or more rows of zeros. The matrix is from the Practice Problem for Section 2.5. For the reduction of $A$ to $V$, pivots were chosen to have the largest possible magnitude (the choice used for “partial pivoting”). Of course, other pivots could have been selected.

\[
A = \begin{bmatrix}
2 & -4 & -2 & 3 \\
6 & -9 & -5 & 8 \\
2 & -7 & -3 & 9 \\
4 & -2 & -2 & -1 \\
-6 & 3 & 3 & 4
\end{bmatrix}
\]

\[
\sim V = \begin{bmatrix}
0 & 0 & 0 & 0 \\
6 & -9 & -5 & 8 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{5}{3} \\
0 & \frac{-6}{2} & 21
\end{bmatrix}
\]

\[
\sim U = \begin{bmatrix}
0 & 0 & 0 & -\frac{5}{3} \\
6 & -9 & -5 & 8 \\
0 & 0 & 0 & \frac{5}{3} \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

The first three columns of $L$ come from the three pivot columns above.

\[
\begin{bmatrix}
2 & 0 & 0 \\
6 & -4 & 0 \\
4 & 4 & 0 \\
-6 & \frac{-6}{2} & 0
\end{bmatrix}
\]

\[
\times \begin{bmatrix}
-1 & 0 & 0 \\
-\frac{5}{3} & 0 & 0 \\
\frac{5}{3} & 0 & 0 \\
\end{bmatrix}
\]

\[
\downarrow \downarrow \downarrow
\]

\[
= \begin{bmatrix}
1/3 & 1/6 & -1 \\
1 & -\frac{2}{3} & 1 \\
1/3 & 2/3 & -1 \\
2/3 & -2/3 & 1
\end{bmatrix}
\]

Copyright © 2016 Pearson Education, Inc.
The matrix $L$ needs two more columns. Use columns 1 and 3 of the $5 \times 5$ identity matrix to place 1’s in the “nonpivot” rows 1 and 3. Fill in the remaining holes with zeros.

\[
\begin{bmatrix}
\frac{1}{3} & \frac{1}{6} & -1 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 \\
\frac{1}{3} & \frac{2}{3} & -1 & 0 & 1 \\
\frac{2}{3} & -\frac{2}{3} & \frac{1}{3} & 0 & 0 \\
-1 & \frac{1}{3} & 0 & 0 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
-1 & 1 & 0 & 0 & 0 \\
\frac{2}{3} & -\frac{2}{3} & 1 & 0 & 0 \\
1/3 & 1/6 & -1 & 1 & 0 \\
1/3 & 2/3 & -1 & 0 & 1
\end{bmatrix} = L
\]

Row reduction of $L$ using only row replacements produces a permuted identity matrix. Moving the 1’s in the “pivot rows” 2, 5, and 4 into rows 1, 2, and 3 of the identity requires the same row swaps as reducing $V$ to $U$. If a further row interchange on the permuted identity is required, it will involve the bottom two rows, which came from the “nonpivot” rows 1 and 3. A corresponding interchange of the bottom two rows of $U$ has no effect on $U$ (and the product $LU$ is unaffected). As a result, $L$ is reduced to $I$ by the same operations that reduce $A$ to $V$ and then to $U$. Check that $A = LU$.

---

**MATLAB  LU Factorization and the Backslash Operator \**

Row reduction of $A$ using the command `gauss` will produce the intermediate matrices needed for an LU factorization of $A$. You can try this on the matrix in Example 2, stored as Exercise 33 in the Laydata Toolbox. The matrices in (5) on page 145 in the text are produced by the commands

- $U=\text{gauss}(A,1)$  
  U has 0’s below the first pivot
- $U=\text{gauss}(U,2)$  
  Now $U$ has 0’s below pivots 1 and 2
- $U=\text{gauss}(U,3)$  
  The echelon form

You can copy the information from the screen onto your paper, and divide by the pivot entries to produce $L$ as in the text. For most text exercises, the pivots are integers and so are displayed accurately.

To construct a permuted LU factorization, use $U=\text{gauss}(U,r,v)$, where $r$ is the row index of the pivot and $v$ is a row vector that lists the rows to be changed by replacement operations. For example, if $A$ has 5 rows and the first pivot is in row 4, use $U=\text{gauss}(A,4,[1 \ 2 \ 3 \ 5])$. If the next pivot is in row 2, use $U=\text{gauss}(U,2,[1 \ 3 \ 5])$. To build the permuted matrix $L$, use full columns from $A$ or the partially reduced $U$, divided by the pivots. Then change entries to zero if they are in a row already selected as a “pivot row.”

The MATLAB command $[L \ U]=\text{lu}(A)$ produces a permuted LU factorization for any square matrix $A$, but it does not handle the general case.
When \( A \) is invertible, the best way to solve \( Ax = b \) with MATLAB is to use the backslash command \( x=A\backslash b \). MATLAB proceeds to compute a permuted LU factorization of \( A \) and then use \( L \) and \( U \) to compute \( x \). The alternative command \( x=\text{inv}(A)*b \) is less efficient and can be less accurate. The command \( \text{inv}(A) \) uses the LU factorization to compute \( A^{-1} \) in the form \( U^{-1}L^{-1} \).

2.6 THE LEONTIEF INPUT-OUTPUT MODEL

If you are in economics, you definitely will need the material in this section for later work. Although most of the discussion concerns economics, the formula for the inverse of \( I - C \) is used in a variety of applications.

STUDY NOTES

The power of Leontief’s model of the economy is that it compresses hundreds of equations in hundreds of variables into the simple matrix equation \( (I - C)x = d \). You should know how to construct the consumption matrix \( C \) and know the algebra that leads from the matrix equation \( x = Cx + d \) to its solution \( x = (I - C)^{-1}d \), under the assumption that the column sums of \( C \) are less than one.

You may need to know the formula (8) for \( (I - C)^{-1} \) on page 135. (Check with your instructor.) The formula is analogous to the formula for the sum of a geometric series of positive numbers:

\[
1 + r + r^2 + r^3 + \cdots = \frac{1}{1 - r} \text{ when } |r| < 1.
\]

SOLUTIONS TO EXERCISES

1. Fill in \( C \) one column at a time, since each column is a unit consumption vector for one sector. Make sure that the order of the sectors is the same for the rows and columns of \( C \). From the way the data are presented, we use the order: manufacturing, agriculture, and services. Read the sentences carefully, to get the data arranged correctly.

<table>
<thead>
<tr>
<th>Purchased from:</th>
<th>Unit consumption vectors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Manuf.</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>.10</td>
</tr>
<tr>
<td>Agriculture</td>
<td>.30</td>
</tr>
<tr>
<td>Services</td>
<td>.30</td>
</tr>
</tbody>
</table>
The intermediate demands created by a production vector $\mathbf{x}$ are given by $C\mathbf{x}$. If agriculture plans to produce 100 units (and the other sectors plan to produce nothing), then the intermediate demand is

$$
C\mathbf{x} = \begin{bmatrix} .10 & .60 & .60 \\ .30 & .20 & .00 \\ .30 & .10 & .10 \end{bmatrix} \begin{bmatrix} 0 \\ 100 \\ 0 \end{bmatrix} = \begin{bmatrix} 60 \\ 20 \\ 10 \end{bmatrix}
$$

7. $C = \begin{bmatrix} .0 & .5 \\ .6 & .2 \end{bmatrix}$, $\mathbf{d} = \begin{bmatrix} 50 \\ 30 \end{bmatrix}$. Let $\mathbf{d}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, the demand for 1 unit of output of sector 1.

a. The production required to satisfy the demand $\mathbf{d}_1$ is the vector $\mathbf{x}_1$ such that $(I - C)\mathbf{x}_1 = \mathbf{d}_1$, namely, $\mathbf{x}_1 = (I - C)^{-1}\mathbf{d}_1$. From Exercise 5,

$$
I - C = \begin{bmatrix} 1 & -0.5 \\ -0.6 & 0.8 \end{bmatrix} \quad \text{and} \quad (I - C)^{-1} = \begin{bmatrix} 1.6 & 1 \\ 1.2 & 2 \end{bmatrix}
$$

so

$$
\mathbf{x}_1 = \begin{bmatrix} 1.6 \\ 1.2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1.6 \\ 1.2 \end{bmatrix}
$$

b. For the final demand $\mathbf{d}_2 = \begin{bmatrix} 51 \\ 30 \end{bmatrix}$, the corresponding production $\mathbf{x}_2$ is given by

$$
\mathbf{x}_2 = (I - C)^{-1}\mathbf{d}_2 = \begin{bmatrix} 1.6 & 1 \\ 1.2 & 2 \end{bmatrix} \begin{bmatrix} 51 \\ 30 \end{bmatrix} = \begin{bmatrix} 111.6 \\ 121.2 \end{bmatrix}
$$

c. From Exercise 5, the production $\mathbf{x}$ corresponding to the demand $\mathbf{d}$ is given by $\mathbf{x} = \begin{bmatrix} 110 \\ 120 \end{bmatrix}$.

Observe from (a) and (b) that $\mathbf{x}_2 = \mathbf{x} + \mathbf{x}_1$. Also, as pointed out in the text, $\mathbf{d}_2 = \mathbf{d} + \mathbf{d}_1$. The sum of the production vectors $\mathbf{x}$ and $\mathbf{x}_1$ gives the production needed to satisfy the sum of the demands $\mathbf{d}$ and $\mathbf{d}_1$. This is expressing the linearity between final demand and production. This relation is true in general, because

$$
\mathbf{x}_2 = (I - C)^{-1}\mathbf{d}_2 = (I - C)^{-1}(\mathbf{d} + \mathbf{d}_1)
$$

$$
= (I - C)^{-1}\mathbf{d} + (I - C)^{-1}\mathbf{d}_1
$$

$$
= \mathbf{x} + \mathbf{x}_1
$$
Warning: In Exercise 9, don’t multiply the consumption matrix $C$ by 10, to get rid of the decimals. That changes the equation $C \mathbf{x} = \mathbf{x} + \mathbf{d}$ into $10C \mathbf{x} = \mathbf{x} + \mathbf{d}$, whose solution is different. However, you may multiply the augmented matrix $[(I - C) \ 0]$ by 10, because the solution of an equation is not affected when both sides are multiplied by a nonzero number.

11. Following the hint in the text, you should obtain $\mathbf{p}^T \mathbf{x} = \mathbf{p}^T C \mathbf{x} + \mathbf{v}^T \mathbf{x}$ (from the price equation). Then, from the production equation, $\mathbf{p}^T \mathbf{x} = \mathbf{p}^T (C \mathbf{x} + \mathbf{d}) = \mathbf{p}^T \mathbf{C} \mathbf{x} + \mathbf{p}^T \mathbf{d}$. Equate the two expressions for $\mathbf{p}^T \mathbf{x}$ to yield $\mathbf{p}^T \mathbf{d} = \mathbf{v}^T \mathbf{x}$.

Another solution: Take transposes in the price equation,

$$\mathbf{p}^T = (C^T \mathbf{p})^T + \mathbf{v}^T = \mathbf{p}^T C + \mathbf{v}^T,$$

so $\mathbf{v}^T = \mathbf{p}^T - \mathbf{p}^T C$ and right-multiply by $\mathbf{x}$ to obtain

$$\mathbf{v}^T \mathbf{x} = \mathbf{p}^T \mathbf{x} - \mathbf{p}^T C \mathbf{x} = \mathbf{p}^T (I - C) \mathbf{x} = \mathbf{p}^T \mathbf{d}$$

From the production equation

13. The data for this exercise are in the Laydata Toolbox. To solve the equation $(I - C) \mathbf{x} = \mathbf{d}$, row reduce the augmented matrix $[(I - C) \ \mathbf{d}]$ rather than compute $(I - C)^{-1}$. (Another reasonable solution method is suggested in Exercise 15.) The numerical solution is given in the text.

2.7 APPLICATIONS TO COMPUTER GRAPHICS

According to my students over the past few years, this section is one of the most interesting application sections in the text, because it shows how matrix calculations, performed millions of times per second, can create the illusion of 3D-motion on a computer screen or in a movie theater. Of course, one short section cannot begin to indicate the vast scope of computer graphics. I encourage you to look at the book by Foley et al., referenced in your text. Chapters 5, 6, and 11 are filled with matrices! The rest of the 1100 pages in the book contains lots of interesting mathematics, detailed discussions of computer algorithms, and scores of spectacular (in some cases, almost unbelievable) color plates.

STUDY NOTES

When a graphical object is represented by a set of polygons, each vertex can be stored as one column of a data matrix $\mathbf{D}$. When a linear transformation $T$ acts on the graphical object, the transformed object is determined by the images of the vertices, because each line segment between vertices is transformed into a line segment between the image vertices. If $\mathbf{A}$ is the matrix of the transformation $T$, then the image vertices are the columns of the matrix $\mathbf{A} \mathbf{D}$, by definition of the product $\mathbf{A} \mathbf{D}$. 

Copyright © 2016 Pearson Education, Inc.
Homogeneous coordinates are needed to make translation act as a linear transformation. Translation by a vector \( \mathbf{p} \) is illustrated by the computation 
\[
\begin{bmatrix} I & \mathbf{p} \\
0^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{x} + \mathbf{p} \end{bmatrix}.
\]
When several transformations are composed, the order of matrix products is important. See Example 6.
For 3D-graphics, homogeneous coordinates are used to compute perspective projections. The text only considers a perspective projection whose center of projection is at \((0, 0, d)\). The matrix for this is displayed just before Example 8. Check with your instructor whether you should memorize this matrix. Examples of good test questions can be found in Exercises 1–8 and 13–16.

**SOLUTIONS TO EXERCISES**

1. From Example 5, the matrix 
\[
\begin{bmatrix}
1 & 0.25 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]
for \( \mathbb{R}^2 \) that the matrix 
\[
\begin{bmatrix}
1 & 0.25 \\
0 & 1
\end{bmatrix}
\]
of Example 2 has on ordinary vectors in \( \mathbb{R}^2 \). Partitioned matrix notation explains why this is true. Let \( A \) be a \( 2 \times 2 \) matrix. The following diagram shows that the action of 
\[
\begin{bmatrix}
A & 0 \\
0 & 1
\end{bmatrix}
\]
on \( \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} \) corresponds to the action of \( A \) on \( \mathbf{x} \).

7. A 60% rotation about the origin in \( \mathbb{R}^2 \) is given by 
\[
\begin{bmatrix}
\cos 60^\circ & -\sin 60^\circ \\
\sin 60^\circ & \cos 60^\circ
\end{bmatrix} = \begin{bmatrix} 1/2 & -\sqrt{3}/2 \\
\sqrt{3}/2 & 1/2
\end{bmatrix},
\]
so the \( 3 \times 3 \) matrix for rotation about \( \begin{bmatrix} 6 \\ 8 \end{bmatrix} \) is
Finally, then, rotate first, translate about the origin by \(-p\),

\[
\begin{bmatrix}
1/2 & \sqrt{3}/2 & -3 + 4\sqrt{3} \\
\sqrt{3}/2 & 1/2 & 4 - 3\sqrt{3} \\
0 & 0 & 1
\end{bmatrix}
\]

13. The answer is given in the text. Notice that the order of the transformations is important. If the translation is done first (that is, if the matrix for the translation is on the right), then

\[
\begin{pmatrix}
A & 0 \\
0 & I + \mathbf{p}^T
\end{pmatrix} =
\begin{pmatrix}
AI + \mathbf{00}^T & A\mathbf{p} + \mathbf{0} \\
0^T & 1
\end{pmatrix}
\]

Here, \(0^T\) is a zero row vector, and so the outer product \(00^T\) is a zero matrix.

19. The matrix \(P\) for the perspective transformation with center of projection at \((0, 0, 10)\) and the data matrix \(D\) using homogeneous coordinates are shown below. The data matrix for the image of the triangle is \(PD\):

\[
PD = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & -1 & 1
\end{bmatrix}
\begin{bmatrix}
4.2 \\
1.2 \\
0 \\
0
\end{bmatrix}
\]

The \(\mathbb{R}^3\) coordinates of the image points come from the top three entries in each column, divided by the corresponding entries in the fourth row.
This section presents the basic ideas from Sections 4.1–4.3 that are needed for Chapters 5 to 7. You should study this section and the next only if your course will omit most or all of Chapter 4. If you reviewed carefully after Sections 1.9 and 2.3, you should be well prepared for the new material here.

KEY IDEAS
There are four fundamental concepts in this section: subspace, column space, null space, and basis. The best mental image for a subspace is a plane in \( \mathbb{R}^3 \) through the origin. The distinguishing feature of such a plane is that the sum of any two vectors in the plane is another vector in the same plane (by the parallelogram rule), and any scalar multiple of any vector in the plane is also in the plane. Other subspaces of \( \mathbb{R}^3 \) are lines through the origin, the zero subspace, and \( \mathbb{R}^3 \) itself.

The main examples of subspaces of \( \mathbb{R}^n \) are column spaces (defined explicitly) and null spaces (defined implicitly). Example 6 is probably the most important example in the section. It illustrates a type of computation needed frequently in Chapters 5 and 7.

Actually, there really is not so much to learn here, because you have already been using these concepts for several weeks, without the terminology. (For instance, the notion of a basis is a combination of the ideas of linear independence and spanning.) But you do need to know the precise definitions of these four terms, and you must move beyond mechanical computations. See “Mastering Linear Algebra Concepts” at the end of this section for help.

SOLUTIONS TO EXERCISES

1. The set is closed under sums but not under multiplication by a negative scalar. A counterexample to the subspace condition is shown at the right. You may also give an algebraic example, such as \( x = (2, 1) \) and \( c = -1 \). Then \( x \) is in \( H \) and \( cx = (-2, -1) \) is not in \( H \). Ask your instructor what type of counterexample would be acceptable if this question were on a test.

7. a. There are three vectors, \( v_1, v_2, \) and \( v_3 \), in the set \( \{ v_1, v_2, v_3 \} \).
   
   b. There are infinitely many vectors in \( \text{Span} \{ v_1, v_2, v_3 \} = \text{Col} \ A \).
   
   c. Deciding whether \( p \) is in \( \text{Col} \ A \) requires calculation:
The equation $Ax = p$ has a solution, so $p$ is in $\text{Col } A$.

13. To produce a vector in $\text{Col } A$, select any column of $A$. For $\text{Nul } A$, solve the equation $Ax = 0$. (Include an augmented column of zeros, to avoid errors.)

The general solution is $x_1 = x_3 - x_4$, and $x_2 = -2x_3 + 4x_4$, with $x_3$ and $x_4$ free. The general solution in parametric vector form is not needed. All that is required here is one nonzero vector. So choose any values for $x_3$ and $x_4$ (not both zero). For instance, set $x_3 = 1$ and $x_4 = 0$ to obtain the vector $(1, -2, 1, 0)$ in $\text{Nul } A$. Another choice, setting $x_3 = 0$ and $x_4 = 1$, might be $(-1, 4, 0, 1)$.

19. No. The vectors cannot be a basis for $\mathbb{R}^3$ because they only span a plane in $\mathbb{R}^3$. Or, point out that the columns of the matrix
cannot possibly span $\mathbb{R}^3$ because the matrix cannot have a pivot in every row. So the columns are not a basis for $\mathbb{R}^3$. Be careful not to say that the vectors are a basis for $\mathbb{R}^2$. They are not in $\mathbb{R}^2$, because they each have three entries.

Warning: $\mathbb{R}^2$ is not a subspace of $\mathbb{R}^3$. The notation $\mathbb{R}^2$ refers explicitly to lists of numbers with exactly two entries. $\mathbb{R}^3$ is the set of all lists of three entries from $\mathbb{R}$.

21. a. Carefully read the definition at the beginning of the section. What is missing?
   b. See the paragraph before Example 4.
   c. See Theorem 12. The numbers $m$ and $n$ need not be equal.
   d. See Example 5.
   e. See the first part of the solution of Example 8.
23. \[ A = \begin{bmatrix} 4 & 5 & 9 & -2 \\ 6 & 5 & 1 & 12 \\ 3 & 4 & 8 & -3 \end{bmatrix} \sim \begin{bmatrix} 1 & 2 & 6 & -5 \\ 0 & 1 & 5 & -6 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \] The echelon form identifies columns 1 and 2 as the pivot columns. A basis for \( \text{Col } A \) uses columns 1 and 2 of \( A \): \[ \begin{bmatrix} 4 \\ 6 \\ 3 \end{bmatrix} \text{, } \begin{bmatrix} 5 \\ 5 \\ 4 \end{bmatrix}. \] This is not the only choice, but it is the “standard” choice. A wrong choice is to select columns 1 and 2 of the echelon form. These columns have zero in the third entry and could not possibly generate the columns displayed in \( A \).

For \( \text{Nul } A \), obtain the reduced (and augmented) echelon form for \( A \mathbf{x} = \mathbf{0} \):
\[
\begin{bmatrix} 1 & 0 & -4 & 7 & 0 \\ 0 & 1 & 5 & -6 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_1 \quad - 4x_3 + 7x_4 = 0
\]
This corresponds to:
\[
x_2 + 5x_3 - 6x_4 = 0.
\]
Solve for the basic variables and write the solution of \( A \mathbf{x} = \mathbf{0} \) in parametric vector form:
\[
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 4 \\ -7 \end{bmatrix} - 4x_3 + 7x_4 = 0
\]
\[
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix} x_3 + \begin{bmatrix} 6 \end{bmatrix} x_4. \] Basis for \( \text{Nul } A \): \[ \begin{bmatrix} 4 \\ -5 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -7 \\ 6 \\ 0 \\ 1 \end{bmatrix} \]

Note: A basis is a set of vectors. For simplicity, the answers here and in the text list the vectors without enclosing the list inside set brackets. Ask your instructor if this format is acceptable.

Warning: A common error is to confuse \( \text{Col } A \) with \( \text{Nul } A \). This happens easily when the definitions of these spaces are not known precisely. Another error is to think that the nonpivot columns of an \( m \times n \) matrix \( A \) form a basis for \( \text{Nul } A \). This is not true in general, even when \( m = n \).

25. \[ A = \begin{bmatrix} 1 & 4 & 8 & -3 & -7 \\ -1 & 2 & 7 & 3 & 4 \\ -2 & 2 & 9 & 5 & 5 \\ 3 & 6 & 9 & -5 & -2 \end{bmatrix} \sim \begin{bmatrix} 1 & 4 & 8 & 0 & 5 \\ 0 & 2 & 5 & 0 & -1 \\ 0 & 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \]
Basis for Col $A$:
\[
\begin{bmatrix}
1 & 4 & -3 \\
-1 & 2 & 3 \\
-2 & 2 & 5 \\
3 & 6 & -5
\end{bmatrix}.
\]

For Nul $A$, obtain the reduced (and augmented) echelon form for $Ax = 0$:
\[
\begin{bmatrix}
1 & 0 & -2 & 0 & 7 & 0 \\
0 & 1 & 2.5 & 0 & -.5 & 0 \\
0 & 0 & 0 & 1 & 4 & 0 \\
0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]
\[
\begin{align*}
x_1 & - 2x_3 + 7x_5 = 0 \\
x_2 + 2.5x_3 - .5x_5 = 0 \\
x_4 + 4x_5 = 0 \\
0 = 0
\end{align*}
\]

The solution in parametric vector form:
\[
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5
\end{bmatrix} = \begin{bmatrix}
2x_3 - 7x_5 \\
-2.5x_3 + .5x_5 \\
x_3 \\
-4x_3 \\
x_3
\end{bmatrix} = \begin{bmatrix}
2 \\
-2.5 \\
1 \\
0 \\
0
\end{bmatrix} x_3 + \begin{bmatrix}
-7 \\
.5 \\
0 \\
-4 \\
1
\end{bmatrix}
\]
\[
\begin{array}{c}
\uparrow \\
u
\end{array}
\quad \begin{array}{c}
\uparrow \\
v
\end{array}
\]

Basis for Nul $A$: \{u, v\}.

Note: This solution illustrates how you can save time on an exam and not copy the ten numbers in the basis vectors for your answer. Just label the basis vectors as $u$ and $v$, and write something such as “$u$ and $v$ form a basis for Nul $A$.” You might ask your instructor if this is acceptable.

Warning: Do not become too attached to the symbols commonly used for certain ideas. For instance, calling a vector “$b$” does not imply that it can only be the “right side” of an equation $Ax = b$. In Exercise 29, you should be looking for a vector $b$ such that $Ab = 0$.

29. A simple construction is to write any nonzero $3 \times 3$ matrix whose columns are obviously linearly dependent, and then make $b$ a vector of weights that come from a linear dependence relation among the columns. For instance, if the first two columns of $A$ are equal, then $a_1 - a_2 + 0a_3 = 0$. So, $b$ could be $(1, -1, 0)$. 

Copyright © 2016 Pearson Education, Inc.
31. The text has an answer. An answer such as “Nul $F$ is a subset of $\mathbb{R}^5$” says something true, but not much. “Nul $F$ is a subspace of $\mathbb{R}^5$” ought to be good for some partial credit, but this fact does not use the information given about the column space of $F$. Probably the best possible answer is that Nul $F$ is a nonzero subspace of $\mathbb{R}^5$.

37. [M] Use the command that produces the reduced echelon form in one step (ref or rref depending on the program). By Theorem 13, the pivot columns of $A$ form a basis for Col $A$.

\[ A = \begin{bmatrix} 3 & -5 & 0 & -1 & 3 \\ -7 & 9 & -4 & 9 & -11 \\ -5 & 7 & -2 & 5 & -7 \\ 3 & -7 & -3 & 4 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 2.5 & -4.5 & 3.5 \\ 0 & 1 & 1.5 & -2.5 & 1.5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \]

Basis for Col $A$: \[
\begin{bmatrix} 3 \\ -7 \\ -5 \\ 3 \end{bmatrix}, \begin{bmatrix} -5 \\ 9 \\ 7 \\ -7 \end{bmatrix}
\]

For Nul $A$, obtain the solution of $Ax = 0$ in parametric vector form:

\[
x_1 + 2.5x_3 - 4.5x_4 + 3.5x_5 = 0 \\
x_2 + 1.5x_3 - 2.5x_4 + 1.5x_5 = 0
\]

Solution: \[
\begin{cases} 
  x_1 = -2.5x_3 + 4.5x_4 - 3.5x_5 \\
  x_2 = -1.5x_3 + 2.5x_4 - 1.5x_5 \\
  x_3, x_4, \text{ and } x_5 \text{ are free}
\end{cases}
\]

\[
x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = x_3 \begin{bmatrix} -2.5 \\ -1.5 \\ 1 \\ 0 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} 4.5 \\ 2.5 \\ 0 \\ 1 \\ 0 \end{bmatrix} + x_5 \begin{bmatrix} -3.5 \\ -1.5 \\ 0 \\ 0 \\ 1 \end{bmatrix} = x_3u + x_4v + x_5w
\]

Basis for Nul $A$: \{u, v, w\}.

---

Mastering Linear Algebra Concepts: Subspace, Column Space, Null Space, Basis

To form strong mental images of a subspace and the two main types of subspaces (Col $A$ and Nul $A$), prepare a review sheet that covers the following categories:

- definitions
- equivalent descriptions

Copyright © 2016 Pearson Education, Inc.
The concept of a basis deserves a separate review sheet. Use the categories below. Also, add notes to your review sheets for Span and Linear Independence that say when a spanning set is a basis for a subspace \( \text{Span}\{v_1, \ldots, v_p\} \) and when a linearly independent set in \( \mathbb{R}^n \) is a basis for \( \mathbb{R}^n \).

- geometric interpretations: Fig. 1, Sentence after Example 1
- special cases: Margin figure on page 147, \( \mathbb{R}^n \) itself, zero subspace
- examples and counterexamples: Examples 1, 2, 3, Exercises 1–4
- typical computations: Examples 4, Practice Problems 1 and 2, Exercises 5–14
- contrast between \( \text{Col} A \) and \( \text{Nul} A \): Table on page 204 (in Chapter 4) and Examples on page 203

MATLAB \( \text{ref} \)

The command \( \text{ref}(A) \) produces the reduced echelon form of \( A \). From that you can write a basis for \( \text{Col} A \) or write the homogeneous equations that describe \( \text{Nul} A \). (Don’t forget that \( A \) is a coefficient matrix, not an augmented matrix.) MATLAB has another command, \( \text{rref} \), which works basically the same as \( \text{ref} \) but is often much slower, because it checks for rational entries in the matrix.

2.9 DIMENSION AND RANK

This section and Section 2.8 cover the ideas from Chapter 4 that you need for Chapters 5–7. There is no need to read this section if your course covers Chapter 4.

KEY IDEAS

The two fundamental concepts in this section are the dimension of a subspace and the rank of a matrix. Coordinate vectors are used to give an intuitive understanding of dimension and a geometric explanation of why a \( k \)-dimensional subspace of \( \mathbb{R}^n \) behaves as if it were \( \mathbb{R}^k \).

The Basis Theorem ties together the concepts of dimension, subspace, linear independence, span, and basis. So does the Invertible Matrix Theorem. Make sure you know the precise wording of the statements in these theorems. If you desire extra review material, you might look at Examples 2, 4, and 5 in Section 4.6 (pages 200 and 202).
The following table lists all statements that are in the Invertible Matrix Theorem at this point in the course. They are arranged in the scheme used in Section 2.3 of this Study Guide. As before, a few extra statements have been added to make the table more symmetrical.

### Statements from the Invertible Matrix Theorem

<table>
<thead>
<tr>
<th>Equivalent statements for an (m \times n) matrix (A).</th>
<th>Equivalent statements for an (n \times n) square matrix (A).</th>
<th>Equivalent statements for an (n \times p) matrix (A).</th>
</tr>
</thead>
<tbody>
<tr>
<td>k. There is a matrix (D) such that (AD = I).</td>
<td>a. (A) is an invertible matrix.</td>
<td>j. There is a matrix (C) such that (CA = I).</td>
</tr>
<tr>
<td>* A has a pivot position in every row.</td>
<td>c. (A) has (n) pivot positions.</td>
<td>* A has a pivot position in every column.</td>
</tr>
<tr>
<td>h. The columns of (A) span (\mathbb{R}^m).</td>
<td>m. The columns of (A) form a basis for (\mathbb{R}^n).</td>
<td>e. The columns of (A) are linearly independent.</td>
</tr>
<tr>
<td>g. The equation (A\mathbf{x} = \mathbf{b}) has at least one solution for each (\mathbf{b}) in (\mathbb{R}^m).</td>
<td>* (A) has a pivot position in every column.</td>
<td>d. The equation (A\mathbf{x} = \mathbf{0}) has only the trivial solution.</td>
</tr>
<tr>
<td>i. The transformation (\mathbf{x} \mapsto A\mathbf{x}) maps (\mathbb{R}^n) onto (\mathbb{R}^m).</td>
<td>* (A) has a pivot position in every column.</td>
<td>f. The transformation (\mathbf{x} \mapsto A\mathbf{x}) is one-to-one.</td>
</tr>
<tr>
<td>n. (\text{Col } A = \mathbb{R}^m).</td>
<td>b. (A) is row equivalent to (I).</td>
<td>q. (\text{Nul } A = {\mathbf{0}}).</td>
</tr>
<tr>
<td>o. (\dim \text{Col } A = m).</td>
<td>l. (A^\top) is invertible.</td>
<td>r. (\dim \text{Nul } A = 0).</td>
</tr>
<tr>
<td>* (\text{rank } A = m).</td>
<td>p. (\text{rank } A = n).</td>
<td>* (\text{rank } A = p).</td>
</tr>
</tbody>
</table>

With many concepts to learn in Sections 2.8 and 2.9, you need to be careful not to combine terms in ways that are undefined, even though they may sound reasonable to you. For example, after you finish your work on this section you should recognize that the following phrases (which have appeared on my students’ papers) are meaningless: “the basis of a matrix,” “the dimension of a basis,” and “the rank of a basis.”

### Solutions to Exercises

1. If \(\begin{bmatrix} x \end{bmatrix}_B = \begin{bmatrix} 3 \\ 2 \end{bmatrix}\), then \(x\) is formed from \(b_1\) and \(b_2\) using weights 3 and 2:
\[ \mathbf{x} = 3\mathbf{b}_1 + 2\mathbf{b}_2 = 3 \begin{bmatrix} 1 \\ 1 \end{bmatrix} + 2 \begin{bmatrix} 2 \\ -1 \end{bmatrix} = \begin{bmatrix} 7 \\ 1 \end{bmatrix} \]

7. The figure in the text suggests that \( \mathbf{w} = 2\mathbf{b}_1 - \mathbf{b}_2 \) and \( \mathbf{x} = 1.5\mathbf{b}_1 + .5\mathbf{b}_2 \), in which case,

\[
[\mathbf{w}]_B = \begin{bmatrix} 2 \\ -1 \end{bmatrix} \quad \text{and} \quad [\mathbf{x}]_B = \begin{bmatrix} 1.5 \\ .5 \end{bmatrix}.
\]

To confirm \([\mathbf{x}]_B\), compute

\[
1.5\mathbf{b}_1 + .5\mathbf{b}_2 = 1.5 \begin{bmatrix} 3 \\ 0 \end{bmatrix} + .5 \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix} = \mathbf{x}
\]

Note: The figures in the text for Exercises 7 and 8 display what Section 4.4 calls \( B \)-graph paper. See page 217.

Study Tip: Exercises 9–16 make good test questions because they do not require much arithmetic. The problem of finding a basis for a null space is particularly important, because this skill is needed throughout Chapters 5 and 7.

13. The four vectors span the column space \( H \) of a matrix that can be reduced to echelon form:

\[
\begin{bmatrix}
1 & -3 & 2 & -4 \\
-2 & 9 & -1 & 5 \\
2 & -6 & 4 & -3 \\
-4 & 12 & 2 & 7
\end{bmatrix}
\begin{align*}
\begin{bmatrix}
1 & -3 & 2 & -4 \\
0 & 0 & 5 & -7 \\
0 & 0 & 0 & 5 \\
0 & 0 & 0 & 0
\end{bmatrix}
\end{align*}
\]

Columns 1, 3, and 4 of the original matrix form a basis for \( H \), so \( \dim H = 3 \).

17. a. Check the definition of coordinates relative to a basis.
   b. Dimension is defined only for a subspace.
   c. See the sentence before Example 3.
   d. See the Rank Theorem.
   e. See the Basis Theorem.

19. The text answer uses the Rank Theorem, which is fine. However, you can also answer Exercises 19–22 without explicit reference to the Rank Theorem. For instance, in Exercise 19, if the null space of a matrix \( A \) is three-dimensional, then the equation \( A\mathbf{x} = \mathbf{0} \) has three free variables, and three of the columns of \( A \) are nonpivot columns. Since a \( 5 \times 7 \) matrix has seven columns, \( A \) must have four pivot columns (which form a basis of \( \text{Col} \ A \)). So \( \text{rank} \ A = \dim \text{Col} \ A = 4 \).

25. The text has the solution. This question provides a good way to test knowledge of the Basis Theorem.

27. a. Start with \( B = [\mathbf{b}_1 \cdots \mathbf{b}_q] \) and \( A = [\mathbf{a}_1 \cdots \mathbf{a}_g] \), where \( g > p \). For \( j = 1, \ldots, g \), the vector \( \mathbf{a}_j \) is in \( W \). Since the columns of \( B \) span \( W \), the vector \( \mathbf{a}_j \) is in the column space of \( B \). That is, \( \mathbf{a}_j = B\mathbf{c}_j \) for some vector \( \mathbf{c}_j \) of weights. Note that \( \mathbf{c}_j \) is in \( \mathbb{R}^p \) because \( B \) has \( p \) columns.
b. Let $C = [c_1, \cdots, c_q]$. Then $C$ is a $p \times q$ matrix because each of the $q$ columns is in $\mathbb{R}^p$. By hypothesis, $q$ is larger than $p$, so $C$ has more columns than rows. By a theorem, the columns of $C$ are linearly dependent and there exists a nonzero vector $u$ in $\mathbb{R}^q$ such that $Cu = 0$.

c. From part (a) and the definition of matrix multiplication

$$A = [a_1, \cdots, a_q] = [Bc_1, \cdots, Bc_q] = BC$$

From part (b), $Au = (BC)u = B(Cu) = B0 = 0$. Since $u$ is nonzero, the columns of $A$ are linearly dependent.

---

**Mastering Linear Algebra Concepts: Dimension and Rank**

The concepts of dimension and rank are relatively simple, but they are used so often later that they deserve a review sheet. Pay attention to how they are used in the sentences of the Invertible Matrix Theorem. “Dimension” is always attached to a subspace (not a matrix or vector or basis), and “rank” is attached to a matrix (not a subspace or other object).

- definitions Page 155
- geometric interpretations Fig. 1, a coordinate system on a 2-dim subspace
- special cases Paragraph before Example 2
- examples and counterexamples Examples 2 and 3
- typical computations Practice Problem 1, Exercises 9–16 and 19–21

---

**MATLAB rank**

You can use `ref(A)` to check the rank of $A$, but roundoff error or an extremely small pivot entry can produce an incorrect echelon form. A more reliable command is `rank(A)`.
Chapter 2  SUPPLEMENTARY EXERCISES  

7. Since \( A^{-1}B \) is the solution of \( AX = B \), row reduction of \( [A \ B] \) to \( [I \ X] \) produces \( X = A^{-1}B \).

See Exercise 15 in Section 2.2. In fact, \( A^{-1}B = \begin{bmatrix} 10 & -1 \\ 9 & 10 \\ -5 & -3 \end{bmatrix} \).

11. c. When \( x_1, \ldots, x_n \) are distinct, the columns of \( V \) are linearly independent, by (b). By the Invertible Matrix Theorem, \( V \) is invertible and its columns span \( \mathbb{R}^n \). So, for every vector \( y = (y_1, \ldots, y_n) \) in \( \mathbb{R}^n \), there is a vector \( c \) such that \( Vc = y \). Let \( p \) be the polynomial whose coefficients are listed in \( c \). Then, by (a), \( p \) is an interpolating polynomial for \( (x_1, y_1), \ldots, (x_n, y_n) \).

17. The text has a solution. In addition, note that it is possible that \( BA \) is invertible. For example, let \( C \) be an invertible \( 4 \times 4 \) matrix and construct \( A = \begin{bmatrix} C^{-1} & 0 \end{bmatrix} \) and \( B = [C^{-1} \ 0] \). Then \( BA = I_4 \), which is invertible.

Chapter 2  GLOSSARY CHECKLIST  

Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

associative law of multiplication:  . . . .

basis (for a subspace \( H \) of \( \mathbb{R}^n \), §2.8): A set \( B = \{v_i, \ldots, v_p\} \) in \( \mathbb{R}^n \) such that:  . . . .

block matrix: See partitioned matrix.

block matrix multiplication: The . . . multiplication of . . . as if . . . .

column space (of an \( m \times n \) matrix \( A \), §2.8): The set \( \text{Col } A \) of . . . .

column sum: The sum of . . . .

commuting matrices: Two matrices \( A \) and \( B \) such that . . . .

composition of linear transformations: A mapping produced by applying . . . .

conformable for block multiplication: Two partitioned matrices \( A \) and \( B \) such that . . . .

collection matrix: A matrix in the . . . model whose columns are . . . .
coordinate vector of \( x \) relative to a basis \( \mathcal{B} = \{ \mathbf{b}_1, \ldots, \mathbf{b}_p \} \) (§2.9): The vector \( [x]_\mathcal{B} \) whose entries \( c_1, \ldots, c_p \) satisfy . . .

diagonal entries (in a matrix): Entries having . . .

diagonal matrix: A square matrix . . . whose entries are . . .

dimension (of a subspace, §2.9): The number . . .

determinant (of \( A \)):
\[
\left[\begin{array}{cc}
a & b \\
c & d
\end{array}\right]
\]
The number . . ., denoted by . . .

distributive laws: (left) . . . (right) . . .

elementary matrix: An invertible matrix that results by . . .

final demand vector (or bill of final demands): The vector \( \mathbf{d} \) in the . . . model that lists . . .

flexibility matrix: A matrix whose \( j \)th column gives . . . of an elastic beam at specified points when . . . is applied at . . .

Householder reflection: A transformation \( \mathbf{x} \mapsto Q\mathbf{x} \), where \( Q = . . . \)

identity matrix: The \( n \times n \) matrix \( I \) or \( I_n \) with . . .

inner product: A matrix product . . . where \( \mathbf{u} \) and \( \mathbf{v} \) are . . .

input-output matrix: See consumption matrix.

input-output model: See Leontief input-output model.

intermediate demands: Demands for goods or services that . . .

inverse (of an \( n \times n \) matrix \( A \)): An \( n \times n \) matrix \( A^{-1} \) such that . . .

invertible linear transformation: A linear transformation \( T: \mathbb{R}^n \rightarrow \mathbb{R}^n \) such that there exists . . .

invertible matrix: A square matrix that . . .

ladder network: An electrical network assembled by connecting . . .

left inverse (of \( A \)): Any rectangular matrix \( C \) such that . . .

Leontief input-output model (or Leontief production equation): The equation . . ., where . . .

lower triangular matrix: A matrix with . . .

lower triangular part (of \( A \)): A . . . matrix whose entries on . . .

LU factorization: The representation of a matrix \( A \) in the form \( A = LU \), where \( L \) is . . . and \( U \) is . . .

main diagonal (of a matrix): The location of the . . .

null space (of an \( m \times n \) matrix \( A \), §2.8): The set \( \text{Nul} \ A \) of all . . .

outer product: A matrix product . . . where \( \mathbf{u} \) and \( \mathbf{v} \) are . . .

partitioned matrix: A matrix whose entries are . . . Sometimes called . . .
permuted lower triangular matrix: A matrix such that . . .
permuted LU factorization: The representation of a matrix \( A \) in the form \( A = LU \), where \( L \) is . . . and \( U \) is . . .
production vector: The vector in the . . . model that lists . . .
rank (of a matrix \( A \), §2.9): . . .
right inverse (of \( A \)): Any rectangular matrix \( C \) such that . . .
row-column rule: The rule for computing a product \( AB \) in which . . .
Schur complement: A certain matrix formed from the blocks of a 2×2 partitioned matrix \( A = [A_{ij}] \). If \( A_{11} \) is invertible, its Schur complement is given by . . . If \( A_{22} \) is invertible, its Schur complement is given by . . .
stiffness matrix: The inverse of a . . . matrix. The \( j \)th column of a stiffness matrix gives . . . at specified points on an elastic beam in order to produce . . .
subspace (of \( \mathbb{R}^n \), §2.8): A subset \( H \) of \( \mathbb{R}^n \) with the properties: . . .
transfer matrix: A matrix \( A \) associated with an electrical circuit having input and output terminals, such that . . .
transpose (of \( A \)): An \( n \times m \) matrix \( A^T \) whose . . . are the corresponding . . . of . . .
unit consumption vector: A column vector in the . . . model that lists . . .
unit lower triangular matrix: A matrix with . . .
upper triangular matrix: A matrix \( U \) with . . .
Vandermonde matrix: An \( n \times n \) matrix \( V \) or its transpose, of the form . . .
This section is relatively short and easy. Some exercises provide computational practice and others allow you to discover properties of determinants to be studied in the next section. You will enjoy Section 3.2 more if you finish your work on this section first.

**KEY IDEAS**
The second paragraph of the section sets the stage for what follows. Read it quickly, without worrying about the details of the row operations. The main idea is that a multiple of the determinant of $A$ is a number that appears along the diagonal of an echelon form of $A$, and this number is nonzero if and only if the matrix is invertible. Later (in Section 3.3) you will see why this idea is important. For now, this $3 \times 3$ case is only used to motivate the definition of $\det A$.

Determinants are defined here via a cofactor expansion along the first row. Since the cofactors involve determinants of smaller matrices, the definition is said to be recursive. For each $n \geq 2$, the determinant of an $n \times n$ matrix is based on the definition of the determinant of an $(n-1) \times (n-1)$ matrix. There are other equivalent definitions of the determinant, but we shall not digress to discuss them.

**Study Tip:** Watch how parentheses are used in Example 2 to avoid a common mistake. The cofactor expansion puts a minus sign in front of $a_{32}$ because $(-1)^{3+2} = -1$. Since $a_{32}$ happens to be negative, the correct term in the expansion is $-(2) \det A_{32}$, not $-2 \det A_{32}$.

**SOLUTIONS TO EXERCISES**
1. By definition, $\det A$ is computed via a cofactor expansion along the first row:

Copyright © 2016 Pearson Education, Inc.
For comparison, a cofactor expansion down the second column yields
\[
\begin{vmatrix}
3 & 0 & 4 \\
2 & 3 & 2 \\
0 & 5 & -1 \\
\end{vmatrix}
= (\begin{vmatrix}
2 & 0 \\
0 & 1 \\
\end{vmatrix}) + (\begin{vmatrix}
3 & 4 \\
0 & -1 \\
\end{vmatrix}) + (\begin{vmatrix}
3 & 4 \\
2 & 2 \\
\end{vmatrix})
\]
\[
= (2 \cdot 1 - 0 \cdot 0) + (3 \cdot (-1) - 4 \cdot 0) + (3 \cdot 2 - 4 \cdot 2)
= 0 + 3(-3 - 0) - 5(6 - 8) = -9 + 10 = 1
\]

**Study Tip:** To save time, omit the zero terms in a cofactor expansion, but be careful to use the proper plus or minus signs with the nonzero terms.

7. By definition,
\[
\begin{vmatrix}
4 & 3 & 0 \\
6 & 5 & 2 \\
9 & 7 & 3 \\
\end{vmatrix}
= 4(\begin{vmatrix}
5 & 2 \\
7 & 3 \\
\end{vmatrix}) - 3(\begin{vmatrix}
6 & 2 \\
9 & 3 \\
\end{vmatrix})
= 4(15 - 14) - 3(18 - 18)
= 4
\]

Using the second column of \(A\) instead,
\[
\begin{vmatrix}
4 & 3 & 0 \\
6 & 5 & 2 \\
9 & 7 & 3 \\
\end{vmatrix}
= -3(\begin{vmatrix}
6 & 2 \\
9 & 3 \\
\end{vmatrix}) + 5(\begin{vmatrix}
4 & 0 \\
9 & 3 \\
\end{vmatrix}) + 7(\begin{vmatrix}
4 & 0 \\
6 & 2 \\
\end{vmatrix})
= -3(18 - 18) + 5(12 - 0) - 7(8 - 0)
= 0 + 60 - 56
= 4
\]

13. Row 2 or column 2 are the best choices because they contain the most zeros. We’ll use row 2. Since the only nonzero entry in that row is 2, the determinant is \((-1)^{2+3} \cdot 2 \cdot A_{23}\).

\[
\det A = (-1)^{2+3} \cdot 2
\]

\[
\begin{vmatrix}
4 & 0 & -1 & 3 & -5 \\
0 & 0 & 2 & 0 & 0 \\
7 & 3 & 6 & 4 & -8 \\
5 & 0 & 5 & 2 & -3 \\
0 & 0 & 9 & -1 & 2 \\
\end{vmatrix}
= (-2) \cdot \begin{vmatrix}
4 & 0 & 3 & -5 \\
7 & 3 & 4 & -8 \\
5 & 0 & 2 & -3 \\
0 & 0 & -1 & 2 \\
\end{vmatrix}
= (-2) \cdot (-1)
\]

Copyright © 2016 Pearson Education, Inc.
The best choice for this $4 \times 4$ determinant is to expand down the second column. Notice that the cofactor associated with the 3 in the $(2, 2)$ position is the $(2, 2)$-cofactor of the $4 \times 4$ matrix. The original location of the “3” in the $5 \times 5$ matrix is irrelevant.

$$
\det A = (-2) \cdot (-1)^{2+2} \cdot (3) \cdot \begin{vmatrix} 4 & 3 & -5 \\ 5 & 2 & -3 \\ 0 & -1 & 2 \end{vmatrix} = -6 \begin{vmatrix} 4 & 3 & -5 \\ 5 & 2 & -3 \\ 0 & -1 & 2 \end{vmatrix}
$$

Finally, use column 1 (although row 3 would work as well).

$$
\det A = (-6) \cdot \begin{vmatrix} 4 & -3 & -5 \\ 2 & -5 & 2 \\ -1 & -1 & 2 \end{vmatrix} + 0
$$

$$
= -6[4(4-3) - 5(6-5)] = -6(4-5) = 6
$$

**Checkpoint:** Try to complete the following statement: “If the $k$th column of the $n \times n$ identity matrix is replaced by a column vector $x$ whose entries are $x_1, ..., x_n$, then the determinant of the resulting matrix is _____.” To discover the answer, compute the determinants of the following matrices:

- **a.**
  \[
  \begin{bmatrix}
  1 & 3 & 0 & 0 \\
  0 & 4 & 0 & 0 \\
  0 & 5 & 1 & 0 \\
  0 & 6 & 0 & 1 \\
  \end{bmatrix}
  \]

- **b.**
  \[
  \begin{bmatrix}
  1 & 0 & 3 & 0 \\
  0 & 1 & 4 & 0 \\
  0 & 0 & 5 & 0 \\
  0 & 0 & 6 & 1 \\
  \end{bmatrix}
  \]

- **c.**
  \[
  \begin{bmatrix}
  1 & 0 & 3 & 0 & 0 \\
  0 & 1 & 4 & 0 & 0 \\
  0 & 0 & 5 & 0 & 0 \\
  0 & 0 & 6 & 1 & 0 \\
  0 & 0 & 7 & 0 & 1 \\
  \end{bmatrix}
  \]

19. \( \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc \) and \( \det \begin{bmatrix} c & d \\ a & b \end{bmatrix} = cb - da = -\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} \).

Interchanging two rows reverses the sign of the determinant, at least for the $2 \times 2$ case. Perhaps this is true for larger matrices.

25. The matrix is triangular, so use Theorem 2.

$$
\det \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & k & 1 \end{bmatrix} = 1 \cdot 1 \cdot 1 = 1 \quad \text{Product of the diagonal entries}
$$
31. A $3 \times 3$ row replacement matrix has one of the following forms:

\[
\begin{bmatrix}
1 & 0 & 0 \\
k & 1 & 0 \\
0 & 0 & 1
\end{bmatrix},
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
k & 0 & 1
\end{bmatrix},
\begin{bmatrix}
1 & k & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix},
\begin{bmatrix}
1 & 0 & k \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix},
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix},
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & k & 1
\end{bmatrix}
\]

In each case the matrix is triangular with 1's on the diagonal, so its determinant equals 1.

The determinant of a row replacement matrix is 1, at least for the $3 \times 3$ case. Perhaps this is true for larger matrices.

37. \[\det A = \det \begin{bmatrix} 3 & 1 \\ 4 & 2 \end{bmatrix} = 3(2) - 1(4) = 6 - 4 = 2.\] Since $5A = \begin{bmatrix} 5 \cdot 3 & 5 \cdot 1 \\ 5 \cdot 4 & 5 \cdot 2 \end{bmatrix}$,
\[\det 5A = (5 \cdot 3)(5 \cdot 2) - (5 \cdot 1)(5 \cdot 4) = 150 - 100 = 50.\]

So, $\det 5A \neq 5 \cdot \det A$. Can you see what the true relation between $\det 5A$ and $\det A$ really is, at least for this example? What about $\det 5A$ for any $2 \times 2$ matrix? Try to guess (and perhaps verify) a formula for $\det rA$, where $r$ is any scalar and $A$ is any $n \times n$ matrix.

39. a. See the paragraph preceding the definition of $\det A$.

b. See the definition of cofactor, preceding Theorem 1.

41. \[\det [u \ v] = \det \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} = 6, \] \[\det [u \ x] = \det \begin{bmatrix} 3 & x \\ 0 & 2 \end{bmatrix} = 6,\] and the areas of the parallelograms determined by $[u \ v]$ and $[u \ x]$ both equal 6. To see why the areas are equal, consider the parallelograms determined by $u = (3, 0)$ and $v = (1, 2)$ and by $u$ and $x = (x, 2)$:

The parallelogram on the left is determined by $u$ and $v$ (and the vertices $u + v$ and $0$). Its base is 3 and its altitude is 2, so the area is $(\text{base})(\text{altitude}) = 6$. The parallelogram on the right, determined by $u$ and $x = (x, 2)$, has the same base. Also, the altitude is 2 for any value of $x$, so the area again equals $3 \cdot 2 = 6$.

Answer to Checkpoint: a. 4 b. 5 c. 5 “If the $k$th column of the $n \times n$ identity matrix is replaced by a column vector $x$ whose entries are $x_1, \ldots, x_n$, then the determinant of the resulting matrix is $x_k$.” Can you explain why this is true? You’ll learn the answer when you begin Section 3.3.
This section presents the main properties of determinants, gives an efficient method of computation, and proves that a matrix is invertible if and only if its determinant is nonzero.

**KEY IDEAS**

It is not surprising that row operations relate nicely to determinants. After all, we found the definition of a $3 \times 3$ determinant by row reducing a $3 \times 3$ matrix. Theorem 3 can be rephrased informally as follows:

- **a.** Adding a multiple of one row (or column) of $A$ to another does not change the determinant.
- **b.** Interchanging two rows (or columns) of $A$ reverses the sign of the determinant.
- **c.** A constant may be factored out of one row (or column) of the determinant of $A$.

The other properties to learn are stated in Theorems 4, 5, and 6, together with the boxed formula for $\det A$ on page 166. Theorem 4 is sometimes stated as: **A square matrix is nonsingular if and only if its determinant is nonzero.** Theorems 4 and 6 will be used extensively in Chapter 5.

Your instructor may or may not want you to know the (multi-) linearity property on page 173. This property is important in more advanced courses but is not used later in the text. **Warning:** in general, $\det(A + B)$ is unequal to $\det A + \det B$.

**SOLUTIONS TO EXERCISES**

1. Rows 1 and 2 are interchanged, so the determinant changes sign.

7. $\begin{bmatrix}
1 & 3 & 0 & 2 \\
-2 & -5 & 7 & 4 \\
3 & 5 & 2 & 1 \\
1 & -1 & 2 & -3
\end{bmatrix} = \begin{bmatrix}
1 & 3 & 0 & 2 \\
0 & 1 & 7 & 8 \\
0 & -4 & 2 & -5 \\
0 & -4 & 2 & 5
\end{bmatrix} = \begin{bmatrix}
1 & 3 & 0 & 2 \\
0 & 1 & 7 & 8 \\
0 & 0 & 30 & 27 \\
0 & 0 & 0 & 0
\end{bmatrix} = 0$

Note, the second array already shows that the determinant is zero, because two rows are equal, as in Example 3.

**Study Tip:** In general, computation of a $3 \times 3$ determinant by row reduction takes 10 multiplications (and divisions), but cofactor expansion only takes 9 multiplications. At $n = 4$, the advantage switches to row reduction, which requires 23 multiplications, cofactor expansion 40 (9 for four $3 \times 3$ determinants, plus 4 multiplications of $a_{ij}$ times $\det A_{ij}$). Often, the best strategy is to combine the two techniques, as in Exercises 11–14.
13. Use row or column operations whenever convenient to create a row or column that has only one nonzero entry. (I recommend using only row operations, because you already have experience with them.) Then use a cofactor expansion to reduce the size of the matrix.

\[
\begin{vmatrix}
2 & 5 & 4 & 1 \\
4 & 7 & 6 & 2 \\
6 & -2 & -4 & 0 \\
-6 & 7 & 7 & 0 \\
\end{vmatrix}
\]

Zero created in column 4

\[
\begin{vmatrix}
2 & 5 & 4 & 1 \\
0 & -3 & -2 & 0 \\
6 & -2 & -4 & 0 \\
-6 & 7 & 7 & 0 \\
\end{vmatrix}
\]

Result of cofactor expansion down column 4

\[
\begin{vmatrix}
0 & -3 & -2 \\
6 & -2 & -4 \\
-6 & 7 & 7 \\
\end{vmatrix}
\]

Zero created in column 1

\[
\begin{vmatrix}
0 & -3 & -2 \\
6 & -2 & -4 \\
0 & 5 & 3 \\
\end{vmatrix}
\]

Result of cofactor expansion down column 1

\[
\begin{vmatrix}
-6 & -3 & -2 \\
5 & 3 \\
\end{vmatrix}
\]

\[= 6 \cdot (-9 + 10) = 6\]

19.\[
\begin{vmatrix}
a & b & c \\
2d + a & 2e + b & 2f + c \\
g & h & i \\
\end{vmatrix}
\]

\[
\begin{vmatrix}
a & b & c \\
2d & 2e & 2f \\
g & h & i \\
\end{vmatrix}
\]

\[= 2 \cdot \begin{vmatrix} a & b & c \\
d & e & f \\
g & h & i \end{vmatrix}\]

\[= 2 \cdot 7 = 14\]

25. By Theorem 4 and the IMT, the set \(\{v_1, v_2, v_3\}\) is linearly independent if and only if \(\det(v_1, v_2, v_3) \neq 0\). Rather than use row operations on \([v_1, v_2, v_3]\), you might choose to expand the determinant by cofactors of the third column:

\[
\begin{vmatrix}
7 & -8 & 7 \\
-4 & 5 & 0 \\
-6 & 7 & -5 \\
\end{vmatrix}
\]

\[= 7\begin{vmatrix} -4 & 5 \\
-6 & 7 \end{vmatrix} + (-5) \begin{vmatrix} 7 & -8 \\
-6 & 7 \end{vmatrix} = 7(-28 + 30) - 5(35 - 32)\]

\[= 7(2) - 5(3) = -1\]

The determinant is nonzero, so the vectors are linearly independent.
Study Tip: For 3×3 matrices, some students tend to prefer the special trick suggested for Exercises 15–18 in Section 3.1, even though in general there are 12 multiplications instead of the 9 multiplications needed for cofactor expansion. Note, however, that numbers in the special method can sometimes be large. For comparison, here are those computations for the matrix studied above in Exercise 25:

\[
\begin{vmatrix}
7 & -8 & 7 \\
-4 & 5 & 0 \\
-6 & 7 & -5
\end{vmatrix}
\]

\[
\begin{align*}
\det [v_1 & \quad v_2 & \quad v_3] = 7(5)(-5) + (-8)(0)(-6) + 7(-4)(7) \\
& - (-6)(5)(7) - 7(0)(7) - (-5)(-4)(-8) \\
& = -175 + 0 + (-196) - (-210) - 0 - (-160) = -1
\end{align*}
\]

27. a. See Theorem 3.
   b. See the paragraph following Example 2.
   c. See the remark following Theorem 4.
   d. See the warning after Example 5.

31. Since the determinant is multiplicative (Theorem 6),
\[
(det A)(det A^{-1}) = det(AA^{-1}) = det I = 1.
\]
So \(det A^{-1} = 1/det A\).

Study Tip: The result of Exercise 31 might be useful on a test.

33. By Theorem 6 (twice), \(det AB = (det A)(det B) = (det B)(det A) = det BA\).

35. By Theorem 5, \(det U^T = det U\). So, by Theorem 6,
\[
det U^T U = (det U^T)(det U) = (det U)^2
\]
If \(U^T U = I\), then \((det U)^2 = det I = 1\), which implies that \(det U = \pm 1\).

37. The solution is in the text. (The determinant of a triangular matrix is the product of the entries on the main diagonal.)

Study Tip: Exercises 15–26, 39, and 40 make good test questions because they check your knowledge of determinant properties without requiring much computation. Exercise 39(b) is the one most likely to be answered incorrectly. What would be the answer to 39(b) if \(A\) were 4×4?

43. Compute \(det A\) by a cofactor expansion down column 3:
\[
det A = (u_1 + v_1) \cdot det A_{13} - (u_2 + v_2) \cdot det A_{23} + (u_3 + v_3) \cdot det A_{33}
\]
\[
= u_1 \cdot det A_{13} - u_2 \cdot det A_{23} + u_3 \cdot det A_{33} + v_1 \cdot det A_{13} - v_2 \cdot det A_{23} + v_3 \cdot det A_{33}
\]
\[
= u_1 \cdot det B_{13} - u_2 \cdot det B_{23} + u_3 \cdot det B_{33} + v_1 \cdot det C_{13} - v_2 \cdot det C_{23} + v_3 \cdot det C_{33}
\]
\[
= det B + det C
\]
45. Suppose $A$ is $m \times n$ with more columns than rows. Then $A^TA$ is $n \times n$ and must be singular. If $A$ is generated with random entries, then $AA^T$ will be nonsingular (invertible) practically all the time. Try to explain why these statements should be true. (Use the IMT.)

**MATLAB Computing Determinants**

To compute $\det A$, set $U = A$ and then repeatedly use the commands $U = \text{gauss}(U, r)$ and $U = \text{swap}(U, r, s)$ as needed to reduce $A$ to an echelon form $U$. Then, except for a $+$ or $-$ sign (depending on how many times you swap rows), the determinant of $A$ is given by the command

$$\prod(\text{diag}(U))$$

The command $\text{diag}(U)$ extracts the diagonal entries of $U$ and places them in a column vector, and $\prod$ computes the product of those entries. You can also use $\det(A)$ to check your work, but the longer sequence of commands helps you think about the process of computing $\det A$.

3.3 **CRAMER’S RULE, VOLUME, AND LINEAR TRANSFORMATIONS**

This section will be a valuable reference for students who plan to take a course in multivariable calculus. Mathematics and statistics majors probably will encounter the material here several times. Also, economics students and engineers (particularly electrical engineers) are likely to need Cramer’s rule and some of the supplementary exercises in later courses.

**KEY IDEAS**

The main results of the section are stated in Theorems 7, 8, 9, and 10. The proof of Theorem 7 is simple and yet involves three important ideas: the definition of a matrix product, the multiplicative property of the determinant, and the evaluation of a determinant by cofactors. Check with your instructor about whether you should be able to reproduce the proof of Theorem 7.

A heuristic proof of Theorem 9 for $2 \times 2$ matrices is given in an appendix at the end of this section. Theorem 10 provides a key idea in calculus and physics needed for the study of double and triple integrals. The determinant used there in calculus is called a *Jacobian*. 
STUDY NOTE

In Exercise 25, you are asked to use Theorem 9 to explain why the determinant of a $3 \times 3$ matrix $A$ is zero if and only if $A$ is not invertible. (A similar explanation holds for the $2 \times 2$ case.) The answer is in the text, so be sure to work on this before looking at the answer section. Work on Exercise 25, even if it is not assigned.

Remember, learning does take place when you think hard about an exercise, even when you are unsuccessful, if you try to look at the problem from different angles, browse back through the text, and perhaps look at earlier exercises. Write your solution, don’t just talk to yourself about what you would write if you had to.

SOLUTIONS TO EXERCISES

1. The system is equivalent to $A\mathbf{x} = \mathbf{b}$, where $A = \begin{bmatrix} 5 & 7 \\ 2 & 4 \end{bmatrix}$ and $\mathbf{b} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$. Write

   $A_1(\mathbf{b}) = \begin{bmatrix} 3 \\ 1 \\ 4 \end{bmatrix}$, \hspace{1cm} A_2(\mathbf{b}) = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix}$

   and compute

   $\det A = 20 - 14 = 6$, \hspace{1cm} \det A_1(\mathbf{b}) = 12 - 7 = 5$, \hspace{1cm} \det A_2(\mathbf{b}) = 5 - 6 = -1$

   $x_1 = \frac{\det A_1(\mathbf{b})}{\det A} = \frac{5}{6}, \hspace{1cm} x_2 = \frac{\det A_2(\mathbf{b})}{\det A} = \frac{-1}{6} = \frac{-1}{6}$

7. The system is equivalent to $A\mathbf{x} = \mathbf{b}$, where $A = \begin{bmatrix} 6s & 4 \\ 9 & 2s \end{bmatrix}$ and $\mathbf{b} = \begin{bmatrix} 5 \\ -2 \end{bmatrix}$. Write

   $A_1(\mathbf{b}) = \begin{bmatrix} 5 \\ -2 \\ 2s \end{bmatrix}$, \hspace{1cm} A_2(\mathbf{b}) = \begin{bmatrix} 6s \\ 9 \\ 5 \end{bmatrix}$

   and compute

   $\det A = 12s^2 - 36 = 12(s^2 - 3) = 12(s - \sqrt{3})(s + \sqrt{3})$

   $\det A_1(\mathbf{b}) = 10s + 8$, \hspace{1cm} \det A_2(\mathbf{b}) = -12s - 45$

   The system has a unique solution when $\det A \neq 0$, that is, when $s \neq \pm \sqrt{3}$. For such a system, the solution is $\mathbf{x} = (x_1, x_2)$, where

   $x_1 = \frac{\det A_1(\mathbf{b})}{\det A} = \frac{10s + 8}{12(s^2 - 3)} = \frac{5s + 4}{6(s^2 - 3)}$

   $x_2 = \frac{\det A_2(\mathbf{b})}{\det A} = \frac{-12s - 45}{12(s^2 - 3)} = \frac{-4s - 15}{4(s^2 - 3)}$

Copyright © 2016 Pearson Education, Inc.
13. First, find the cofactors of \( A = \begin{bmatrix} 3 & 5 & 4 \\ 1 & 0 & 1 \\ 2 & 1 & 1 \end{bmatrix} \).

\[
C_{11} = + \begin{vmatrix} 0 & 1 \\ 2 & 1 \end{vmatrix} = -1, \quad C_{12} = - \begin{vmatrix} 1 & 1 \\ 2 & 1 \end{vmatrix} = 1, \quad C_{13} = + \begin{vmatrix} 1 & 0 \\ 2 & 1 \end{vmatrix} = 1 \\
C_{21} = - \begin{vmatrix} 5 & 4 \\ 0 & 1 \end{vmatrix} = -1, \quad C_{22} = + \begin{vmatrix} 4 & 1 \\ 1 & 1 \end{vmatrix} = -5, \quad C_{23} = - \begin{vmatrix} 5 & 4 \\ 1 & 1 \end{vmatrix} = 7 \\
C_{31} = + \begin{vmatrix} 5 & 4 \\ 0 & 1 \end{vmatrix} = 5, \quad C_{32} = - \begin{vmatrix} 4 & 1 \\ 1 & 1 \end{vmatrix} = 1, \quad C_{33} = + \begin{vmatrix} 5 & 4 \\ 1 & 0 \end{vmatrix} = -5
\]

Then, arrange the transpose of the array of cofactors into the adjugate of \( A \).

\[
\text{adj } A = \begin{bmatrix} -1 & -1 & 5 \\ 1 & -5 & 1 \\ 1 & 7 & -5 \end{bmatrix}
\]

Were you to compute \( \det A \) now, you could write \( A^{-1} \), but you would still need to check whether your calculations are correct. To build in this check, compute

\[
A \cdot \text{adj } A = \begin{bmatrix} 3 & 5 & 4 \\ 1 & 0 & 1 \\ 2 & 1 & 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & 5 \\ 1 & -5 & 1 \\ 1 & 7 & -5 \end{bmatrix} = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 6 \end{bmatrix}
\]

If any off-diagonal entries in the product are nonzero, or if the diagonal entries are not all the same, then some errors have been made, and you can recheck your cofactor calculations. (One possible mistake is to forget the ± signs in front of the 2 × 2 determinants. Another error is to not transpose the array of cofactors.) In this case, the calculations above are correct and \( \det A \) must be 6. So

\[
A^{-1} = \frac{1}{\det A} \text{adj } A = \frac{1}{6} \begin{bmatrix} -1 & -1 & 5 \\ 1 & -5 & 1 \\ 1 & 7 & -5 \end{bmatrix}
\]

19. The parallelogram with vertices \((0, 0), (5, 2), (6, 4), (11, 6)\) is shown below. If no vertex were zero, we would have to translate the parallelogram to the origin by subtracting one vertex from all four vertices. Since one vertex already is zero, use the two vertices adjacent to the origin to construct the columns of \( A \), and compute \( |\det A| \).

\[
A = \begin{bmatrix} 5 & 6 \\ 2 & 4 \end{bmatrix}, \quad \text{(area of the parallelogram)} = |\det A| = |20 - 12| = 8
\]
25. The answer is in the text. I hope you took the advice at the beginning of this Study Guide section and worked the problem (or at least tried hard to work the problem) before checking the answer section. If you were successful, you should be proud of yourself; you are mastering the material—not only determinants but also linear dependence!

31. Let \( \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \), \( \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \), and \( A = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \). Also, let \( S \) be the unit ball in \( \mathbb{R}^3 \), whose bounding surface consists of all vectors \( \mathbf{u} \) such that \( u_1^2 + u_2^2 + u_3^2 = 1 \), and let \( S' \) be the image of \( S \) under the mapping \( \mathbf{u} \mapsto A\mathbf{u} \).

   a. If \( \mathbf{x} \) is in \( S' \), then \( \mathbf{x} = A\mathbf{u} \) for some \( \mathbf{u} \) in \( S \), and \( \mathbf{u} = A^{-1}\mathbf{x} = \begin{bmatrix} x_1 / a \\ x_2 / b \\ x_3 / c \end{bmatrix} \). The condition on \( u_1, u_2, u_3 \) shows that \( \left( \frac{x_1}{a} \right)^2 + \left( \frac{x_2}{b} \right)^2 + \left( \frac{x_3}{c} \right)^2 = 1 \).

   b. Since the volume of the unit ball bounded by \( S \) is \( 4\pi/3 \) and the determinant of \( A \) is \( abc \), Theorem 10 shows that the volume of the region bounded by \( S' \) is \( 4\pi abc/3 \).
Appendix: A Geometric Proof of a Determinant Property

\[ \det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc \]

\[ (a + b)(c + d) = ac + ad + bc + bd \]

\[ \begin{array}{c}
-2 \begin{bmatrix} b \\ c \end{bmatrix} \quad -2 \begin{bmatrix} d \\ b \end{bmatrix} \\
\quad = -2bc - bd \\
-2 \begin{bmatrix} a \\ c \end{bmatrix} \\
(Area \ of \ the \ Parallelogram) \\
\quad = ad - bc
\end{array} \]
Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

**adjugate (or classical adjoint):** The matrix adj $A$ formed from a square matrix $A$ by replacing the $(i, j)$-entry of $A$ by . . . .

**cofactor:** A number $C_{ij} = . . . , called the (i, j)-cofactor of A, where $A_{ij}$ is the submatrix formed by deleting . . . .

**cofactor expansion:** A formula for det $A$ using cofactors associated with one row or one column, such as for row 1: . . . .

**Cramer’s Rule:** A formula for each entry in . . . .

**interchange (matrix):** An elementary matrix obtained by interchanging . . . .

**row replacement (matrix):** An elementary matrix obtained from the identity matrix by . . . .

**scale by r (matrix):** An elementary matrix obtained by multiplying . . . .
4.1 VECTOR SPACES AND SUBSPACES

The main focus of the chapter is on $\mathbb{R}^n$ and its subspaces. However, Section 4.1 builds a framework within which the theory for $\mathbb{R}^n$ rests. Most of the exercises in the chapter concern subspaces of $\mathbb{R}^n$, but some are designed to help you learn gradually about other important vector spaces.

**KEY IDEAS**

A vector space is any collection of objects that behave as vectors do in $\mathbb{R}^n$. (The precise meaning of “behave” is described by the axioms on page 190.) A vector is simply any object that belongs to a vector space. Arrows, polynomials, and infinite sequences of numbers are all examples of vectors, in different vector spaces.

The most important vector spaces in this text are subspaces of $\mathbb{R}^n$. Visualize subspaces as lines or planes through the origin, the origin by itself, or the entire space $\mathbb{R}^n$. In this section, Theorem 1 is a useful tool to show that a set is a vector space. To show that a set is not a subspace, show that one of the properties in the subspace definition is violated. (See Exercises 1–4.)

**STUDY NOTES**

Parts of this chapter are somewhat more theoretical than the earlier chapters, but that is necessary in order to give a solid foundation for the rest of the course. Learn the key definitions and theorems as they appear in the text (rather than waiting until just before an exam). You need this knowledge to get through the conceptual exercises and to be prepared for later sections.

In Example 5, the concept of a function as a single “vector” in a vector space is difficult to absorb on a first reading, and you should not expect to master it in a few days.

The word subspace should usually be accompanied by a phrase such as “of $V$”, or “of $\mathbb{R}^n$”. Without such a phrase, the nature of the elements in the subspace is unknown. For instance, the statement “$H$ is a subspace” does not say whether the elements in $H$ are pairs of numbers, or
polynomials, or something else. But the phrase “$H$ is a subspace of $P_3$” includes the information that each vector in $H$ is a polynomial of degree three or less.

**Set Notation:** The notation introduced in Example 8 is sometimes used in this chapter as an efficient way to describe a set. The symbols and phrases inside the set brackets describe the set. The part to the left of the colon displays the basic nature of the elements in the set (such as vectors in $\mathbb{R}^3$) while the part to the right adds any qualifying conditions that must be met in order for an element to belong to the set (such as all vector entries must be positive). For instance, the set in Example 8 can also be written as

$$H = \left\{ \begin{bmatrix} s \\ t \\ u \end{bmatrix} : s \text{ and } t \text{ are real, and } u = 0 \right\}$$

As another example, the set in Example 11 can be written as

$$W = \left\{ \begin{bmatrix} c \\ d \\ a \\ b \end{bmatrix} : c = a - 3b \text{ and } d = b - a, \text{ where } a \text{ and } b \text{ are arbitrary} \right\}$$

Here, and elsewhere, reference to the scalars $a$ and $b$ as “arbitrary” means that the scalars can be any real numbers.

**Study Tip:** Review Sections 1.3–1.5 and 1.7 before you reach Section 4.3.

---

**SOLUTIONS TO EXERCISES**

1. **a.** $V$ is a subset of $\mathbb{R}^2$. The defining property of $V$ is that the entries of every vector in $V$ are nonnegative. So, if $u$ and $v$ are in $V$, their entries are nonnegative. Since a sum of nonnegative numbers is nonnegative, the vector $u + v$ satisfies the condition that defines $V$. That is, $u + v$ is in $V$.

   **b.** The text’s solution gives a specific $u$ and $c$. One specific “counterexample” suffices to show that $V$ is not a vector space. However, you could also simply say, “If any nonzero vector $v$ in $V$ is multiplied by a negative scalar $c$, then $cv$ is not in $V$ because at least one of its entries is negative.”

7. Most examples in the text involve integers, to make calculations simple, and one can easily overlook the fact that a subspace must be closed under multiplication by *all* real numbers.

**Checkpoint:** Let $H$ be the set of all points $(x, y, z)$ in $\mathbb{R}^3$ that satisfy the condition $x^2 + y^2 - z^2 = 0$. Is $H$ a subspace of $\mathbb{R}^3$?
13. a. \( \mathbf{w} \) is certainly not one of the three vectors in \( \{ \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \} \).
   b. \( \text{Span} \{ \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \} \) contains infinitely many vectors.
   c. \( \mathbf{w} \) is in the subspace (of \( \mathbb{R}^3 \)) spanned by \( \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \) if and only if the equation \( x_1 \mathbf{v}_1 + x_2 \mathbf{v}_2 + x_3 \mathbf{v}_3 = \mathbf{w} \) is consistent (has a solution). Row reduce the augmented matrix:

\[
\begin{bmatrix}
1 & 2 & 4 & 3 \\
0 & 1 & 2 & 1 \\
-1 & 3 & 6 & 2
\end{bmatrix}
\quad \begin{bmatrix}
1 & 2 & 4 & 3 \\
0 & 1 & 2 & 1 \\
0 & 5 & 10 & 5
\end{bmatrix}
\quad \begin{bmatrix}
1 & 2 & 4 & 3 \\
0 & 1 & 2 & 1 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

There is no pivot in the augmented column, so the vector equation is consistent, and \( \mathbf{w} \) is in \( \text{Span} \{ \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \} \).

19. Let \( H \) be the set of all functions described in (5). Then \( H \) is a subset of the vector space \( V \) of all real-valued functions, and \( H \) consists of all linear combinations of the functions \( \cos wt \) and \( \sin wt \). By Theorem 1, \( H \) is a subspace of \( V \), and hence is a vector space.

23. a. See Example 5.
   b. See the definition of a vector.
   c. See Exercises 1, 2, or 3.
   d. See the paragraph before Example 6.
   e. See Example 3.

25. Axiom 4 (plus Axiom 2) shows that \( \mathbf{0} + \mathbf{w} = \mathbf{w} \). Exercises 25–30 show how facts that we take for granted in \( \mathbb{R}^n \) depend only on a few basic properties of \( \mathbb{R}^n \), properties that are now axioms for a general vector space.

31. Let \( H \) be a subspace of \( V \) that contains the vectors \( \mathbf{u} \) and \( \mathbf{v} \). Since \( H \) is closed under multiplication by scalars, \( H \) must contain all scalar multiples of \( \mathbf{u} \) and \( \mathbf{v} \). Since \( H \) is also closed under vector addition, it contains all sums of scalar multiples of \( \mathbf{u} \) and \( \mathbf{v} \). That is, \( H \) contains all vectors in \( \text{Span} \{ \mathbf{u}, \mathbf{v} \} \).

Answer to Checkpoint: \( H \) is not a subspace. Counterexample: \( (1, 0, 1) \) and \( (0, 1, 1) \) are in \( H \) and yet their sum, \( (1, 1, 2) \), is not in \( H \) (because \( 1^2 + 1^2 - 2^2 \neq 0 \)). Many counterexamples are possible. Only one is needed.

MATLAB Graphing Functions

The following commands will graph the function \( f \) in Exercise 37

\[
\begin{align*}
t &= \text{linspace}(0, 2\pi); \\
f &= t.^0 - 8*\cos(t).^2 + 8*\cos(t).^4; \\
\text{grid on} \\
\text{plot}(t, f)
\end{align*}
\]

Here, \( t \) is a vector with 101 entries, the endpoints of 100 equal subintervals of \([0, 2\pi]\); \( \cos(t) \) is a vector whose entries are the cosines of the corresponding entries in \( t \), and \( t.^0 \) is a vector of 1’s. In general, \( t.^k \) is a vector whose entries are the \( k \)th powers of the corresponding entries in \( t \). The command \text{grid on} (or \text{off}) turns on (or off)
gridlines the next time a display is created. Use \texttt{hold on} if you want the next graph to appear on the current display. The command \texttt{hold off} makes each graph appear in a separate display. See the topic “Basic Plotting” in MATLAB’s \texttt{Help} menu for examples of plotting options.

To take advantage of MATLAB’s interactive plotting features see the topic “Plotting Tools in MATLAB’s \texttt{Help} menu. For example, typing in \texttt{plottools} at the command line will summon the plotting tools.

4.2 NULL SPACES, COLUMN SPACES, AND LINEAR TRANSFORMATIONS

Many problems in linear algebra involve a subspace in one way or another. This section provides an opportunity to become comfortable with the concept. The foundation for this section was laid in Sections 1.3 and 1.5. Have you reviewed those sections yet?

\textbf{KEY IDEAS}

Theorems 2 and 3 describe the main types of subspaces. (The proof of Theorem 2 makes a good exam question, because it tests both the definition of \text{Nul} A and the definition of a subspace.)

Theorem 3 actually has two conclusions: \text{Col} A is a subspace of \( \mathbb{R}^m \).

The first phrase tells us that linear combinations of vectors in \text{Col} A remain in \text{Col} A. The phrase “of \( \mathbb{R}^m \)” reminds us that each vector has \( m \) entries (because \( A \) has \( m \) rows). A similar remark applies to the statement from Theorem 2 that \text{Nul} A is a subspace of \( \mathbb{R}^n \).

The box after Example 4 shows that the statement “\text{Col} A = \mathbb{R}^m” can be added to the list of equivalent statements in Theorem 4 of Section 1.4.

\textbf{STUDY NOTES}

In Example 3, the statement “\text{Nul} A = \text{Span}\{u, v, w\}” would be an \textit{explicit} description of \text{Nul} A (provided you specify what \( u, v, \) and \( w \) are).

In some applications, it is important to know that for a given \( m \times n \) matrix \( A \), every equation \( Ax = b \) has a solution (assuming \( b \) is in \( \mathbb{R}^m \)). Yet it may require some effort to determine whether this is the case. If not every equation \( Ax = b \) has a solution, then not every \( b \) belongs to \text{Col} A, and hence \text{Col} A is a proper subspace of \( \mathbb{R}^m \). One of the goals of the next few sections is to obtain a method for determining when \text{Col} A = \mathbb{R}^m.

\textit{Checkpoint 1:} How many pivot positions does an \( m \times n \) matrix \( A \) have if \text{Col} A = \mathbb{R}^m?

\textit{Study Tip:} Theorems 1, 2, and 3 are the main tools for showing that a set is a vector space (that is, a subspace of some known vector space). Review these theorems now, before starting the exercises.

Copyright © 2016 Pearson Education, Inc.
SOLUTIONS TO EXERCISES

1. Now is the time to learn the definition of Nul \( A \). A vector \( x \) is in Nul \( A \) precisely when the product \( Ax \) is defined and \( Ax = 0 \). Given \( x \), simply compute \( Ax \) to determine whether \( Ax \) is zero.

\[
Ax = \begin{bmatrix} 3 & -5 & -3 \\ 6 & -2 & 0 \\ -8 & 4 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ -4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \text{ so } \begin{bmatrix} 3 \\ -4 \end{bmatrix} \text{ is in Nul } A.
\]

**Warning:** In Exercises 3–6, writing an equation \( x = cu + dv \) is not the same as listing, say, the vectors \( u \) and \( v \) that span the null space. The appropriate answer for these exercises is a list of a small finite number of vectors that span Nul \( A \), not a description of *all* the vectors in Nul \( A \).

**Study Tip:** Try Practice Problem 1 before you work on Exercises 7–14. If you can’t find *two* ways to work the practice problem, reread the first paragraph of Section 4.2 (but don’t look at it until you have attempted the practice problem).

7. The set \( W \) is a subset of \( \mathbb{R}^3 \). If \( W \) were a vector space (under the standard operations in \( \mathbb{R}^3 \)), it would be a subspace of \( \mathbb{R}^3 \). But \( W \) fails *every* property of a subspace, so it is not a vector space. For instance, the vector \((0, 0, 0)\) does not satisfy the condition \( a + b + c = 2 \), and so the zero vector is not in \( W \).

13. A typical element of \( W \) can be written as follows:

\[
\begin{bmatrix} c - 6d \\ d \\ c \end{bmatrix} = \begin{bmatrix} c \\ 0 \\ c \end{bmatrix} + \begin{bmatrix} -6d \\ d \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} c + \begin{bmatrix} -6 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} d = \begin{bmatrix} c \\ 1 - 6d \\ 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}
\]

Since \( c \) and \( d \) are any real numbers, this calculation shows that \( W \) is a subspace of \( \mathbb{R}^3 \) (and hence is a vector space) by Theorem 3. Alternatively, this calculation shows that \( W \) is the same as Span\{\( u, v \)\}, so \( W \) is a subspace of \( \mathbb{R}^3 \), by Theorem 1.

**Checkpoint 2:** Why is \( W \) a subspace “of \( \mathbb{R}^3 \)”?

19. The matrix \( A \) is \( 2 \times 5 \), so vectors in Nul \( A \) must have 5 entries and vectors in Col \( A \) must have 2 entries. Thus Nul \( A \) is a subspace of \( \mathbb{R}^5 \) and Col \( A \) is a subspace of \( \mathbb{R}^2 \).

**Study Tip:** Exercises 17–20 may seem simple, but they will help you in Section 4.5.

25. **a.** Check the definition before Example 1.

**b.** See Theorem 2.

**c.** See the remark just before Example 4.
d. See the table that contrasts Nul $A$ and Col $A$.

e. See Fig. 2.

f. See the remark after Theorem 3.

31. The solution in the text shows that $T$ is a linear transformation. If $T(p)$ is the zero vector, then $p(0) = 0$ and $p(1) = 0$, by definition of $T$. One such polynomial is $p(t) = t(t - 1)$. Any other quadratic polynomial that vanishes at 0 and 1 must be a multiple of $p$, so $p$ spans the kernel of $T$.

For the range of $T$, observe that the image of the constant 1 function is $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$, and the image of the polynomial $t$ is $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$. Denote these two images by $u$ and $v$, respectively. Since the range of $T$ is a subspace of $\mathbb{R}^2$ that contains $u$ and $v$, the range must contain all linear combinations of $u$ and $v$. By inspection, $u$ and $v$ are linearly independent, so they span $\mathbb{R}^2$. Thus the range of $T$ must contain all of $\mathbb{R}^2$.

37. The vector $w$ is in Col $A$ because row reduction of the augmented matrix $[A \ w]$ shows that the equation $Ax = w$ is consistent. The vector $w$ is not in Nul $A$ because $Ax$ is not the zero vector.

Answers to Checkpoints:

1. An $m \times n$ matrix $A$ must have $m$ pivot positions in order for Col $A$ to be all of $\mathbb{R}^m$.

2. $W$ is a subspace “of $\mathbb{R}^3$” because each vector in $W$ has three entries.

Mastering Linear Algebra Concepts: Vector Space, Subspace, and Column and Null Spaces

A vector space is a nonempty set of objects on which are defined two operations, called addition and multiplication by scalars, that satisfy ten axioms. Make one review sheet for vector space and subspace. Include as much of the definition of a vector space as you are required to know. (Ask your instructor.) Include the full definition of a subspace.

Organize what you have learned in Sections 4.1 and 4.2 (together with Sections 1.3–1.5), using the categories listed below. Your examples of vector spaces will be the same as your examples of subspaces. (See the paragraph just before Example 6 in Section 4.1.) The column space and null space of a matrix should be among your examples of subspaces. (The references cited below contain information that will help you to construct your review sheets.)

- definitions of vector space, subspace Pages 190 and 193
- equivalent descriptions Paragraph before Example 6 in Sec. 4.1
- geometric interpretations Figs. 6, 7, 9, in Sec. 4.1
- subspaces described explicitly Theorems 1, 3; Exercises 15–18 in Sec. 4.1
4.3 LINEARLY INDEPENDENT SETS; BASES

The definition of linear independence carries over from $\mathbb{R}^n$ to any vector space. The geometric interpretations in Chapter 1 of linearly independent and dependent sets should help you visualize these concepts here.

**KEY IDEAS**

In general, you cannot use an ordinary matrix equation $Ax = 0$ to study linear dependence of $\{v_1, \ldots, v_p\}$. You have to work with the vector equation $c_1v_1 + \cdots + c_pv_p = 0$, or with Theorem 4, unless the vectors happen to be $n$-tuples of numbers.

A set $\{v\}$ with $v \neq 0$ is linearly independent because the vector equation $x_1v = 0$ has only the trivial solution. (See Exercise 30 in Section 4.1.) The set $\{0\}$ is linearly dependent, because the equation $x_10 = 0$ has many nontrivial solutions.

Theorem 6 is important for later work, but its proof is rather subtle. Study Examples 8 and 9 carefully, as well as the proof of the theorem.

A basis for a vector space $V$ is a set in $V$ that is large enough to span $V$ and small enough to be linearly independent. See also the subsection *Two Views of a Basis*. The plural of *basis* is *bases*.

**Warning:** If a set in $V$ does not span $V$, the set may or may not be linearly dependent.

**SOLUTIONS TO EXERCISES**

1. The complete solution is in the text. For a general set of $n$ vectors in $\mathbb{R}^n$, row operations on a matrix will usually be needed to determine if the matrix has $n$ pivot positions.

7. Again, the solution is in the text. Any set in $\mathbb{R}^n$ with fewer than $n$ vectors cannot span $\mathbb{R}^n$ and therefore cannot be a basis for $\mathbb{R}^n$. Such a set may or may not be linearly independent. What
similar statement can you make about a set in \( \mathbb{R}^n \) with more than \( n \) vectors? See Exercise 8 for ideas.

**Study Tip:** Theorem 4 in Section 1.4 may help you decide whether a set of vectors spans \( \mathbb{R}^n \).

13. The matrix \( B \) is in echelon form and displays the pivot columns. A basis for \( \text{Col } A \) consists of columns 1 and 2 of \( A \):

\[
\begin{bmatrix}
-2 & 4 \\
2 & -6 \\
-3 & 8
\end{bmatrix}
\]

This is not the only correct choice, but it is the “standard” choice. A *wrong* choice would be columns 1 and 2 of \( B \). See the Warning after Theorem 6.

For the null space, solve \( Ax = 0 \):

\[
\begin{bmatrix}
1 & 0 & 6 & 5 & 0 \\
0 & 2 & 5 & 3 & 0 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix} \sim \begin{bmatrix}
1 & 0 & 6 & 5 & 0 \\
0 & 1 & 5/2 & 3/2 & 0 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

Then \( x_1 = -6x_3 - 5x_4 \), \( x_2 = -(5/2)x_3 - (3/2)x_4 \), with \( x_3 \) and \( x_4 \) free. The general solution is

\[
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4
\end{bmatrix} = \begin{bmatrix}
-6x_3 \\
-(5/2)x_3 \\
x_3 \\
x_4
\end{bmatrix} = x_3 \begin{bmatrix}
-6 \\
-5/2 \\
1 \\
0
\end{bmatrix} + x_4 \begin{bmatrix}
-5 \\
-3/2 \\
0 \\
1
\end{bmatrix}
\]

This equation describes all vectors in \( \text{Nul } A \), not just a basis for \( \text{Nul } A \). For a basis, the “standard” choice is \((-6, -5/2, 1, 0)\) and \((-5, -3/2, 0, 1)\). Another choice is \((-12, -5, 2, 0)\) and \((-10, -3, 0, 2)\), which avoids fractions.

**Warning:** You really need to know the definition of \( \text{Nul } A \) and the definition of \( \text{Col } A \), not just the procedures for finding bases for these spaces. The definitions will help you avoid the fairly common mistake of attempting to use the null space procedure to find a basis for a column space, or vice-versa.

19. You can solve the equation \( 4v_1 + 5v_2 - 3v_3 = 0 \) for any one of the three vectors in terms of the others. By the Spanning Set Theorem, the set spanned by all three vectors is the same as the set spanned by any two of the vectors—any one of the three vectors can be discarded. If you discard \( v_3 \), then \( v_1 \) and \( v_2 \) span \( H \) and are obviously linearly independent. Hence \{\( v_1, v_2 \)\} is a basis for \( H \). The same reasoning applies to \{\( v_1, v_3 \)\} and \{\( v_2, v_3 \)\}. These are the most likely answers. But, once you realize that \{\( v_1, v_2 \)\} is a basis for \( H \), you can make others, such as \{\( v_1, v_2 + rv_1 \)\} for any scalar \( r \). Showing that this set is a basis for \( H \) does take some work, however. You might try to do this. In some courses that spend time on theoretical questions, this problem could appear on a test.
21. a. See the paragraph preceding Theorem 4.
   b. See the definition of a basis.
   c. See Example 3.
   d. See the subsection Two Views of a Basis.
   e. See the box before Example 9.

23. Let \( A = [v_1 \ v_2 \ v_3 \ v_4] \). Since \( A \) is square and its columns span \( \mathbb{R}^4 \), the columns of \( A \) must be linearly independent, by the Invertible Matrix Theorem. So \( \{v_1, v_2, v_3, v_4\} \) is a basis for \( \mathbb{R}^4 \).

Checkpoint: Suppose \( \{v_1, \ldots, v_n\} \) is a basis for \( \mathbb{R}^n \) and \( A \) is an invertible \( n \times n \) matrix. Explain why \( \{Av_1, \ldots, Av_n\} \) is a basis for \( \mathbb{R}^n \).

25. The displayed equation shows only that \( \text{Span} \{v_1, v_2, v_3\} \) contains \( H \). In fact, the vectors \( v_1, v_2, v_3 \) are not all in \( H \), so \( \text{Span} \{v_1, v_2, v_3\} \) cannot be \( H \). Therefore \( \{v_1, v_2, v_3\} \) cannot be a basis for \( H \). (It is easy to check that \( \{v_1, v_2, v_3\} \) is a basis for \( \mathbb{R}^3 \).)

31. (This generalizes Exercise 31 in Section 1.8.) Suppose \( \{v_1, \ldots, v_p\} \) is linearly dependent. Then there exist \( c_1, \ldots, c_p, \) not all zero, such that

\[
c_1 v_1 + \cdots + c_p v_p = 0
\]

Then, since \( T \) is linear,

\[
T(c_1 v_1 + \cdots + c_p v_p) = T(0) = 0
\]

See the boxed statement on page 66.

and

\[
c_1 T(v_1) + \cdots + c_p T(v_p) = 0
\]

Since not all the \( c_i \) are zero, \( \{T(v_1), \ldots, T(v_p)\} \) is linearly dependent.

Study Tip: The solution of Exercise 31 illustrates how to use linear dependence: If \( \{v_1, v_2, \ldots, v_p\} \) is known to be linearly dependent, then you can write \( c_1 v_1 + \cdots + c_p v_p = 0 \), assume that not all the \( c_k \) are zero, and use this equation in some way.

Answer to Checkpoint: Let \( B = [v_1 \cdots v_n] \). Then \( B \) is invertible because \( \{v_1, v_2, \ldots, v_n\} \) is a basis for \( \mathbb{R}^n \). Since \( A \) is also an invertible \( n \times n \) matrix, so is \( AB \); hence the columns of \( AB \), namely \( Av_1, \ldots, Av_n \), form a basis for \( \mathbb{R}^n \).
Mastering Linear Algebra Concepts: Basis

To the review sheet(s) you have on linear independence, add Examples 1, 2, and 6 from this section. The definition and geometric interpretations are unchanged. Add a note about not using the matrix equation $Ax = 0$ in the general case.

Start a separate review sheet for “basis”, even though it involves two other concepts (span and linear independence) already being reviewed.

- definition Page 209
- geometric interpretation Fig. 1
- special cases Standard bases for $\mathbb{R}^n$ and $\mathbb{P}_n$
- examples and counterexamples Example 10, Exercise 25
- algorithms and computations Examples 7 and 9; Example 3 in Sec. 4.2
- connections with other concepts Invertible Matrix Theorem Unique Representation Theorem (in Sec. 4.4)

MATLAB ref and cos

The command `ref(A)` produces the reduced echelon form of $A$. From that you can write a basis for Col $A$ or write the homogeneous equations that describe Nul $A$. (Don’t forget that $A$ is a coefficient matrix, not an augmented matrix.) MATLAB has another command, `rref`, that works basically the same as `ref` but often is much slower because it checks for rational entries in the matrix.

In some cases, roundoff error or an extremely small pivot entry can cause `ref` to produce an incorrect echelon form. The more reliable singular value decomposition (see Section 7.4) can produce bases for Col $A$ and Nul $A$, but `ref` is satisfactory for our purposes.

For Exercise 38, see the MATLAB box for Section 4.1.

4.4 COORDINATE SYSTEMS

This section contains a variety of geometric and algebraic explanations of the idea of a coordinate system for a vector space.

Copyright © 2016 Pearson Education, Inc.
KEY IDEAS

The coordinate mapping from a vector space $V$ (with a basis of $n$ elements) onto $\mathbb{R}^n$ is a rule for giving “$\mathbb{R}^n$-names” to vectors in $V$ in such a way that the vector space structure of $V$ is still visible in $\mathbb{R}^n$. Every vector space calculation in $V$ is precisely mirrored by the same calculation in $\mathbb{R}^n$.

An important special case is when $V$ is itself $\mathbb{R}^n$, and each vector $x$ and its coordinate vector $[x]_B$ are related by a matrix equation $x = P_B[x]_B$.

Everything in the section depends on the Unique Representation Theorem. The proof of that theorem could appear on an exam because it shows precisely why the two properties of a basis $B$ are important, and it illustrates how linear independence can be used in an argument:

Any vector in $V$ has “coordinates” because $B$ spans $V$, and the coordinates are uniquely determined because $B$ is linearly independent.

If you are asked to prove the theorem, make sure your proof shows exactly where each property of a basis is needed in the proof. Also, be careful not to use a matrix in the proof. The vectors $v_1, \ldots, v_n$ cannot be arranged as the columns of an ordinary matrix when the vectors are in some abstract vector space.

Checkpoint: Let $B = \{b_1, \ldots, b_n\}$ be a basis for $\mathbb{R}^n$. Apply the Invertible Matrix Theorem to the matrix $A = [b_1 \ldots b_n]$ and deduce the Unique Representation Theorem for the case when $V = \mathbb{R}^n$.

STUDY NOTES

Be careful to distinguish between $x$ and $[x]_B$. They are not equal in general, even if $x$ itself is in $\mathbb{R}^n$ (unless $B$ is the standard basis for $\mathbb{R}^n$).

Theorem 8 and Exercises 25 and 26 show that the coordinate mapping translates vector space statements or calculations in $V$ into equivalent (and familiar) calculations in $\mathbb{R}^n$. The table below lists some examples of typical linear algebra statements.

**CORRESPONDING STATEMENTS IN ISOMORPHIC VECTOR SPACES**

<table>
<thead>
<tr>
<th>Linear Algebra in $V$</th>
<th>Matrix Algebra in $\mathbb{R}^n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. $u, v$, and $w$ are in $V$</td>
<td>$[u]_B, [v]_B, \text{ and } [w]_B$ are in $\mathbb{R}^n$</td>
</tr>
<tr>
<td>b. $w$ is in Span{$u, v$}, or $w$ is in the subspace of $V$ spanned by $u$ and $v$</td>
<td>$[w]_B$ is in Span{$[u]_B, [v]_B$}, or $[w]_B$ is in the subspace of $\mathbb{R}^n$ spanned by $[u]_B$ and $[v]_B$</td>
</tr>
<tr>
<td>c. $w = cu + dv$</td>
<td>$[w]_B = c[u]_B + d[v]_B$</td>
</tr>
<tr>
<td>d. ${v_1, \ldots, v_p}$ is lin. indep.</td>
<td>${[v_1]_B, \ldots, [v_p]_B}$ is lin. indep.</td>
</tr>
<tr>
<td>e. ${v_1, \ldots, v_p}$ spans $V$</td>
<td>${[v_1]_B, \ldots, [v_p]_B}$ spans $\mathbb{R}^n$</td>
</tr>
<tr>
<td>f. ${v_1, \ldots, v_n}$ is a basis for $V$</td>
<td>${[v_1]_B, \ldots, [v_n]_B}$ is a basis for $\mathbb{R}^n$</td>
</tr>
</tbody>
</table>
1. Since \([\mathbf{x}]_B = \begin{bmatrix} 5 \\ 3 \end{bmatrix}\), we have \(\mathbf{x} = 5\mathbf{b}_1 + 3\mathbf{b}_2 = 5\begin{bmatrix} 3 \\ -5 \end{bmatrix} + 3\begin{bmatrix} -4 \\ 6 \end{bmatrix} = \begin{bmatrix} 3 \\ -7 \end{bmatrix}\).

7. The \(B\)-coordinates of \(\mathbf{x}\) are scalars \(c_1, c_2, c_3\) that satisfy \(c_1\mathbf{b}_1 + c_2\mathbf{b}_2 + c_3\mathbf{b}_3 = \mathbf{x}\). To solve this vector equation, row reduce the augmented matrix:

\[
\begin{bmatrix}
1 & -3 & 2 & 8 \\
-1 & 4 & -2 & -9
\end{bmatrix}
\sim
\begin{bmatrix}
1 & -3 & 2 & 8 \\
0 & 1 & 0 & -1
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 0 & -1 \\
0 & 0 & 1 & 3
\end{bmatrix}
\]

So \([\mathbf{x}]_B = \begin{bmatrix} -1 \\ 3 \end{bmatrix}\).

13. The \(B\)-coordinates of \(\mathbf{p}\) are scalars \(c_1, c_2, c_3\) that satisfy

\[
c_1(1 + t^2) + c_2(t + t^2) + c_3(1 + 2t + t^2) = \mathbf{p}(t) = 1 + 4t + 7t^2
\]

Multiply out terms on the left:

\[
c_1 + c_1t^2 + c_2t + c_2t^2 + c_3 + 2c_3t + c_3t^2 = 1 + 4t + 7t^2
\]

On the left, group the constant terms, the terms involving \(t\), and the terms involving \(t^2\):

\[
(c_1 + c_3) + (c_2 + 2c_3)t + (c_1 + c_2 + c_3)t^2 = 1 + 4t + 7t^2
\]

Equate coefficients of like powers of \(t\) to obtain the system of equations:

\[
\begin{align*}
    c_1 + c_3 &= 1 \\
    c_2 + 2c_3 &= 4 \\
    c_1 + c_2 + c_3 &= 7
\end{align*}
\]

Row reduce the augmented matrix to obtain

\[
\begin{bmatrix}
1 & 0 & 1 & 1 \\
0 & 1 & 2 & 4 \\
1 & 1 & 1 & 7
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 0 & 2 \\
0 & 1 & 0 & 6 \\
0 & 0 & -2 & 2
\end{bmatrix}
\
\text{Thus, } [\mathbf{p}]_B = \begin{bmatrix} 2 \\ 6 \\ -1 \end{bmatrix} \quad (2)
\]

Perhaps you can skip writing the second and third displayed equations and mentally go from (1) directly to the systems of equations. That will save writing time, but mistakes can occur.

A shorter solution uses Theorem 8 and the fact that a calculation in \(P_2\) can be done instead with coordinate vectors relative to the standard basis \(\{1, t, t^2\}\). Using this idea, you go directly from equation (1) to the equivalent equation using coordinate vectors:
This vector equation is, of course, equivalent to the system of equations above, and you solve it by row reducing the augmented matrix as in (2).

**Warning**: The second solution for Exercise 13 is faster, but students can easily forget what their calculations mean. I expect my students to write about what they are doing, to show that they understand what they are calculating. For an acceptable solution to Exercise 13, write that the $B$–coordinates of $p$ are scalars $c_1$, $c_2$, $c_3$ that satisfy equation (1), write something about using coordinate vectors to express (1) in the equivalent form (3), and then solve (3) by the calculations in (2).

15. a. See the definition of a $B$-coordinate vector.
   b. See equation (4).
   c. See Example 5.

19. The set $S$ spans $V$ because every $x$ in $V$ has a representation as a (unique) linear combination of elements of $S$. To show linear independence, suppose that $S = \{v_1, \ldots, v_n\}$ and $c_1v_1 + \cdots + c_nv_n = 0$ for some scalars $c_1, \ldots, c_n$. The case when $c_1 = \cdots = c_n = 0$ is one possibility. By hypothesis, this is the only possible representation of the zero vector as a linear combination of the elements of $S$. So $S$ is linearly independent and hence is a basis for $V$.

23. Suppose that $[u]_B = [w]_B$ for some $u$ and $w$ in $V$, and denote the entries in this coordinate vector by $c_1, \ldots, c_n$. By definition of the coordinate vectors, $u = c_1b_1 + \cdots + c_nb_n$ and $w = c_1b_1 + \cdots + c_nb_n$

which shows that $u = w$. Since $u$ and $w$ were arbitrary elements of $V$, this shows that the coordinate mapping is one-to-one.

25. Since the coordinate mapping is one-to-one, the following equations have the same solutions, $c_1, \ldots, c_p$:

$$c_1u_1 + \cdots + c_pu_p = 0 \quad \text{(the zero vector in } V)$$

$$[c_1u_1 + \cdots + c_pu_p]_B = [0]_B \quad \text{(the zero vector in } \mathbb{R}^n)$$

Since the coordinate mapping is linear, (2) is equivalent to

$$c_1[u_1]_B + \cdots + c_p[u_p]_B = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

Hence $c_1, \ldots, c_p$ satisfy (1) if and only if they satisfy (3). So (1) has only the trivial solution if and only if (3) has only the trivial solution. It follows that $\{u_1, \ldots, u_p\}$ is linearly independent if and only if $\{[u_1]_B, \ldots, [u_p]_B\}$ is linearly independent. (This fact is also an immediate consequence of Exercises 31 and 32 in Section 4.3.)
Study Tip: Exercises 27–34 tell you to explain your work and justify your conclusions. This requirement is to help you think about your calculations. Check with your instructor. A similar requirement is likely to appear on exam questions.

Warning: The standard mistake in Exercises 27–34 is to write the coordinate vectors as the rows of a matrix and then to check the linear independence of the columns. This is completely wrong! Since we mainly work with column vectors, it is wise to write the coordinate vectors first (as columns) and afterwards write a matrix that can be row reduced to check for linear independence. See the second solution of Exercise 13 and the sample solution for Exercise 27, below.

27. The coordinate vectors of the polynomials are

\[
\begin{bmatrix}
1 \\
0 \\
0 \\
1
\end{bmatrix}, \begin{bmatrix}
3 \\
1 \\
-2 \\
0
\end{bmatrix}, \begin{bmatrix}
0 \\
-1 \\
3 \\
-1
\end{bmatrix}
\] (relative to the standard basis).

To check linear independence of these vectors in \(\mathbb{R}^4\), compute

\[
\begin{bmatrix}
1 & 3 & 0 & 0 \\
0 & 1 & -1 & 0 \\
0 & -2 & 3 & 0 \\
1 & 0 & -1 & 0
\end{bmatrix} \sim \begin{bmatrix}
1 & 0 & -1 & 0 \\
0 & 1 & -1 & 0 \\
0 & -2 & 3 & 0 \\
2 & 3 & -1 & 0
\end{bmatrix} \sim \cdots \sim \begin{bmatrix}
1 & 0 & -1 & 0 \\
0 & 1 & -1 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

The three coordinate vectors are linearly independent in \(\mathbb{R}^4\), because the equation \(x_1v_1 + x_2v_2 + x_3v_3 = 0\) has only the trivial solution. By the isomorphism with \(P_3\), the three polynomials are linearly independent in \(P_3\).

Note: When you write that an equation has only the trivial solution, you must indicate in some way what equation you have in mind.

Study Tip: Exercises 27–30 could be expanded to ask whether the given polynomials form a basis for \(P_3\). You might see such a question on an exam.

31. In each part, place the coordinate vectors of the polynomials into the columns of a matrix and reduce the matrix to echelon form:
4.4 • Coordinate Systems  4-15

The four coordinate vectors do not span \( \mathbb{R}^3 \) because there is no pivot in row 3. Because of the isomorphism between \( \mathbb{R}^3 \) and \( \mathbb{P}_2 \), the corresponding polynomials do not span \( \mathbb{P}_2 \).

The four coordinate vectors span \( \mathbb{R}^3 \) because there is a pivot in each row. Because of the isomorphism between \( \mathbb{R}^3 \) and \( \mathbb{P}_2 \), the corresponding polynomials span \( \mathbb{P}_2 \).

**Study Tip:** Carefully study the solutions in Exercise 31, because some student papers have a discussion that is far removed from a correct answer. After creating the matrix from the coordinate vectors and row reducing, a student might write something such as “They do not span because there is not a pivot in each row.”

Error #1: “They” is a pronoun with no mention of what “they” is. Does the student mean the polynomials, the coordinate vectors, or the columns of the matrix?

Error #2: Span what? The verb “span” requires an object, such as \( \mathbb{P}_2 \) or \( \mathbb{R}^3 \).

Error #3. There is no mention of “isomorphism.” The phrase “not a pivot in each row” has relevance only to the fact that the coordinate vectors do not span \( \mathbb{R}^3 \). The only way to get from this fact to the polynomials in the exercise is to use the isomorphism between \( \mathbb{P}_2 \) and \( \mathbb{R}^3 \).

(Some instructors may permit the term “correspondence” instead of the more precise “isomorphism.”)

**Answer to Checkpoint:** The columns of the matrix \( A = [b_1 \ldots b_n] \) form a basis for \( \mathbb{R}^n \), so \( A \) is invertible, by the Invertible Matrix Theorem. By Theorem 5 in Section 2.2, for each \( x \) in \( \mathbb{R}^n \) there exists a unique vector \( c = (c_1, \ldots, c_n) \) such that \( x = Ac \), that is, \( x = c_1 b_1 + \cdots + c_n b_n \).

**MATLAB The Backslash Operator \**

If an equation \( Ax = b \) has a unique solution, MATLAB will automatically produce \( x \) if you use the command

\[
\textbf{x} = \textbf{A}\backslash\textbf{b}
\]

In this section, the equation probably will have the form \( Pu = x \), with \( u \) the \( B \)-coordinate vector of \( x \), and the command will be \( \textbf{u} = \textbf{P}\backslash\textbf{x} \).
The “backslash” command works in two different ways. When $A$ is square, the command $A\backslash b$ causes MATLAB to create an LU factorization of $A$ (see Section 2.5); if $A$ is invertible, the factorization is used to produce the unique solution to $Ax = b$; and if $A$ is not invertible, MATLAB gives the error message “matrix is singular” (even if the system $Ax = b$ has a solution). When $A$ is not square, $A\backslash b$ creates a least-squares solution (see Section 6.5).

4.5 THE DIMENSION OF A VECTOR SPACE

This short section provides a convenient way to compare the “sizes” of various subspaces of a vector space. The notion of dimension will be used frequently throughout the rest of the text.

KEY IDEAS

Theorem 10 shows that the dimension of a finite-dimensional vector space does not depend on the particular basis for the space. Example 4 shows how to visualize subspaces of various dimensions. Theorems 9 and 12 are important for later theory and applications. You might remember Theorem 9 more easily in this form:

In an $n$-dimensional vector space, any set of more than $n$ vectors must be linearly dependent.

Theorem 12, the Basis Theorem, may be restated as follows:

If $V$ is a $p$-dimensional vector space, with $p \geq 1$, and if $S$ is a subset of $V$ that contains exactly $p$ elements, then $S$ is linearly independent if and only if $S$ spans $V$.

Warning: Theorem 11 shows that any basis of a subspace $H$ of a finite-dimensional space $V$ can be extended to a basis of $V$. But it is not true that any basis of $V$ can be cut down to a basis for $H$. That is, if $S$ is a basis for $V$, it is not likely that a subset of $S$ is a basis for $H$. For instance, suppose $S$ is the standard basis for $\mathbb{R}^3$ and $H$ is a plane in $\mathbb{R}^3$ that contains the origin but none of the coordinate axes. Then no subset of $S$ can be a basis for $H$.

SOLUTIONS TO EXERCISES

1. Since $\begin{bmatrix} s - 2t \\ s + t \\ 3t \end{bmatrix} = s \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} + t \begin{bmatrix} 1 \\ 1 \\ 3 \end{bmatrix}$ for all $s$, $t$, the set $\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$ certainly spans the subspace, call it $H$. Also, the set is obviously linearly independent (because the vectors are not multiples), so the set is a basis for $H$. Hence, $\dim H = 2$.

3. The given subspace, call it $H$, is the set of all linear combinations of the vectors
First determine if \( \{v_1, v_2, v_3\} \) is linearly independent. One way to do this is to row reduce the augmented matrix \([v_1 \ v_2 \ v_3 \ 0]\). A faster way is to use Theorem 4 in Section 4.3. Clearly, \( v_1 \neq 0 \), \( v_2 \) is not a multiple of \( v_1 \), and \( v_3 \) is not a linear combination of the vectors \( v_1 \), \( v_2 \) that precede it, because the first entry in \( v_3 \) is not zero. Hence \( \{v_1, v_2, v_3\} \) is linearly independent and thus is a basis for the space \( H \) it spans. Thus \( \dim H = 3 \).

7. Standard calculations show that the set of solutions of the homogeneous system consists of only the trivial solution. So the subspace is \( \{0\} \), and it has no basis. (The vector \( 0 \) spans the space, but \( \{0\} \) is a linearly dependent set.) By definition, the dimension is zero. [Note: Instructors who want every subspace to have a basis often define the empty set to be a basis for \( \{0\} \). The number of vectors in this basis is zero, so the dimension of \( \{0\} \) is still zero.]

13. \( A \) has three pivot columns, so \( \dim \text{Col } A = 3 \). There are two columns without pivot positions, so the equation \( Ax = 0 \) has two free variables, and \( \dim \text{Nul } A = 2 \).

19. a. See the box before Example 5.  
b. Read Example 4 carefully.  
c. See Example 1.  
d. See Theorem 10.  
e. See Practice Problem 2. (You should be working the practice problems before you start the exercises.)

21. Form the matrix whose columns are the coordinate vectors of the Hermite polynomials, relative to the standard basis \( \{1, t, t^2, t^3\} \):

\[
A = \begin{bmatrix}
1 & 0 & -2 & 0 \\
0 & 2 & 0 & -12 \\
0 & 0 & 4 & 0 \\
0 & 0 & 0 & 8
\end{bmatrix}
\]

The matrix has four pivots and hence is invertible. So its columns, the coordinate vectors, are linearly independent. Hence the Hermite polynomials themselves are linearly independent in \( P_3 \). Since there are four Hermite polynomials, and \( \dim P_3 = 4 \), we conclude from The Basis Theorem that the Hermite polynomials form a basis for \( P_3 \).

Note: You could, of course, say that the columns of the matrix \( A \) span \( \mathbb{R}^4 \). But you cannot stop with that assertion, because you need the polynomials to span \( \mathbb{P}_3 \). You have to go on and point out that because of the isomorphism between \( \mathbb{P}_3 \) and \( \mathbb{R}^4 \), a set of vectors spans \( \mathbb{P}_3 \) if and only if the set of coordinate vectors (the columns of \( A \)) spans \( \mathbb{R}^4 \). So the solution is shorter if you appeal to the Basis Theorem.

25. Note that \( n \geq 1 \), because \( S \) cannot have fewer than 0 vectors. If \( \dim V = n \geq 1 \), then \( V \neq \{0\} \). If \( S \) spans \( V \), then a subset \( S' \) of \( S \) is a basis for \( V \), by the Spanning Set Theorem.
But if $S$ has fewer than $n$ vectors, then $S'$ also has fewer than $n$ vectors. This is impossible, by Theorem 10, because $\dim V = n$. So $S$ cannot span $V$.

27. If $\mathbb{P}$ were finite-dimensional, then Theorem 11 would imply that $n + 1 = \dim \mathbb{P}_n \leq \dim \mathbb{P}$ for each $n$, because each $\mathbb{P}_n$ is a subspace of $\mathbb{P}$. This is impossible, so $\mathbb{P}$ must be infinite-dimensional.

29. a. True. Apply the Spanning Set Theorem to the set $\{v_1, \ldots, v_p\}$ and produce a basis for $V$. This basis will have no more than $p$ elements in it, so $\dim V$ must be no more than $p$.

b. True. By Theorem 11, $\{v_1, \ldots, v_p\}$ can be expanded to a basis for $V$. The basis will have at least $p$ elements in it, so $\dim V$ must be at least $p$.

c. True. Take any basis (of $p$ vectors) for $V$ and adjoin the zero vector. Spanning sets can be arbitrarily large. The dimension of $V$ being $p$ only keeps spanning sets from having fewer than $p$ elements.

31. Since $H$ is a nonzero subspace of a finite-dimensional space, $H$ is finite-dimensional and has a basis, say, $v_1, \ldots, v_p$. Any vector in $T(H)$ has the form $T(y)$ for some $y$ in $H$. Since $\{v_1, \ldots, v_p\}$ spans $H$, there exist scalars $c_1, \ldots, c_p$ such that $y = c_1v_1 + \cdots + c_pv_p$. Since $T$ is linear, $T(y) = c_1T(v_1) + \cdots + c_pT(v_p)$. This shows that $\{T(v_1), \ldots, T(v_p)\}$ spans $T(H)$. By Exercise 29(a), $\dim T(H) \leq p = \dim H$.

Second proof: Let $k = \dim T(H)$. If $k = 0$, then $k < \dim H$. Otherwise, $T(H)$ has a basis, which can be written in the form $T(v_1), \ldots, T(v_k)$ for some vectors $v_1, \ldots, v_k$ in $H$. Since $\{T(v_1), \ldots, T(v_k)\}$ is linearly independent, so is $\{v_1, \ldots, v_k\}$, by Exercise 31 in Section 4.3. Since $v_1, \ldots, v_k$ are in $H$, the dimension of $H$ must be at least $k$.

Hint for Exercise 32: Use an exercise in Section 4.3.

Study Tip: The next section is quite important. Do your best to get caught up now. Otherwise, you may have difficulty relating the various concepts and facts about matrices that will be reviewed in Section 4.6.

4.6 RANK

This section gives you a chance to put together most of the ideas of the chapter in the same way that Section 2.3 collected the main ideas of the sections that preceded it.

KEY IDEAS

The Rank Theorem is the main result. By definition, $\text{rank } A = \dim \text{Col } A$. But because rank $A$ is also the dimension of $\text{Row } A$, the displayed equation in the theorem leads to the equation: $\dim \text{Row } A + \dim \text{Nul } A = n$. **Copyright © 2016 Pearson Education, Inc.**
Equivalent Descriptions of Rank

The rank of an \( m \times n \) matrix \( A \) may be described in several ways:

- the dimension of the column space of \( A \), (our definition)
- the number of pivot positions in \( A \), (from Theorem 6)
- the maximum number of linearly independent columns in \( A \),
- the dimension of the row space of \( A \), (from the Rank Theorem)
- the maximum number of linearly independent rows in \( A \),
- the number of nonzero rows in an echelon form of \( A \),
- the maximum number of columns in an invertible submatrix of \( A \).

(Supplementary Exercise 17 at the end of the chapter)

Pay attention to how Theorem 13 differs from the results in Section 4.3 about \( \text{Col } A \): If you are interested in rows of \( A \), use the nonzero rows of an echelon form \( B \) as a basis for \( \text{Row } A \); if you are interested in the columns of \( A \), only use \( B \) to obtain information about \( A \) (namely, to identify the pivot columns), and use the pivot columns of \( A \) as a basis for \( \text{Col } A \). For \( \text{Nul } A \), it is important to use the reduced echelon form of \( A \).

When a matrix \( A \) is changed into a matrix \( B \) by one or more elementary row operations, the row space, null space, and column space of \( A \) may or may not be the same as the corresponding subspaces for \( B \). The following table summarizes what can happen in this situation. Effects of Elementary Row Operations

- Row operations do not affect the linear dependence relations among the columns. (That is, the columns of \( A \) have exactly the same linear dependence relations as the columns of any matrix that is row-equivalent to \( A \).)
- Row operations usually change the column space.
- Row operations never change the row space.
- Row operations never change the null space.

The four subspaces shown in Figure 1 in the text are called the fundamental subspaces determined by \( A \). (See Exercises 27–29.) The main difficulty here is to avoid confusion between \( \text{Row } A \), \( \text{Nul } A \), and \( \text{Col } A \). The fourth subspace will appear again in Sections 6.1 and 7.4.

The following table lists all statements that are in the Invertible Matrix Theorem at this point in the course, arranged in the scheme used in Section 2.3 of this Study Guide. The statements in all three columns are equivalent when \( A \) is square \((m = n = p)\). As before, a few extra statements have been added to make the table more symmetrical.
## STATEMENTS FROM THE INVERTIBLE MATRIX THEOREM

<table>
<thead>
<tr>
<th>Equivalent statements for an ( m \times n ) matrix ( A ).</th>
<th>Equivalent statements for an ( n \times n ) square matrix ( A ).</th>
<th>Equivalent statements for any ( n \times p ) matrix ( A ).</th>
</tr>
</thead>
<tbody>
<tr>
<td>k. There is a matrix ( D ) such that ( AD = I ).</td>
<td>a. ( A ) is an invertible matrix.</td>
<td>j. There is a matrix ( C ) such that ( CA = I ).</td>
</tr>
<tr>
<td>* A has a pivot position in every row.</td>
<td>c. ( A ) has ( n ) pivot positions.</td>
<td>* A has a pivot position in every column.</td>
</tr>
<tr>
<td>h. The columns of ( A ) span ( \mathbb{R}^m ).</td>
<td>m. The columns of ( A ) form a basis for ( \mathbb{R}^n ).</td>
<td>e. The columns of ( A ) are linearly independent.</td>
</tr>
<tr>
<td>g. The equation ( A x = b ) has at least one solution for each ( b ) in ( \mathbb{R}^m ).</td>
<td>* The equation ( A x = b ) has a unique solution for each ( b ) in ( \mathbb{R}^n ).</td>
<td>d. The equation ( A x = 0 ) has only the trivial solution.</td>
</tr>
<tr>
<td>i. The transformation ( x \mapsto A x ) maps ( \mathbb{R}^n ) onto ( \mathbb{R}^m ).</td>
<td>* The transformation ( x \mapsto A x ) is invertible.</td>
<td>f. The transformation ( x \mapsto A x ) is one-to-one.</td>
</tr>
<tr>
<td>n. ( \text{Col} \ A = \mathbb{R}^n ).</td>
<td>b. ( A ) is row equivalent to ( I ).</td>
<td>q. ( \text{Nul} \ A = {0} ).</td>
</tr>
<tr>
<td>o. ( \text{dim} \ \text{Col} \ A = m ).</td>
<td>i. ( A^T ) is invertible.</td>
<td>r. ( \text{dim} \ \text{Nul} \ A = 0 ).</td>
</tr>
<tr>
<td>* rank ( A = m ).</td>
<td>p. rank ( A = n ).</td>
<td>* rank ( A = p ).</td>
</tr>
</tbody>
</table>

With so many concepts in your linear algebra vocabulary, you need to be careful not to combine terms in ways that are undefined, even though they may sound reasonable to you. For example, after you finish your work on this section, you should recognize that the following phrases (which have appeared on my students’ papers) are meaningless: “the basis of a matrix,” “the dimension of a basis,” and “the rank of a basis.”

### SOLUTIONS TO EXERCISES

1. \[
\begin{bmatrix}
  1 & -4 & 9 & -7 \\
  -1 &  2 & -4 &  1 \\
  5 & -6 & 10 &  7
\end{bmatrix}
\sim
\begin{bmatrix}
  1 & 0 & -1 &  5 \\
  0 &  2 &  5 & -6 \\
  0 &  0 &  0 &  0
\end{bmatrix}
\]

Look at \( B \), and conclude that \( A \) has two pivot columns and the equation \( A x = 0 \) has two free variables. So rank \( A = 2 \) and dim \( \text{Nul} \ A = 2 \). In fact, the first two columns of \( A \) are pivot columns, so

\[
\text{Basis for Col} \ A : \left\{ \begin{bmatrix} 1 \\ -1 \\ 5 \end{bmatrix}, \begin{bmatrix} -4 \\ 2 \\ -6 \end{bmatrix} \right\}
\]

For the row space, use the rows in the echelon form \( B \). That is,
Basis for Row $A$: $\{(1, 0, -1, 5), (0, -2, 5, -6)\}$

For the null space, use the reduced echelon form of $A$ to solve $Ax = 0$:

$$A \sim B \sim \begin{bmatrix} 1 & 0 & -1 & 5 \\ 0 & 1 & -5/2 & 3 \end{bmatrix} ; \begin{cases} x_1 - x_3 + 5x_4 = 0 \\ x_2 - (5/2)x_3 + 3x_4 = 0 \\ 0 = 0 \end{cases}$$

Thus $x_1 = x_3 - 5x_4$, $x_2 = (5/2)x_3 - 3x_4$, with $x_3$, $x_4$ free. The general solution of $Ax = 0$ is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_3 - 5x_4 \\ (5/2)x_3 - 3x_4 \\ x_3 \\ x_4 \end{bmatrix} = x_3 \begin{bmatrix} 1 \\ 5/2 \\ 1 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} -5 \\ -3 \\ 0 \\ 1 \end{bmatrix} = x_3 u + x_4 v$$

Thus $\{u, v\}$ is a basis for Nul $A$.

**Study Tip:** Because rank $A = 2$ in Exercise 1, any two linearly independent columns of $A$ form a basis for Col $A$, and any two linearly independent rows of $A$ form a basis for Row $A$. When the rank of a matrix exceeds 2, selecting bases in this way is not so easy. (That is why you examine an echelon form of $A$.) On an exam, you should always choose the pivot columns of $A$ as the basis for Col $A$ and the nonzero rows of an echelon form of $A$ as the basis for Row $A$. This will show that you can handle matrices with any rank.

7. Yes, Col $A = \mathbb{R}^4$, because Col $A$ is a 4-dimensional subspace of $\mathbb{R}^4$ and hence coincides with $\mathbb{R}^4$. No, Nul $A$ cannot be $\mathbb{R}^3$, because the vectors in Nul $A$ have 7 entries. Nul $A$ is a 3-dimensional subspace of $\mathbb{R}^7$, by the Rank Theorem.

13. If $A$ is either a $7 \times 5$ matrix or a $5 \times 7$ matrix, then $A$ has at most 5 pivot positions. So 5 is the largest possible value for rank $A$.

17. a. See the paragraph before Example 1.
   b. See the warning after Example 2.
   c. See the Rank Theorem.
   d. See the Rank Theorem.
   e. See the Numerical Note before the Practice Problem.

19. Visualize the system as $Ax = 0$ where $A$ is a $5 \times 6$ matrix. The information in the problem implies that the solution space is one-dimensional. By the Rank Theorem, rank $A = 6 - 1 = 5$. So dim Col $A = 5$. But Col $A$ is a subspace of $\mathbb{R}^5$. Hence Col $A = \mathbb{R}^5$. Thus $Ax = b$ has a solution for all $b$.

**Study Tip:** Exercises 19–25 make good exam questions.

21. Visualize the system as $Ax = b$, where $A$ is $9 \times 10$ matrix. You are told that the system has a solution for all $b$, so $A$ must have a pivot position in each row. (That is, rank $A = 9$.) Since $A$
has 10 columns, the Rank Theorem implies that dim Nul $A = 1$. So it is not possible to find two linearly independent vectors in Nul $A$.

23. The set of interest is the null space of a $12 \times 8$ matrix $A$. The description of this set implies that dim Nul $A = 2$. By the Rank Theorem, rank $A = 8 - 2 = 6$. So the equation $Ax = 0$ is equivalent to $Bx = 0$, where $B$ is an echelon form of $A$ with 6 nonzero rows. The answer to the question is that six homogeneous equations are sufficient.

25. Let $A$ be the $10 \times 12$ coefficient matrix. By hypothesis, there are three free variables in the system $Ax = b$, so dim Nul $A = 3$. By the Rank Theorem, dim Col $A = 12 - 3 = 9$. Since Col $A$ is a subspace of $\mathbb{R}^{10}$ (because $A$ has 10 rows), Col $A$ cannot be all of $\mathbb{R}^{10}$, so some nonhomogeneous equations $Ax = b$ will not have solutions.

31. The solution is in the text.

33. Let $A = [u \ u_2 \ u_3]$. If $u \neq 0$, then $u$ must be a basis for Col $A$, since Col $A$ is one-dimensional. Hence there exist scalars $r$ and $s$ such that $u_2 = ru$ and $u_3 = su$, so that

$$A = [u \ ru \ su] = u[1 \ r \ s] = uv^T,$$
where $v = \begin{bmatrix} 1 \\ r \\ s \end{bmatrix}$.

If the first column of $A$ is zero and the second column, call it $u$, is nonzero, then

$$A = [0 \ u \ ru]$$
for some $r$. In this case, take $v = (0, 1, r)$. If $A = [0 \ 0 \ u]$, take $v = (0, 0, 1)$.

35. a. Let $C$ and $N$ be the matrices you construct whose columns are bases for Col $A$ and Nul $A$, respectively. For the specific $5 \times 7$ matrix $A$ in this problem, rank $A = 4$ and dim Nul $A = 3$. So $C$ should be $5 \times 4$ (because Col $A$ is a four-dimensional subspace of $\mathbb{R}^5$), and $N$ should be $7 \times 3$ (because Nul $A$ is a three-dimensional subspace of $\mathbb{R}^7$). Also, if the rows of $R$ form a basis for Row $A$, then $R$ should be $4 \times 7$, because dim Row $A = \text{rank } A = 4$ and Row $A$ is a subspace of $\mathbb{R}^7$. (Make sure you understand these statements.)

b. The matrix $S = [R^T \ N]$ is $7 \times 7$, because $R^T$ is $7 \times 4$ and $N$ is $7 \times 3$. If $M$ is a matrix whose columns form a basis for Nul $A^T$, then $M$ should be $5 \times 1$, because Nul $A^T$ is a one-dimensional subspace of $\mathbb{R}^5$, by Exercise 28(b). Since $C$ is $5 \times 4$, the matrix $T[C \ M]$ should be $5 \times 5$.

In general, the matrix $S$ is $n \times n$ because the dimensions of Row $A$ (spanned by the columns of $R^T$) and Nul $A$ add up to $n$, the number of columns of $A$, and both Row $A$ and Nul $A$ are subspaces of $\mathbb{R}^n$. The matrix $T$ is $m \times m$ because the dimensions of the column space and Nul $A^T$ add up to $m$, the number of rows of $A$, and both Col $A$ and Nul $A$ are subspaces of $\mathbb{R}^m$.
Mastering Linear Algebra Concepts: Eight Basic Ideas

Sometime between now and when you finish the chapter, you should do a major review of the eight key concepts introduced in this chapter: vector space, subspace, column space, null space, basis, coordinate vector, dimension, and rank. (The row space of $A$ is not really a separate concept; it is just the column space of $A^T$.) Study your old review sheets, and prepare new summary sheets for coordinate vector, dimension, and rank. Use as many of the standard categories (special cases, examples, algorithms, etc.) as possible. The tables in this section will be helpful. Also, add cross-references about dimension and rank to other sheets (subspace, column space, etc.) and update your summary sheet for the Invertible Matrix Theorem.

MATLAB ref, rank, and randomint

In this course, you can use either \texttt{ref(A)} or \texttt{rank(A)} to check the rank of $A$. In practical work, you should use the more reliable command \texttt{rank(A)}, based on the singular value decomposition (Section 7.4).

The Laydata5 command \texttt{randomint(m,n)} produces an $m \times n$ matrix with integer entries between $-9$ and $9$.

4.7 CHANGE OF BASIS

This section will help you better understand coordinate systems. A review of Section 4.4 now is strongly recommended.

KEY IDEAS

Figure 1 and the accompanying discussion will help you visualize the main idea of the section. Imagine superimposing the $C$-graph paper (Figure 1-b) on the $B$-graph paper (Figure 1-a). Can you see where $b_1$ will lie on the $C$-coordinate system? Four units in the $c_1$-direction and one unit in the $c_2$-direction. That is the geometric interpretation of the equation $[b_2]_C = \begin{bmatrix} 4 \\ 1 \end{bmatrix}$ in Example 1.

Similarly, since $[b_2]_C = \begin{bmatrix} -6 \\ 1 \end{bmatrix}$, $b_2$ lies six units in the negative $c_1$-direction and one unit in the $c_2$-direction.

In general, the locations of $b_1$ and $b_2$ on the $C$-graph paper are precisely what you must find in order to build the columns of the change-of-coordinates matrix:

$$P_{C \rightarrow B} = \begin{bmatrix} [b_1]_C \\ [b_2]_C \end{bmatrix}$$
The notation for this matrix should help you remember the basic equation for changing $\mathcal{B}$-coordinates into $\mathcal{C}$-coordinates:

$$[x]_C = P_{c\rightarrow B} [x]_B$$

The calculations are simple when $\mathcal{B}$ and $\mathcal{C}$ are bases for $\mathbb{R}^n$. The box after Example 2 illustrates the algorithm for computing the change-of-coordinates matrix. In general,

$$[c_1 \cdots c_n | b_1 \cdots b_n] \sim [I | P_{c\rightarrow B}]$$

Equivalently, using the notation of Section 4.4,

$$[P_C P_B] \sim [I | P_{c\rightarrow B}]$$

where $P_B$ is the matrix $[b_1 \cdots b_n]$ that changes $\mathcal{B}$-coordinates to standard coordinates, and $P_C$ is similarly defined. If you refer back to Exercise 15 of Section 2.2, you will see that $P_{c\rightarrow B}$ is the same as $(P_C)^{-1} P_B$. Since $(P_C)^{-1}$ changes standard coordinates to $\mathcal{C}$-coordinates, you can obtain $[x]_C$ from $[x]_B$ as follows:

$$[x]_B \xrightarrow{\text{B-coordinates}} P_B [x]_B \xrightarrow{\text{standard coordinates}} (P_C)^{-1} P_B [x]_B = P_{c\rightarrow B} [x]_B = [x]_C$$

This diagram provides another way of viewing the change of coordinates.

**SOLUTIONS TO EXERCISES**

1. a. From $b_1 = 6c_1 - 2c_2$ and $b_2 = 9c_1 - 4c_2$, write

$$[b_1]_C = \begin{bmatrix} 6 \\ -2 \end{bmatrix}, [b_2]_C = \begin{bmatrix} 9 \\ -4 \end{bmatrix}, \text{ and } P_{c\rightarrow B} = \begin{bmatrix} 6 & 9 \\ -2 & -4 \end{bmatrix}$$

b. Since $x = -3b_1 + 2b_2$,

$$[x]_B = \begin{bmatrix} -3 \\ 2 \end{bmatrix} \text{ and } [x]_C = \begin{bmatrix} 6 & 9 \\ -2 & -4 \end{bmatrix}^{-1} \begin{bmatrix} -3 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \end{bmatrix}$$

7. Unlike Exercise 1, you do not have direct information from which you can write $[b_1]_C$ and $[b_2]_C$. Rather than compute these two coordinate vectors separately, use the algorithm from Example 2:

$$[c_1 \ c_2 \ b_1 \ b_2] = \begin{bmatrix} 1 & -2 & 7 & -3 \\ -5 & 2 & 5 & -1 \end{bmatrix} \sim \begin{bmatrix} 1 & -2 & 7 & -3 \\ 0 & -8 & 40 & -16 \end{bmatrix}$$

$$\sim \begin{bmatrix} 1 & -2 & 7 & -3 \\ 0 & 1 & -5 & 2 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & -3 & 1 \\ 0 & 1 & -5 & 2 \end{bmatrix}$$

Thus $P_{c\rightarrow B} = \begin{bmatrix} -3 & 1 \\ -5 & 2 \end{bmatrix}$. The change-of-coordinates matrix from $\mathcal{C}$ to $\mathcal{B}$ is
11. \( \text{See Theorem 15.} \)
\( \text{b. See the first paragraph in the subsection on \textit{Change of Basis} in } \mathbb{R}^n. \)

13. Let \( \mathbf{b}_1 \) represent the polynomial \( 1 - 2t + t^2 \), let \( \mathbf{b}_2 \) be \( 3 - 5t + 4t^2 \), let \( \mathbf{b}_3 \) be \( 2t + 3t^2 \), and let \( \mathcal{C} \) be the standard basis \( \{1, t, t^2\} \) for \( \mathbb{P}^2 \). The \( \mathcal{C} \)-coordinate vectors of the vectors \( \mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3 \) are:

\[
\begin{bmatrix}
\mathbf{b}_1 \rangle_{\mathcal{C}} &=& \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}, \\
\mathbf{b}_2 \rangle_{\mathcal{C}} &=& \begin{bmatrix} 3 \\ -5 \\ 4 \end{bmatrix}, \\
\mathbf{b}_3 \rangle_{\mathcal{C}} &=& \begin{bmatrix} 0 \\ 2 \\ 3 \end{bmatrix}, \text{ and}
\end{bmatrix}
\]

\[
P_{\mathcal{C} \rightarrow \mathcal{B}} = \begin{bmatrix} 1 & 3 & 0 \\ -2 & -5 & 2 \\ 1 & 4 & 3 \end{bmatrix}
\]

The coordinate vector \([-1 + 2t]_B\) satisfies

\[
P_{\mathcal{C} \rightarrow \mathcal{B}}[-1 + 2t]_a = [-1 + 2t]_c = \begin{bmatrix} -1 \\ 2 \\ 0 \end{bmatrix}
\]

This equation can be solved by row reduction:

\[
\begin{bmatrix} 1 & 3 & 0 & -1 \\ -2 & -5 & 2 & 2 \\ 1 & 4 & 3 & 0 \end{bmatrix} \sim ... \sim \begin{bmatrix} 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & -2 \\ 0 & 0 & 1 & 1 \end{bmatrix} ; \begin{bmatrix} -1 + 2t \rangle_B = \begin{bmatrix} 5 \\ -2 \\ 1 \end{bmatrix} \]
\]

17. \( [M] \)
\( \text{a. Since we found } P \text{ in Exercise 34 of Section 4.5, we can calculate that} \)
\[
P^{-1} = \frac{1}{32} \begin{bmatrix} 32 & 0 & 16 & 0 & 12 & 0 & 10 \\ 0 & 32 & 0 & 24 & 0 & 20 & 0 \\ 0 & 0 & 16 & 0 & 16 & 0 & 15 \\ 0 & 0 & 0 & 8 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 6 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
\]

\( \text{b. Since } P \text{ is the change-of-coordinates matrix from } \mathcal{C} \text{ to } \mathcal{B}, \ P^{-1} \text{ will be the change-of-coordinates matrix from } \mathcal{B} \text{ to } \mathcal{C}. \text{ By Theorem 15, the columns of } P^{-1} \text{ will be the } \mathcal{C} \)-coordinate vectors of the basis vectors in } \mathcal{B}. \text{ Thus} \)
\[
\begin{align*}
\cos^2 t &= \frac{1}{2}(1 + \cos 2t) \\
\cos^3 t &= \frac{1}{4}(3\cos t + \cos 3t) \\
\cos^4 t &= \frac{1}{8}(3 + 4\cos 2t + \cos 4t) \\
\cos^5 t &= \frac{1}{16}(10\cos t + 5\cos 3t + \cos 5t) \\
\cos^6 t &= \frac{1}{32}(10 + 15\cos 2t + 6\cos 4t + \cos 6t)
\end{align*}
\]

19. a. If \( P \) is to be the change-of-coordinates matrix from \{\(u_1, u_2, u_3\}\} to \{\(v_1, v_2, v_3\}\}, then the columns of \( P \) should be \( C \)-coordinate vectors, where \( C = \{v_1, v_2, v_3\}\). That is, the columns of \( P \) should be \( [u_1]_C \), \( [u_2]_C \), and \( [u_3]_C \). You know \( P \), but you do not know \( u_1 \), \( u_2 \), or \( u_3 \). Ask yourself, for example, what is the meaning of \( [u_1]_C \)? By definition, a \( C \)-coordinate vector tells how to build a vector out of the \( C \)-basis vectors, \( v_1 \), \( v_2 \), and \( v_3 \). So, for \( j = 1, 2, 3 \),

\[
[u_j]_C = [v_1 \ v_2 \ v_3][u_j]_C = V[u_j]_C
\]

where \( V = [v_1 \ v_2 \ v_3] \). Then, by the definition of matrix multiplication,

\[
[u_1 \ u_2 \ u_3] = [V[u_1]_C \ V[u_2]_C \ V[u_3]_C] = V[u_1]_C \ V[u_2]_C \ V[u_3]_C = VP
\]

You know \( V \) and \( P \), so you can compute \( u_1 \), \( u_2 \), and \( u_3 \).

\[
VP = \begin{bmatrix}
-2 & -8 & -7 \\
2 & 5 & 2 \\
3 & 2 & 6
\end{bmatrix}
\begin{bmatrix}
1 & 2 & -1 \\
-3 & -5 & 0 \\
4 & 6 & 1
\end{bmatrix}
= \begin{bmatrix}
-6 & -6 & -5 \\
-5 & -9 & 0 \\
21 & 32 & 3
\end{bmatrix}
\]

Thus, \( u_1 = \begin{bmatrix} -6 \\ -5 \\ 21 \end{bmatrix} \), \( u_2 = \begin{bmatrix} -6 \\ -9 \\ 32 \end{bmatrix} \), \( u_3 = \begin{bmatrix} -5 \\ 0 \\ 3 \end{bmatrix} \).

b. Stop here! Part (a) was fairly difficult. If you were not able to work it by yourself and you have read the solution, then you should try part (b) by yourself. Here are the steps of the solution:

(i) Write in symbols what the columns of \( P \) should be.

(ii) Decide how these columns are related to the matrix \( W = [w_1 \ w_2 \ w_3] \).
(iii) Obtain a matrix equation that involves $W$ in some way.
(iv) Compute $W$, and list its columns as the answer to the problem.

Study the solution to (a), close the Study Guide, and work on (b) as if it were a new problem. The solution of (b) is at the end of the solutions for Section 4.8.

**MATLAB Change-of-Coordinates Matrix**

The Laydata5 Toolbox has data for Exercises 7–10 and 17–19. The command `ref(M)` row reduces a matrix such as $\begin{bmatrix} c_1 & c_2 & b_1 & b_2 \end{bmatrix}$ to the desired form.

### 4.8 APPLICATIONS TO DIFFERENCE EQUATIONS

Difference equations are the discrete analogues of differential equations. Both are important in science and engineering. The discrete and continuous theories are remarkably parallel, and linear algebra is applied in similar ways, although the calculations are somewhat easier for difference equations. A variety of examples and exercises here illustrate some difference equations you may encounter later in your work.

#### KEY IDEAS

Each signal in $\mathbb{S}$ is an infinite list of numbers. Linear independence of a set of signals can often be demonstrated by looking at short segments of the signals, that is, by showing that a Casorati matrix is invertible. A Casorati matrix cannot be used in general to demonstrate linear dependence of a set. However, see the appendix at the end of this Study Guide section. The main focus of the section is on difference equations. Given a homogeneous difference equation, you should be able to:

- determine whether a specified signal is a solution of the equation;
- find solutions of the equation, using the auxiliary equation;
- give the general solution (when the auxiliary equation has no multiple roots and no complex roots).

Theorem 17 is the key result that enables you to write the general solution. Just finding some specific solutions is not enough; you must show that they span the set of all solutions. But Theorem 17 and the Basis Theorem in Section 4.5 together show that for an $n$th order equation, you only need to find $n$ linearly independent solutions. (See Example 5.) The same principle applies to an $n$th order differential equation (discussed later in Section 5.7). This principle is one of the most powerful applications of linear algebra in the text.

The subsection on nonhomogeneous equations is optional. If this is covered, you should be able to work Exercises 25–28. The general principle is illustrated in Figure 4, page 250:
The final subsection shows the modern way to study an $n$th order linear difference equation, rewriting it as a first order system $x_{k+1} = Ax_k$ ($k = 1, 2, \ldots$). Such systems were introduced in Section 1.10 and they will be discussed further in Sections 4.9 and 5.6.

**SOLUTIONS TO EXERCISES**

1. If $y_k = (-4)^k$, then $y_{k+1} = (-4)^{k+1}$ and $y_{k+2} = (-4)^{k+2}$. Substitute these formulas into the left side of the equation:

$$y_{k+2} + 2y_{k+1} = 8y_k = (-4)^{k+2} + 2(-4)^{k+1} - 8(-4)^k$$

$$= (-4)^k[-4^2 + 2(-4) - 8]$$

$$= (-4)^k[16 - 8 - 8] = 0 \text{ for all } k$$

Since the difference equation holds for all $k$, $(-4)^k$ is a solution. The text answer displays the similar calculations for $y_k = 2^k$.

7. Compute the Casorati matrix for the signals $1^k, 2^k,$ and $(-2)^k$, setting $k = 0$ for convenience:

$$\begin{bmatrix}
1^0 & 2^0 & (-2)^0 \\
1^1 & 2^1 & (-2)^1 \\
1^2 & 2^2 & (-2)^2
\end{bmatrix} = \begin{bmatrix}
1 & 1 & 1 \\
1 & 2 & -2 \\
1 & 4 & 4
\end{bmatrix} \sim \begin{bmatrix}
1 & 1 & 1 \\
0 & 1 & -3 \\
0 & 3 & 3
\end{bmatrix} \sim \begin{bmatrix}
1 & 1 & 1 \\
0 & 1 & -3 \\
0 & 0 & 12
\end{bmatrix}$$

This Casorati matrix has three pivots and hence is invertible, by the IMT. Hence the set of signals $\{1^k, 2^k, (-2)^k\}$ is linearly independent in $\mathbb{S}$. We know (from the text) that these signals are in the solution space $H$ of a third-order difference equation. By Theorem 17, $\dim H = 3$. Since the three signals are linearly independent, they form a basis for $H$, by the Basis Theorem in Section 4.5.

**Warning:** Many student papers for Exercise 7 suffer from a lack of precision, often confusing linear independence of the columns of the Casorati matrix with linear independence of the signals in $\mathbb{S}$. There is no need to discuss the columns of the Casorati matrix—just observe that the matrix is invertible. But you must point out that the three signals are linearly independent, in order to apply the Basis Theorem to the vector space $H$ of solutions to the difference equation.

13. The auxiliary equation for $y_{k+2} - y_{k+1} + \frac{2}{9}y_k = 0$ is $r^2 - r + \frac{2}{9} = 0$. By the quadratic formula,

$$r = \frac{1 \pm \sqrt{1 - 8/9}}{2} = \frac{1 \pm 1/3}{2} = \frac{2}{3} \text{ or } \frac{1}{3}$$

Two solutions of the difference equation are $\left(\frac{2}{3}\right)^k$ and $\left(\frac{1}{3}\right)^k$. These signals are obviously linearly independent because neither is a multiple of the other. Since the solution space is
two-dimensional (Theorem 17), the two signals form a basis for the solution space, by the Basis Theorem.

**Study Tip:** I think Exercises 7–19 (and 25–28) make good test questions because they illustrate how important Theorem 17 and the Basis Theorem really are. Probably, you do not have to remember the specific number of Theorem 17, but your discussion should show that you have it in mind and know how to use it with the Basis Theorem. (Check with your instructor.)

19. The auxiliary equation for \( y_{k+2} + 4y_{k+1} + y_k = 0 \) is \( r^2 + 4r + 1 = 0 \). By the quadratic formula,

\[
    r = \frac{-4 \pm \sqrt{16-4}}{2} = \frac{-4 \pm 2\sqrt{3}}{2} = -2 \pm \sqrt{3}
\]

Two solutions of the difference equation are \((-2 + \sqrt{3})^k\) and \((-2 - \sqrt{3})^k\). They are obviously linearly independent because neither is a multiple of the other. Since the solution space is two-dimensional (Theorem 17), the two signals form a fundamental set of solutions by the Basis Theorem, and the general solution has the form \( c_1(-2 + \sqrt{3})^k + c_2(-2 - \sqrt{3})^k \).

25. To prove that \( y_k = k^2 \) is a solution of

\[
    y_{k+2} + 3y_{k+1} - 4y_k = 10k + 7 \quad \text{(1)}
\]

show that when \( k^2, (k+1)^2, \) and \((k+2)^2\) are substituted for \( y_k, y_{k+1}, \) and \( y_{k+2} \), respectively, the resulting equation is true for all \( k \):

\[
    (k + 2)^2 + 3(k+1)^2 - 4k^2 = (k^2 + 4k + 4) + 3(k^2 + 2k + 1) - 4k^2 \\
    = (1+3-4)k^2 + (4+6)k + (4+3) \\
    = 10k + 7 \text{ for all } k
\]

So \( k^2 \) is a solution of (1). The auxiliary equation for the homogeneous difference equation

\[
    y_{k+2} + 3y_{k+1} - 4y_k = 0 \text{ for all } k \quad \text{(2)}
\]

is \( r^2 + 3r - 4 = 0 \), which factors as \((r-1)(r+4) = 0\), so \( r = 1, -4 \). Thus \( 1^k \) and \((-4)^k\) are solutions of (2). The signals are linearly independent (for neither is a multiple of the other), so they form a basis for the two-dimensional solution space. The general solution of (2) is \( c_1 \cdot 1^k + c_2(-4)^k \). Add this to a particular solution of (1) and obtain the general solution \( k^2 + c_1 + c_2(-4)^k \) of (1).

31. The full explanation is in the text’s answer section.

35. For \( \{y_k\} \) and \( \{z_k\} \) in \( \mathbb{S} \), the \( k \)th term of \( \{y_k\} + \{z_k\} \) is \( y_k + z_k \). Hence

\[
    T(\{y_k\} + \{z_k\}) = (y_{k+2} + z_{k+2}) + a(y_{k+1} + z_{k+1}) + b(y_k + z_k) \\
    = (y_{k+2} + ay_{k+1} + by_k) + (z_{k+2} + az_{k+1} + bz_k) \\
    = T\{y_k\} + T\{z_k\}
\]

For any scalar \( r \), the \( k \)th term of \( r\{y_k\} \) is \( ry_k \), and so
Thus \( T \) has the two properties that define a linear transformation.

37. We compute that
\[
(TD)(y_0, y_1, y_2, \ldots) = T(D(y_0, y_1, y_2, \ldots)) = T(0, y_0, y_1, y_2, \ldots) = (y_0, y_1, y_2, \ldots)
\]
while
\[
(DT)(y_0, y_1, y_2, \ldots) = D(T(y_0, y_1, y_2, \ldots)) = D(y_1, y_2, y_3, \ldots) = (0, y_1, y_2, y_3, \ldots)
\]
Thus \( TD = I \) (the identity transformation on \( \mathbb{S}_0 \)), while \( DT \neq I \).

19. b. \textit{(This solution is for Section 4.7.)} The columns of a change-of-coordinates matrix \( P \) from \( \{v_1, v_2, v_3\} \) to \( \{w_1, w_2, w_3\} \) are \( D \)-coordinate vectors, where \( D = \{w_1, w_2, w_3\} \). That is, the columns of \( P \) are \([v_1]_D \), \([v_2]_D \), and \([v_3]_D \). How are these columns related to the matrix \( W = [w_1 \ w_2 \ w_3] \)? A \( D \)-coordinate vector tells how to build a vector out of the \( D \)-basis vectors (the columns of the matrix \( W \)). For \( j = 1, 2, 3 \),
\[
v_j = [w_1 \ w_2 \ w_3][v_j]_D = W[v_j]_D
\]
By definition of matrix multiplication,
\[
V = [v_1 \ v_2 \ v_3] = [W[v_1]_D \ W[v_2]_D \ W[v_3]_D] = W[[v_1]_D \ [v_2]_D \ [v_3]_D] = WP
\]
You know \( V \) and \( P \), so compute \( W \) from \( VP^{-1} = W \). Use MATLAB or other matrix program to compute \( P^{-1} \). Then
\[
W = VP^{-1} = \begin{bmatrix}
-2 & -8 & -7 \\
5 & 8 & 5 \\
2 & 5 & 2 \\
3 & 2 & 6 \\
\end{bmatrix} = \begin{bmatrix}
28 & 38 & 21 \\
-9 & -13 & -7 \\
-3 & 2 & 3 \\
\end{bmatrix}
\]
Thus, \( w_1 = [28] \), \( w_2 = [38] \), \( w_3 = [21] \).

---

**Appendix: The Casorati Test**

Let \( \{y_1, \ldots, y_n\} \) be a set of signals in \( \mathbb{S} \). For \( j = 1, \ldots, n \) and for any \( k \), let \( y_j(k) \) denote the \( k \)th entry in the signal \( y_j \) and let
\[
C(k) = \begin{bmatrix}
y_1(k) & \cdots & y_n(k) \\
y_1(k+1) & \cdots & y_n(k+1) \\
\vdots & & \vdots \\
y_1(k+n-1) & \cdots & y_n(k+n-1)
\end{bmatrix}
\]
The Casorati matrix

a. If \( C(k) \) is invertible for some \( k \), \( \{y_1, \ldots, y_n\} \) is linearly independent.

b. If \( y_1, \ldots, y_n \) all satisfy a homogeneous difference equation of order \( n \),

Copyright © 2016 Pearson Education, Inc.
\[ y_{k+n} + a_1 y_{k+n-1} + \cdots + a_n y_k = 0 \] for all \( k \) \hspace{1cm} (\ast)

(with \( a_n \neq 0 \)), and if the Casorati matrix \( C(k) \) is not invertible for some \( k \), then \( \{y_1, \ldots, y_n\} \) is linearly dependent in \( S \), and for all \( k \), \( C(k) \) is not invertible.

**Proof.** (a) The argument given in the text (page 245) for a set of three signals generalizes immediately to \( n \) signals. (b) Suppose that \( y_1, \ldots, y_n \) are in the set \( H \) of solutions of \((\ast)\) and \( C(k_0) \) is not invertible for some \( k_0 \). It is readily verified that if \( T: H \rightarrow \mathbb{R}^n \) is defined by

\[
T(y) = \begin{bmatrix}
y(k_0) \\
y(k_0 + 1) \\
\vdots \\
y(k_0 + n - 1)
\end{bmatrix}
\]

then \( T \) is a linear transformation. The proof of Theorem 16 is easily modified to show that \((\ast)\) has a unique solution \( y \) whenever \( y(k_0), \ldots, y(k_0 + n - 1) \) are specified. This means that \( T \) is a one-to-one mapping of \( H \) onto \( \mathbb{R}^n \). Furthermore, the images \( T(y_1), \ldots, T(y_n) \) form the columns of the Casorati matrix \( C(k_0) \) and hence are linearly dependent, because \( C(k_0) \) is not invertible. Since \( T \) is one-to-one, \( \{y_1, \ldots, y_n\} \) is linearly dependent, by Exercise 32 in Section 4.3. This proves the first statement in (b). The second statement follows immediately from part (a), because if \( C(k) \) were invertible for some \( k \), then \( \{y_1, \ldots, y_n\} \) would be linearly independent, which is not true. So \( C(k) \) is not invertible for each \( k \).

**MATLAB roots**

In Exercises 7–16 and 25–28, the polynomial in the auxiliary equation is stored in a row vector \( p \), with coefficients in descending order. For instance, if the auxiliary equation is \( r^2 + 6r + 9 = 0 \), then \( p = [1 \ 6 \ 9] \)

The MATLAB command `roots(p)` produces a column vector whose entries are the roots of the polynomial described by \( p \).

**4.9 APPLICATIONS TO MARKOV CHAINS**

This section builds on the population movement example in Section 1.9. You should review that example now. Markov chains are widely used in applications and there is a rich theory connected with them. The simple examples and exercises in this section provide a basic foundation on which you can build later as needed. Two of the examples here will be analyzed from a different point of view in Section 5.2.
KEY IDEAS

A probability vector is a list of nonnegative numbers that sum to one. A Markov chain is a sequence of probability vectors \( \{x_k\} \) that satisfy a difference equation \( x_{k+1} = Px_k \) \((k = 0, 1, \ldots)\) for some stochastic matrix \( P \) (whose columns are themselves probability vectors).

The theory of this chapter can be used to show that \( P - I \) always has a nontrivial null space when \( P \) is a stochastic matrix (Exercise 17). Advanced texts show that the null space of \( P - I \) always includes at least one probability vector, which then is a steady-state vector for \( P \), because the equation \( (P - I)q = 0 \) is equivalent to \( Pq = q \). (Also, see Exercise 18.)

Our main interest is in a regular stochastic matrix \( P \). In this case the steady-state vector is unique, according to Theorem 18. The key to predicting the distant future for a Markov chain associated with such a \( P \) is to find the steady-state vector \( q \), since the sequence \( \{x_k\} \) converges to \( q \) no matter what the initial state.

SOLUTIONS TO EXERCISES

1. a. To set up the stochastic matrix \( P \), label the columns \( N \) (for news) and \( M \) (for music) in some order; use the same order for the rows. (Failure to keep the same order is a common source of error in this type of problem.) The data should be arranged so you read down a column and then to the right along a row.

   From:
   
   \[
   \begin{array}{cc}
   N & M \\
   .7 & .6 \\
   .3 & .4 \\
   \end{array}
   \]
   
   To:
   
   \[
   \begin{array}{c}
   \text{News} \\
   \text{Music} \\
   \end{array}
   \]

   b. You are told that 100\% of the listeners are listening to the news at 8:15 a.m., so start the Markov chain then, with \( x_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \).

   c. There are two breaks between 8:15 and 9:25, so you need \( x_2 \)

   \[
   x_1 = Px_0 = \begin{bmatrix} .7 & .6 \\ .3 & .4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} .7 \\ .3 \end{bmatrix}
   \]

   \[
   x_2 = Px_1 = \begin{bmatrix} .7 & .6 \\ .3 & .4 \end{bmatrix} \begin{bmatrix} .7 \\ .3 \end{bmatrix} = \begin{bmatrix} .67 \\ .33 \end{bmatrix}
   \]
The entries in $x_2$ show that after two station breaks, 67% of the audience is listening to the news and 33% is listening to music.

**Study Tip:** When you compute a typical probability vector $Px$, be sure to compute all of the entries in the product $Px$. Then check your work by verifying that the entries sum to 1.

7. To find the steady state vector for a regular stochastic matrix $P$:
   (i) set up the matrix $P - I$;
   (ii) find the general solution of $(P - I)x = 0$;
   (iii) choose a basis vector for $\text{Nul}(P - I)$ whose entries sum to 1.

$$P = \begin{bmatrix} .7 & .1 & .1 \\ .2 & .8 & .2 \\ .1 & .1 & .7 \end{bmatrix}, P - I = \begin{bmatrix} .7 & .1 & .1 \\ .2 & .8 & .2 \\ .1 & .1 & .7 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -.3 & .1 & .1 \\ .2 & -.2 & .2 \\ .1 & .1 & -.3 \end{bmatrix}$$

Solve $(P - I)x = 0$:

$$\begin{bmatrix} -.3 & .1 & .1 & 0 \\ .2 & -.2 & .2 & 0 \\ .1 & .1 & -.3 & 0 \end{bmatrix} \sim \begin{bmatrix} .1 & .1 & -.3 & 0 \\ .1 & .1 & .1 & 0 \\ -3 & .1 & 1 & 0 \end{bmatrix}$$

Interchange rows 1 and 3
Scale row 1 by 10

$$\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x_1 = x_3, \quad x_2 = 2x_3; \quad x_3 = \frac{1}{2}x_3, \quad x_3 = x_3$$

The entries in $x_3$ sum to 4, so $q = \frac{1}{4} = \frac{1}{2}$ or $.50$.

**Study Tip:** Notice that the column sums are all zero for the matrix $I - P$ of Exercise 7. This always happens (see Exercise 17), and so you have a fast way to check your arithmetic for the entries in $P - I$.

**Warning:** You may have noticed that in Exercise 7, I scaled rows by 10, to avoid decimals. A common mistake is to do this only to $P$, before forming $I - P$. That changes $P$ drastically. The scaling I did was permissible because it was applied to all the coefficients in an equation.

13. a. From Exercise 3, $P = \begin{bmatrix} .95 & .45 \\ .05 & .55 \end{bmatrix}$. So $P - I = \begin{bmatrix} -.05 & .45 \\ .05 & -.45 \end{bmatrix}$. Solve $(P - I)x = 0$:

$$\begin{bmatrix} -.05 & .45 & 0 \\ .05 & -.45 & 0 \end{bmatrix} \sim \begin{bmatrix} -05 & .45 & 0 \\ 0 & 0 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & -9 & 0 \\ 0 & 0 & 0 \end{bmatrix} x_1 = 9x_3, \quad x_2 \text{ is free}$$
A basis for $\text{Nul}(P - I)$ is $\begin{bmatrix} 9 \\ 1 \end{bmatrix}$; the steady-state vector is $q = \begin{bmatrix} .9 \\ .1 \end{bmatrix}$.

b. The description in Exercise 3 may be interpreted as saying that the “state” of any specified person (in some group of students) on day $k$ is predicted by a probability vector, say, $x_k$. The second entry in $x_k$ is the probability that the person is ill on day $k$. The starting vector for a specified person is $(1,0)$ if the person is well today, and $(0,1)$ if the person is ill. This situation applies to each person, because the exercise says, for example, that every healthy student has a 95% probability of being healthy the next day. That is, the stochastic matrix $P$ applies to each person in the group.

The question in part (b) is about $x_k$ for a large value of $k$. By Theorem 18, $x_k$ approaches $q$, so it is reasonable to assume that $q$ may be used to answer a question about $x_k$. Thus, the probability is .10 that after many days a specific student is ill. The second question essentially asks, “If $x_0 = (0,1)$, does this have any affect on $x_k$ for large $k$?” No, by Theorem 18, because the sequence $\{x_k\}$ approaches $q$ no matter what $x_0$ is.

19. a. The product $Sx$ equals the sum of the entries in $x$. Thus, by definition, $x$ is a probability vector if and only if its entries are nonnegative and $Sx = 1$.

b. Let $P = [p_1 \ p_2 \ \cdots \ p_n]$, where the $p_i$ are probability vectors. By matrix multiplication and part (a),

$$SP = [Sp_1 \ Sp_2 \ \cdots \ Sp_n] = [1 \ 1 \ \cdots \ 1] = S$$

c. By part (b), $S(Px) = (SP)x = Sx = 1$. The entries in $Px$ are obviously nonnegative, because $P$ and $x$ have only nonnegative entries. By (a), the condition $S(Px) = 1$ shows that $Px$ is a probability vector.

MATLAB `randomstoc`

The MATLAB box for Section 1.10 contains information that is useful here. The command `randomstoc(n)` produces a random $n \times n$ stochastic matrix.

Chapter 4 GLOSSARY CHECKLIST

Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

**auxiliary equation**: A polynomial equation in a variable $r$, created from . . . .

**basis** (for a nonzero subspace $H$): A set $\mathcal{B} = \{v_1, \ldots, v_p\}$ in $V$ such that: . . . .

**$B$-coordinates of $x$**: See coordinates of $x$ relative to the basis $B$. 

Copyright © 2016 Pearson Education, Inc.
change-of-coordinates matrix (from a basis $B$ to a basis $C$): A matrix $P_{C \rightarrow B}$ that transforms . . .

column space (of an $m \times n$ matrix $A$): The set $\text{Col} \ A$ of . . . In set notation, $\text{Col} \ A = \{ : \}.$

controllable (pair of matrices): A matrix pair $(A, B)$ where $A$ is $n \times n$, $B$ has $n$ rows, and . . .

coordinate mapping (determined by an ordered basis $B$ in a vector space $V$): A mapping that associates to each . . .

coordinates of $x$ relative to the basis $B = \{b_1, ..., b_n\}$: . . .

coordinate vector of $x$ relative to $B$: The vector $[x]_B$ whose entries . . .

dimension (of a vector space $V$): The number . . .

explicit description (of a subspace $W$ of $\mathbb{R}^n$): A parametric representation of $W$ as the set of . . .

finite-dimensional (vector space): A vector space that is . . .

full rank (matrix): An $m \times n$ matrix whose rank is . . .

fundamental set of solutions: A . . . for the set of solutions of . . .

fundamental subspaces (determined by $A$): The . . . of $A$, . . .

implicit description (of a subspace $W$ of $\mathbb{R}^n$): A set of one or more . . .

infinite-dimensional (vector space): A nonzero vector space $V$ that . . .

isomorphism: A . . . mapping from one vector space . . .

kernel (of a linear transformation $T: V \rightarrow W$): The set of . . . such that . . .

linear combination: A sum . . .

linear dependence relation: A . . . vector equation where . . .

linear filter: A . . . equation used to transform . . .

linearly dependent (vectors): A set $\{v_1, ..., v_p\}$ with the property that . . .

linearly independent (vectors): A set $\{v_1, ..., v_p\}$ with the property . . .

linear transformation $T$ (from a vector space $V$ into a vector space $W$): A rule $T: V \rightarrow W$ that to each vector $x$ in $V$ assigns a unique vector $T(x)$ in $W$, such that: . . .

Markov chain: A sequence of . . . vectors $v_0, v_1, v_2, . . .$, together with a . . . matrix $P$ such that . . .

maximal linearly independent set (in $V$): A linearly independent set $B$ in $V$ such that if . . . , then . . .

minimal spanning set (for a subspace $H$): A set $B$ that spans $H$ and has the property that if . . . , then . . .

null space (of an $m \times n$ matrix $A$): The set $\text{Nul} \ A$ of all . . . In set notation, $\text{Nul} \ A = \{ : \}.$

probability vector: A vector in $\mathbb{R}^n$ whose entries . . .
proper subspace: Any subspace of a vector space $V$.

range (of a linear transformation $T: V \rightarrow W$): The set of all vectors.

rank (of a matrix $A$): 

regular stochastic matrix: A stochastic matrix $P$ such that.

row space (of a matrix $A$): The set $\text{Row } A$ of all; also denoted by.

signal (or discrete-time signal): An element in the space $S$ of all.

Span{$v_1, \ldots, v_p$}: The set of. Also, the spanned.

spanning set (for a subspace $H$): Any set {$v_1, \ldots, v_p$} such that.

standard basis: The basis for $\mathbb{R}^n$ consisting of, or the basis for $\mathbb{P}_n$.

state vector: A vector. In general, a vector that describes, often in connection with a difference equation.

steady-state vector (for a stochastic matrix $P$): A vector $v$ such that.

stochastic matrix: A matrix whose columns.

submatrix (of $A$): Any matrix obtained by.

subspace: A subset $H$ of some vector space $V$ such that $H$ has these properties.

vector space: A set of objects, called vectors, on which.

zero subspace: The subspace consisting of.
5.1 EIGENVECTORS AND EIGENVALUES

This section introduces eigenvectors and eigenvalues. A hint about the connection with dynamical systems appears at the end of the section.

KEY IDEAS

In words, a nonzero vector \( v \) is an eigenvector of a matrix \( A \) if and only if the transformed vector \( Ax \) points in the same or opposite direction of \( v \).

Notice that while an eigenvalue \( \lambda \) might be zero, an eigenvector is never zero (by definition). An eigenspace contains eigenvectors together with the zero vector.

The two equations \( Ax = \lambda x \) and \( (A - \lambda I)x = 0 \) are equivalent. See Example 3. The first equation is useful for understanding what eigenvalues and eigenvectors are, and it shows the geometric effect of the linear transformation \( x \mapsto Ax \) on an eigenvector. The second equation shows that the eigenspace is a subspace (because it is the null space of the matrix \( A - \lambda I \)), and the equation is used to find a basis for the eigenspace, when \( \lambda \) is a known eigenvalue. The second equation will be used again in Section 5.2 for another purpose.

SOLUTIONS TO EXERCISES

1. The number 2 is an eigenvalue of \( A \) if and only if the equation \( Ax = 2x \) has a nontrivial solution. This equation is equivalent to \( (A - 2I)x = 0 \). Compute

\[
A - 2I = \begin{bmatrix} 3 & 2 \\ 3 & 8 \end{bmatrix} - \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 6 \end{bmatrix}
\]

The columns of \( A \) are obviously linearly dependent, so \( (A - 2I)x = 0 \) has a nontrivial solution, and so 2 is an eigenvalue of \( A \).

7. Proceed as in Exercise 1:
You need to know whether $A - 4I$ is invertible. This could be checked in several ways, but since you are asked for an eigenvector, in the event that one exists, the best strategy is to row reduce the augmented matrix for $(A - 4I)x = 0$:

\[
\begin{bmatrix}
-1 & 0 & -1 & 0 \\
2 & -1 & 1 & 0 \\
-3 & 4 & 1 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
-1 & 0 & -1 & 0 \\
0 & -1 & -1 & 0 \\
0 & 4 & 4 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
-1 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

Now it is clear that 4 is an eigenvalue of $A$ [because $(A - 4I)x = 0$ has a nontrivial solution]. The coordinates of an eigenvector satisfy $-x_1 - x_3 = 0$ and $-x_2 - x_3 = 0$. The general solution is not requested, so take any nonzero value for $x_3$ to produce an eigenvector. If $x_3 = 1$, then $x = (-1, -1, 1)$.

Checkpoint 1: The answer in the text is different, namely, $(1, 1, -1)$. Why is this also correct?

Helpful Hint: Suppose you think that 4 is an eigenvalue of a matrix, as in Exercise 7, and you row reduce the augmented matrix for $(A - 4I)x = 0$. If you discover that there are no free variables, then there are only two possibilities: (1) 4 is not an eigenvalue of $A$, or (2) you have made an arithmetic error.

13. For $\lambda = 1$:

\[
A - I = \begin{bmatrix}
4 & 0 & 1 \\
-2 & 1 & 0 \\
-2 & 0 & 1
\end{bmatrix} - \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix} = \begin{bmatrix}
3 & 0 & 1 \\
-2 & 0 & 0 \\
-2 & 0 & 0
\end{bmatrix}
\]

The equations for $(A - I)x = 0$ are easy to solve: \[
\begin{cases}
3x_1 + x_3 = 0 \\
-2x_1 = 0
\end{cases}
\]

Row operations hardly seem necessary. Obviously $x_1$ is zero, and hence $x_3$ is also zero. There are three-variables, so $x_2$ is free. The general solution of $(A - I)x = 0$ is $x_2e_2$, where $e_2 = (0,1,0)$, and so $e_2$ provides a basis for the eigenspace.

For $\lambda = 2$:

\[
A - 2I = \begin{bmatrix}
4 & 0 & 1 \\
-2 & 1 & 0 \\
-2 & 0 & 1
\end{bmatrix} - \begin{bmatrix}
2 & 0 & 0 \\
0 & 2 & 0 \\
0 & 0 & 2
\end{bmatrix} = \begin{bmatrix}
2 & 0 & 1 \\
-2 & -1 & 0 \\
-2 & 0 & 1
\end{bmatrix}
\]
5.1 • Eigenvectors and Eigenvalues  5-3

\[
\begin{bmatrix}
2 & 0 & 1 & 0 \\
-2 & -1 & 0 & 0 \\
-2 & 0 & -1 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
2 & 0 & 1 & 0 \\
0 & -1 & 1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 1/2 & 0 \\
0 & 1 & -1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

So \(x_1 = -(1/2)x_3, x_2 = x_3,\) with \(x_3\) free. The general solution of \((A - 2I)x = 0\) is

\[
\begin{bmatrix}
1/2 \\
1 \\
1
\end{bmatrix}
\]

A nice basis vector for the eigenspace is

\[
\begin{bmatrix}
-1 \\
2 \\
2
\end{bmatrix}
\]

For \(\lambda = 3:\)

\[
A - 3I =
\begin{bmatrix}
4 & 0 & 1 \\
-2 & 1 & 0 \\
-2 & 0 & 1
\end{bmatrix} -
\begin{bmatrix}
3 & 0 & 0 \\
0 & 3 & 0 \\
0 & 0 & 3
\end{bmatrix}
=
\begin{bmatrix}
1 & 0 & 1 \\
-2 & -2 & 0 \\
-2 & 0 & -2
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 0 & 1 & 0 \\
-2 & -2 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 1 & 0 \\
0 & -2 & 2 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 1 & 0 \\
0 & 1 & -1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

So \(x_1 = -x_3, x_2 = x_3,\) with \(x_3\) free. A basis vector for the eigenspace is

\[
\begin{bmatrix}
-1 \\
1 \\
1
\end{bmatrix}
\]

Study Tip: The text's answer to Exercise 15 is likely to be the same as yours, but there are many answers. What should you do if your vectors differ from those in the answer key? Example 2 gives the answer. Whenever you compute an \(x\) that you think is an eigenvector of \(A\), you can check this simply by computing \(Ax\). There is a little more to do in Exercise 15, however. The answer shows a basis of two eigenvectors, which means that the eigenspace is two-dimensional. So your answer must consist of two linearly independent eigenvectors. You can check that they are indeed eigenvectors, and then their linear independence can be checked by inspection.

19. The matrix

\[
\begin{bmatrix}
1 & 2 & 3 \\
1 & 2 & 3 \\
1 & 2 & 3
\end{bmatrix}
\]

is not invertible because its columns are linearly dependent. So the number \(0\) is an eigenvalue of the matrix. See the discussion following Example 5.

21. a. Carefully read the definition of an eigenvalue.
   b. See the paragraph before Example 5
   c. See the discussion of equation (3).
   d. See Example 2 and the paragraph preceding it. Also, see the Numerical Note.
   e. See the Warning after Example 3.
23. If a $2 \times 2$ matrix $A$ had three distinct eigenvalues, then by Theorem 2 there would correspond three linearly independent eigenvectors (one for each eigenvalue). This is impossible because the vectors all belong to a two-dimensional vector space in which any set of three vectors is linearly dependent. See Theorem 8 in Section 1.6. In general, if an $n \times n$ matrix has $p$ distinct eigenvalues, then by Theorem 2 there would be a linearly independent set of $p$ eigenvectors (one for each eigenvalue). Since these vectors belong to an $n$-dimensional vector space, $p$ cannot exceed $n$.

25. Let $x$ be a nonzero vector such that $Ax = \lambda x$. Then $A^{-1}Ax = A^{-1}(\lambda x)$, and $x = \lambda(A^{-1}x)$. Since $x \neq 0$ (and since $A$ is invertible), $\lambda$ cannot be zero. Then $\lambda^{-1}x = A^{-1}x$, which shows that $\lambda^{-1}$ is an eigenvalue of $A^{-1}$, because $x \neq 0$.

Note: The relation between the eigenvalues of $A$ and $A^{-1}$ is important in the so-called inverse power method for estimating an eigenvalue of a matrix. (See Section 5.8.)

27. For any $\lambda$, $(A - \lambda I)^T = A^T - (\lambda I)^T = A^T - \lambda I$. Since $(A - \lambda I)^T$ is invertible if and only if $A - \lambda I$ is invertible (by Theorem 6(c) in Section 2.2), we conclude that $A^T - \lambda I$ is not invertible if and only if $A - \lambda I$ is not invertible. That is, $\lambda$ is an eigenvalue of $A^T$ if and only if $\lambda$ is an eigenvalue of $A$.

33. The solution is given in the text. This exercise is important because it sets the stage for work later in the chapter. You ought to spend at least a little time on Exercise 34, too, even if that is not assigned.

Answer to Checkpoint: The answer in the text is also correct because it is a nonzero multiple of the eigenvector found in the solution to Exercise 7, and any nonzero multiple of an eigenvector is another eigenvector. (The eigenspace is a subspace and so is closed under scalar multiplication.)

MATLAB Finding Eigenvectors

When you know an eigenvalue, MATLAB can produce a basis for the corresponding eigenspace. For example, if $A$ is a $5 \times 5$ matrix with an eigenvalue 7, then the commands

```
C = A - 7*eye(5)
B = nulbasis(C)
```
or, simply, $B = \text{nulbasis}(A - 7\text{eye}(5))$, produce a matrix $B$ whose columns form a basis for the eigenspace for $A$ corresponding to $\lambda = 7$. In general, $\text{eye}(k)$ is the $k \times k$ identity matrix, and $\text{nulbasis}(C)$ is a matrix whose columns form a basis for $\text{Nul} \ C$ (the same basis you would get if you started with $\text{ref}(C)$ and made the calculations by hand). The command $\text{nulbasis}$ is in the Laydata5 Toolbox.
If the numbers in the basis matrix $B$ are messy, use `format rat; B`, which will display the entries in $B$ as rational numbers. (All eigenvectors calculated in this section have rational entries, so the rational format introduces no error.) To return to the usual decimal number display, enter `format short`.

For Exercises 37–40, you need the command `eig(A)`, which lists the eigenvalues of $A$ accurately to many decimal places, in most cases. For example, enter

$$
ev = \text{eig}(A)$$
$$B = \text{nulbasis}(A - \ev(2) \cdot \text{eye}(5))$$

To compute a basis for the eigenspace corresponding to the second eigenvalue listed in the vector $\ev$. It is dangerous to type `eig(A)` and simply “look” at the list of eigenvalues to use in the `nulbasis` command. You may make a mistake when you type an eigenvalue, particularly when the eigenvalue has sixteen nonzero digits and MATLAB displays only six of them.

---

5.2 **THE CHARACTERISTIC EQUATION**

There are several equivalent definitions of the determinant of a matrix. The definition here in terms of the pivots in an echelon form has the advantage that it is easy to state and understand, and in most cases it provides the most efficient way to compute a determinant.

---

**KEY IDEAS**

When $A$ is $3 \times 3$, the geometric interpretation of $\det A$ as a volume explains why $\det A = 0$ if and only if $A$ is not invertible:

- The determinant of $A$ is zero.
- $\iff$ The parallelepiped determined by the columns of $A$ has zero volume.
- $\iff$ One column of $A$ is in the subspace spanned by the other columns.
- $\iff$ The columns of $A$ are linearly dependent.
- $\iff$ The matrix $A$ is not invertible.

If $A$ is $n \times n$, then $\det(A - \lambda I) = 0$ if and only if $A - \lambda I$ is not invertible, and this happens if and only if $\lambda$ is an eigenvalue of $A$.

Exercises 1–14 are designed only to provide some basic familiarity with characteristic polynomials. The main use of $\det(A - \lambda I)$ is as a tool for studying eigenvalues rather than computing them.

Sometimes the characteristic polynomial is defined as $\det(\lambda I - A)$. A property of determinants implies that $\det (\lambda I - A) = (-1)^n \det (A - \lambda I)$, when $A$ is $n \times n$, so the two polynomials are either the same (when $n$ is even) or they are negatives of one another. The use of $\det (A - \lambda I)$ tends to make hand calculations easier and less prone to copying errors.
SOLUTIONS TO EXERCISES

1. \( A = \begin{bmatrix} 2 & 7 \\ 7 & 2 \end{bmatrix} \), \( A - \lambda I = \begin{bmatrix} 2 & 7 \\ 7 & 2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 2-\lambda & 7 \\ 7 & 2-\lambda \end{bmatrix} \), the characteristic polynomial is
\[
\det(A - \lambda I) = (2-\lambda)^2 - 7^2 = 4 - 4\lambda + \lambda^2 - 49 = \lambda^2 - 4\lambda - 45
\]
In factored form, the characteristic equation is \((\lambda - 9)(\lambda + 5) = 0\), so the eigenvalues of \( A \) are 9 and -5.

Warning: Don’t row reduce a matrix \( A \) to find its eigenvalues. Row reduction preserves the null space of \( A \) but not the eigenvalues of \( A \).

7. \( A = \begin{bmatrix} 5 & 3 \\ -4 & 4 \end{bmatrix} \), \( A - \lambda I = \begin{bmatrix} 5-\lambda & 3 \\ -4 & 4-\lambda \end{bmatrix} \), the characteristic polynomial is
\[
\det(A - \lambda I) = (5-\lambda)(4-\lambda) - (3)(-4) = 20 - 9\lambda + \lambda^2 + 12 = \lambda^2 - 9\lambda + 32
\]
The characteristic polynomial does not factor easily, but the quadratic formula provides the solutions of \( \lambda^2 - 9\lambda + 32 = 0 \).
\[
\lambda = \frac{-9 \pm \sqrt{81 - 4(32)}}{2}
\]
These values for \( \lambda \) are not real numbers, so \( A \) has no real eigenvalues. There is no nonzero vector \( x \) in \( \mathbb{R}^2 \) such that \( Ax = \lambda x \) for such a \( \lambda \). (For any \( x \neq 0 \) in \( \mathbb{R}^2 \), the vector \( Ax \) has only real entries and thus could not equal a complex multiple of \( x \).)

Study Tip: If you are asked to work some of Exercises 9–14, you may be tested on them. This is one way of finding out if you know what the characteristic polynomial is and how it is connected with eigenvalues. Also, you can show that you know some elementary properties of determinants.

13. \( A - \lambda I = \begin{bmatrix} 6-\lambda & -2 & 0 \\ -2 & 9-\lambda & 0 \\ 5 & 8 & 3-\lambda \end{bmatrix} \)

The method using the special 3 \( \times \) 3 formula will produce the characteristic polynomial 
\(-\lambda^3 + 18\lambda^2 - 95\lambda + 150\). Factoring such a polynomial to find the eigenvalues requires a little experience. (See the appendix at the end of the exercise solutions.). However, if you use a
cofactor expansion down the third column (see Section 3.1), you immediately obtain

\[
\det(A - \lambda I) = (3 - \lambda) \cdot \det \begin{bmatrix} 6 - \lambda & -2 \\ -2 & 9 - \lambda \end{bmatrix} = (3 - \lambda)[(6 - \lambda)(9 - \lambda) - 4] = (3 - \lambda)(\lambda^2 - 15\lambda + 50)
\]

The characteristic polynomial is already partially factored, and the remaining quadratic factor is itself easily factored. The factored characteristic polynomial is \((3 - \lambda)(\lambda - 10)(\lambda - 5)\) or, equivalently, \(-(\lambda - 3)(\lambda - 5)(\lambda - 10)\).

**Note:** The solutions of Exercises 11–14 are similar to that of Exercise 13. These matrices have the property that if a cofactor expansion is chosen along a column or row that contains two zeros, then the characteristic polynomial appears in a partially factored form.

19. Since the equation \(\det(A - \lambda I) = (\lambda_i - \lambda)(\lambda_j - \lambda) \cdots (\lambda_n - \lambda)\) holds for all \(\lambda\), set \(\lambda = 0\) and conclude that \(\det A = \lambda_1\lambda_2 \cdots \lambda_n\).

21. a. See Example 1. 
   b. See Theorem 3. 
   c. See Theorem 3. 
   d. See the solution of Example 4.

23. If \(A = QR\), with \(Q\) invertible, and if \(A_1 = RQ\), then \(A_1 = Q^{-1}QR = Q^{-1}AQ\), which shows that \(A_1\) is similar to \(A\).

25. \(A = \begin{bmatrix} .6 & .3 \\ .4 & .7 \end{bmatrix}\), \(v_1 = \begin{bmatrix} 3/7 \\ 4/7 \end{bmatrix}\), \(x_0 = [.5]

   a. The problem statement implies that \(v_1\) is an eigenvector of \(A\). This is readily verified: 

   \[Av_1 = \begin{bmatrix} .6 & .3 \\ .4 & .7 \end{bmatrix} \begin{bmatrix} 3/7 \\ 4/7 \end{bmatrix} = \begin{bmatrix} (1.8 + 1.2)/7 \\ (1.2 + 2.8)/7 \end{bmatrix} = \begin{bmatrix} 3/7 \\ 4/7 \end{bmatrix} = 1 \cdot v_1\]

   So \(v_1\) is an eigenvector corresponding to the eigenvalue 1. To find another eigenvector, first compute the characteristic polynomial:

   \[
   \det(A - \lambda I) = \det \begin{bmatrix} .6 - \lambda & .3 \\ .4 & .7 - \lambda \end{bmatrix} = (.6 - \lambda)(.7 - \lambda) - .12 = .42 - 1.3\lambda + \lambda^2 - .12 = \lambda^2 - 1.3\lambda + .3
   \]

   Factoring might be a little difficult, but since 1 is an eigenvalue \(\lambda - 1\) must be a factor of the polynomial. This helps to see that \(\det(A - \lambda I) = (\lambda - 1)(\lambda - .3)\). So .3 is the other eigenvalue. For a corresponding eigenvector, solve \((A - .3I)x = 0\):

   \[
   [(A - .3I) \ 0] = \begin{bmatrix} .3 & .3 & 0 \\ .4 & .4 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad x_1 + x_2 = 0
   \]
5-8   CHAPTER 5   •   Eigenvalues and Eigenvectors

So \( x_1 = -x_2 \), with \( x_2 \) free. An eigenvector is \( v_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \). Now, \( \{v_1, v_2\} \) is linearly independent because the eigenvectors correspond to different eigenvalues (also, they are not multiples). Thus, \( \{v_1, v_2\} \) is basis for \( \mathbb{R}^2 \), by the Basis Theorem, because the set contains two vectors and \( \mathbb{R}^2 \) is two-dimensional. (You can use other arguments, too.)

b. To show that \( x_0 = v_1 + cv_2 \), just compare \( x_0 - v_1 \) with \( v_2 \), to find \( c \):

\[
\begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix} - \begin{bmatrix} 3/7 \\ 4/7 \end{bmatrix} = \begin{bmatrix} 1/14 \\ -1/14 \end{bmatrix} = \frac{1}{14} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \frac{1}{14} v_2
\]

So \( x_0 = v_1 + (1/14)v_2 \).

c. Compute \( x_1 = Ax_0 = A(v_1 + (1/14)v_2) = Av_1 + (1/14)Av_2 = v_1 + (1/14)(.3)v_2 \), because \( v_1 \) and \( v_2 \) are eigenvectors of \( A \) corresponding to eigenvalues 1 and .3, respectively. To continue, recall from Practice Problem 2 in Section 5.1 that \( A^k x = \lambda^k x \) when \( x \) is an eigenvector corresponding to an eigenvalue \( \lambda \). Thus,

\[
\begin{align*}
x_2 &= A^2 x_0 = A^2(v_1 + (1/14)v_2) = A^2 v_1 + (1/14)A^2 v_2 = v_1 + (1/14)(.3)^2 v_2 \\
x_3 &= A^3 x_0 = A^3(v_1 + (1/14)v_2) = A^3 v_1 + (1/14)A^3 v_2 = v_1 + (1/14)(.3)^3 v_2
\end{align*}
\]

Note: Another way to find these formulas is to compute

\[
\begin{align*}
x_2 &= A^2 x_0 = A(Ax_0) = A(v_1 + (1/14)(.3)v_2) = Av_1 + (1/14)(.3)Av_2 = v_1 + (1/14)(.3)^2 v_2 \\
x_3 &= A^3 x_0 = A(A^2 x_0) = A(v_1 + (1/14)(.3)^2 v_2) = Av_1 + (1/14)(.3)^3 Av_2 \\
&= v_1 + (1/14)(.3)^3 v_2
\end{align*}
\]

By inspection, it seems that

\[
x_k = v_1 + (1/14)(.3)^k v_2
\]

As \( k \to \infty \), the powers of .3 tend to 0, and \( x_k \) tends to \( v_1 \).

\[
\text{Appendix: Factoring a Polynomial}
\]

In general it is difficult to factor a polynomial of degree 3 or higher (unless you have one of several powerful computer programs available). Fortunately, textbook examples and exercises tend to have small integer solutions. The following observation is helpful.

Let \( p(\lambda) \) be a polynomial with integer coefficients. If \( p(c) = 0 \) for some integer \( c \), then \( \lambda - c \) is a factor of \( p(\lambda) \) and \( c \) is a divisor of the constant term of \( p(\lambda) \).

**EXAMPLE**

Find the eigenvalues of the matrix \( A \) whose characteristic polynomial is \( p(\lambda) = -\lambda^3 + 18\lambda^2 - 96\lambda + 160 \).

**Solution**

By the observation above, any integer eigenvalue of \( A \) must be a divisor of the constant term 160 in the characteristic polynomial. There are twenty-four such divisors: 1, 2, 4, 5, 8, 10, 16, 20, 32, 40, 80, and 160, together with the negatives of these numbers. We let \( c \) be one of these numbers and try to divide \( \lambda - c \) into the polynomial by long division. The terms \( \lambda + 1 \) and \( \lambda - 2 \) don’t work, and the first successful division is...

Copyright © 2016 Pearson Education, Inc.
The characteristic polynomial is \((\lambda - 4)(\lambda^2 + 14\lambda - 40)\). The quadratic polynomial factors easily, and the characteristic equation is \((\lambda - 4)(\lambda - 4)(10 - \lambda) = 0\). The eigenvalues of \(A\) are 4 (with multiplicity two) and 10.

**MATLAB poly, plot**

You can use the MATLAB command `poly(A)` to check your answers in Exercises 9–14. Note that if \(A\) is \(n \times n\), this command lists the coefficients of the characteristic polynomial of \(A\) in order of decreasing powers of \(\lambda\), beginning with \(\lambda^n\). If the polynomial is of odd degree, the coefficients are multiplied by \(-1\), to make +1 the coefficient of \(\lambda^n\). This corresponds to using \(\det(\lambda I - A)\) instead of \(\det(A - \lambda I)\).

For Exercises 28 and 29, use `randomint(4)` to create a \(4 \times 4\) matrix with random integer entries. For Exercise 29, use `gauss` and perhaps `swap` to create the echelon form without row scaling. See the MATLAB box for Section 1.4.

The following commands will produce the graph of the characteristic polynomial of the matrix \(A\) in Exercise 30 (with \(a = 32\)).

```matlab
x = linspace(0,3); Choose 100 points between 0 and 3
grid on Include a grid on the display
hold on Add the next graph to the display
A(3,2)= 32; p = poly(A); Compute the characteristic polynomial
v = polyval(p,x); Evaluate it at the points in \(x\)
plot(x,v,’b’) Plot the graph in blue.
```

Edit line 4 to change the value of \(a\) (from 32 to another value). Edit line 6 to change the color of the graph. When the commands are run again, the old graph(s) will remain visible. If you do not specify the color of the graph, MATLAB will automatically cycle through a set of colors, one for each graph on the display. To create a fresh display, enter `hold off`.

To take advantage of MATLAB’s interactive plotting features, see the topic “Plotting Tools” in MATLAB’s `Help` menu.
The factorization $A = PDP^{-1}$ is used to compute powers of $A$, decouple dynamical systems in Sections 5.6 and 5.7, and study symmetric matrices and quadratic forms in Chapter 7.

### KEY IDEAS

Example 3 gives the algorithm for diagonalizing a matrix $A$. After you construct $P$ and $D$, check your calculations:

1. Compute $AP$ and $PD$, and check that $AP = PD$.
2. Make sure the columns of $P$ are linearly independent. Use Theorem 7 to save time. You only have to verify that for each eigenvalue, the corresponding eigenvectors are linearly independent. That’s easy if the dimension of the eigenspace is 2 or 1.

The key equation in this section is $AP = PD$. It will help you to keep the order of the factors correct when you write $A = PDP^{-1}$, and it also leads immediately to $P^{-1}AP = D$, which you may need occasionally. Possible test question: If $AP = PD$, explain why the first column of $P$ is an eigenvector of $A$ (if the column is nonzero). (Study the proof of the Diagonalization Theorem.)

**Warning:** Do not confuse the property of being diagonalizable with the property of being invertible. They are not connected. The matrix in Example 5 is diagonalizable, but it is not invertible because 0 is an eigenvalue. The matrix $\begin{bmatrix} 1 & -2 \\ 0 & 1 \end{bmatrix}$ is invertible, but it is not diagonalizable because the eigenspace for $\lambda = 1$ is only one-dimensional.

### SOLUTIONS TO EXERCISES

1. $P = \begin{bmatrix} 5 & 7 \\ 2 & 3 \end{bmatrix}$, $D = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$, $A = PDP^{-1}$, and $A^4 = PD^4P^{-1}$. Next compute $P^{-1} = \begin{bmatrix} 3 & -7 \\ -2 & 5 \end{bmatrix}$, $D^4 = \begin{bmatrix} 16 & 0 \\ 0 & 1 \end{bmatrix}$. Putting this together,

   $A^4 = \begin{bmatrix} 5 & 7 \\ 2 & 3 \end{bmatrix} \begin{bmatrix} 16 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 3 & -7 \\ -2 & 5 \end{bmatrix} = \begin{bmatrix} 80 & 7 \\ 32 & 3 \end{bmatrix} \begin{bmatrix} 3 & -7 \\ -2 & 5 \end{bmatrix} = \begin{bmatrix} 226 & -525 \\ 90 & -209 \end{bmatrix}$

7. $A = \begin{bmatrix} 1 & 0 \\ 6 & -1 \end{bmatrix}$. The eigenvalues are obviously $\pm 1$ (since $A$ is triangular).
For $\lambda = 1$: $A - 1I = \begin{bmatrix} 0 & 0 \\ 6 & -2 \end{bmatrix}$. The equation $(A - I)x = 0$ amounts to $6x_1 - 2x_2 = 0$. So $x_1 = (1/3)x_2$, with $x_2$ free. The general solution is $x = \begin{bmatrix} 1/3 \\ 1 \end{bmatrix}$. A nice basis vector for the eigenspace is $u = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$.

For $\lambda = -1$: $A - (-1)I = \begin{bmatrix} 2 & 0 \\ 6 & 0 \end{bmatrix}$. The equation $(A + I)x = 0$ amounts to $2x_1 = 0$, with $x_2$ free. The general solution is $x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. Take $u = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ as a basis vector for the eigenspace.

From $u_1$ and $u_2$, construct $P = [u_1 \quad u_2] = \begin{bmatrix} 1 & 0 \\ 3 & 1 \end{bmatrix}$. Then set $D = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, where the eigenvalues in $D$ correspond to $u_1$ and $u_2$, respectively.

**Warning:** The $3 \times 3$ matrices in Exercises 12–18 may be diagonalizable even though each matrix has only two distinct eigenvalues. (Theorem 6 gives only a sufficient condition for diagonalizability.) You have to check for three linearly independent eigenvectors.

13. $A = \begin{bmatrix} 2 & 2 & -1 \\ 1 & 3 & -1 \\ -1 & -2 & 2 \end{bmatrix}$. The eigenvalues 5 and 1 are given. Because $A$ is $3 \times 3$, you need three linearly independent eigenvectors.

For $\lambda = 5$: Solve $(A - 5I)x = 0$. Form $A - 5I = \begin{bmatrix} -3 & 2 & -1 \\ 1 & -2 & -1 \\ -1 & -2 & -3 \end{bmatrix}$ and compute

$\begin{bmatrix} -3 & 2 & -1 \\ 1 & -2 & -1 \\ -1 & -2 & -3 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & -2 & -1 \\ 1 & -2 & -1 \\ -1 & -2 & -3 \end{bmatrix} \sim \cdots \sim \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

So $x_1 = -x_3$, $x_2 = -x_3$, with $x_3$ free. Take $v = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, for instance, as a basis vector for the eigenspace. At this point, you don’t know if $A$ is diagonalizable. Your only hope is to find two linearly independent eigenvectors inside the eigenspace for $\lambda = 1$, because there are no other eigenspaces in which to look.
For \( \lambda = 1 \): Form \( A - I = \begin{bmatrix} 1 & 2 & -1 \\ 1 & 2 & -1 \\ -1 & -2 & 1 \end{bmatrix} \). The equation \((A - I)x = 0\) reduces to \(x_1 + 2x_2 - x_3 = 0\).

So \(x_1 = -2x_2 + x_3\), with \(x_2\) and \(x_3\) free. At this point you know that the eigenspace is two-dimensional (because there are two free variables). So there are the necessary two linearly independent eigenvectors, and hence \(A\) is diagonalizable. To produce the eigenvectors, write the solution of \((A - I)x = 0\) in the form

\[
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} x_2 \\ 1 + x_3 \\ 0 \end{bmatrix} = x_2 \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} + x_3 \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \]

Set \(v_2 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}\), \(v_1 = \begin{bmatrix} -1 \\ -2 \\ 1 \end{bmatrix}\), and \(P = [v_1 \ v_2 \ v_3] = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}\).

The columns of \(P\) are linearly independent, by Theorem 7, because the eigenvectors form bases for their respective eigenspaces. So \(P\) is invertible. Since the first column of \(P\) corresponds to \(\lambda = 5\), the first diagonal entry in \(D\) must be 5, which means that

\[
D = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\]

Warning: A common mistake in Exercise 13 is to build a \(3 \times 3\) matrix \(P\) in the usual way and then take \(D\) to be a \(2 \times 2\) matrix, such as \(D = \begin{bmatrix} 5 & 0 \\ 0 & 1 \end{bmatrix}\).

Of course this doesn’t make sense because \(PD\) isn’t even defined. Another mistake is to make a \(2 \times 2\) matrix \(D\) as above and build \(P\) with only two of the three eigenvectors, say one from each eigenspace. Now \(AP = PD\), but \(P\) is a \(3 \times 2\) matrix and is not invertible.

\[
19. \quad A = \begin{bmatrix} 5 & -3 & 0 & 9 \\ 0 & 3 & 1 & -2 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \text{. The eigenvalues are obviously 5, 3, and 2. (Why?)}
\]

For \(\lambda = 2\): To solve \((A - 2I)x = 0\), completely reduce \([(A - 2I) \quad 0]\):

\[
\begin{bmatrix} 3 & -3 & 0 & 9 & 0 \\ 0 & 1 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \sim \begin{bmatrix} 3 & 0 & 3 & 3 & 0 \\ 0 & 1 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}
\]

Copyright © 2016 Pearson Education, Inc.
So \( x_1 = -x_3 - x_4, \ x_2 = -x_3 + 2x_4 \), with \( x_3 \) and \( x_4 \) free. The usual calculations produce a basis for the eigenspace:

\[
\begin{bmatrix}
 x_1 \\
 x_2 \\
 x_3 \\
 x_4
\end{bmatrix} = \begin{bmatrix}
 -x_3 - x_4 \\
 -x_3 + 2x_4 \\
 x_3 \\
 x_4
\end{bmatrix} = x_3 \begin{bmatrix}
 -1 \\
 -1 \\
 1 \\
 0
\end{bmatrix} + x_4 \begin{bmatrix}
 -1 \\
 2 \\
 0 \\
 1
\end{bmatrix}.
\]

Basis: \( \mathbf{v}_1 = \begin{bmatrix} -1 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \ \mathbf{v}_2 = \begin{bmatrix} -1 \\ 2 \\ 0 \\ 0 \end{bmatrix} \).

**Checkpoint:** If you happened to choose \( \lambda = 2 \) first, as in this solution, would you have enough information at this point to determine whether \( A \) is diagonalizable?

For \( \lambda = 3 \): To solve \((A - 3I)x = 0\), completely reduce \([(A - 3I) \ 0]\):

\[
\begin{bmatrix}
 2 & -3 & 0 & 9 & 0 \\
 0 & 0 & 1 & -2 & 0 \\
 0 & 0 & -1 & 0 & 0 \\
 0 & 0 & -3 & 0 & 0
\end{bmatrix} \sim \begin{bmatrix}
 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0
\end{bmatrix},
\]

Choosing \( x_2 = 2 \) produces the eigenvector \( \mathbf{v}_3 = (3, 2, 0, 0) \).

For \( \lambda = 5 \): To solve \((A - 5I)x = 0\), completely reduce \([(A - 5I) \ 0]\):

\[
\begin{bmatrix}
 0 & -3 & 0 & 9 & 0 \\
 0 & -2 & 1 & -2 & 0 \\
 0 & 0 & -3 & 0 & 0 \\
 0 & 0 & 0 & -3 & 0
\end{bmatrix} \sim \begin{bmatrix}
 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0
\end{bmatrix},
\]

A basis vector for the eigenspace is \( \mathbf{v}_4 = (1, 0, 0, 0) \). Set

\[
P = [\mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3 \ \mathbf{v}_4] = \begin{bmatrix}
 -1 & -1 & 3 & 1 \\
 -1 & 2 & 2 & 0 \\
 1 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0
\end{bmatrix}, \ \ D = \begin{bmatrix}
 2 & 0 & 0 & 0 \\
 0 & 2 & 0 & 0 \\
 0 & 0 & 3 & 0 \\
 0 & 0 & 0 & 5
\end{bmatrix}
\]

This answer differs from that in the text. There, \( P = [\mathbf{v}_4 \ \mathbf{v}_3 \ \mathbf{v}_1 \ \mathbf{v}_2] \), and the entries in \( D \) are rearranged to match the new order of the eigenvectors. According to the Diagonalization Theorem, both answers are correct.

21. a. The symbol \( D \) does not automatically denote a diagonal matrix.

   b. See the remark after the statement of the Diagonalization Theorem.

   c. Check out the matrix in Example 4.

   d. See Example 4.

23. \( A \) is diagonalizable because you know that five linearly independent eigenvectors exist: three in the three-dimensional eigenspace and two in the two-dimensional eigenspace. Theorem 7 guarantees that the set of all five eigenvectors is linearly independent.
25. Let \( \{v_1\} \) be a basis for the one-dimensional eigenspace, let \( v_2 \) and \( v_3 \) form a basis for the two-dimensional eigenspace, and let \( v_4 \) be any eigenvector in the remaining eigenspace. By Theorem 7, \( \{v_1, v_2, v_3, v_4\} \) is linearly independent. Since \( A \) is \( 4 \times 4 \), the Diagonalization Theorem shows that \( A \) is diagonalizable.

27. If \( A \) is diagonalizable, then \( A = PD P^{-1} \) for some invertible \( P \) and diagonal \( D \). Since \( A \) is invertible, 0 is not an eigenvalue of \( A \). So the diagonal entries in \( D \) (which are eigenvalues of \( A \)) are not zero, and \( D \) is invertible. By the theorem on the inverse of a product, \( A^{-1} = (PD P^{-1})^{-1} = (P^{-1})^{-1} D^{-1} P^{-1} = PD^{-1}P^{-1} \). Since \( D^{-1} \) is obviously diagonal, \( A^{-1} \) is diagonalizable.

**A Second Proof:** If \( A \) is \( n \times n \), it has \( n \) linearly independent eigenvectors, say, \( v_1, \ldots, v_n \). Then \( v_1, \ldots, v_n \) are also eigenvectors of \( A^{-1} \), by the solution of Exercise 25 in Section 5.1. Hence \( A^{-1} \) is diagonalizable, by the Diagonalization Theorem.

29. The diagonal entries in \( D_1 \) are reversed from those in \( D \). So interchange the (eigenvector) columns of \( P \) to make them correspond properly to the eigenvalues in \( D_1 \). In this case,

\[
P = \begin{bmatrix} 1 & 1 \\ -2 & -1 \end{bmatrix} \quad D_1 = \begin{bmatrix} 3 & 0 \\ 0 & 5 \end{bmatrix}
\]

Although the first column of \( P \) must be an eigenvector corresponding to the eigenvalue 3, there is nothing to prevent us from selecting some multiple of \( \begin{bmatrix} 1 \\ -2 \end{bmatrix} \), say \( \begin{bmatrix} -3 \\ 6 \end{bmatrix} \), and letting \( P_1 = \begin{bmatrix} -3 & 1 \\ 6 & -1 \end{bmatrix} \). We now have three different factorizations or “diagonalizations” of \( A \): \( A = PD P^{-1} = P D_1 P^{-1} = P D_2 P^{-1} \).

**Answer to Checkpoint:** Yes. In Exercise 19, the fact that the eigenspace for \( \lambda = 2 \) is two-dimensional guarantees that \( A \) is diagonalizable, because each of the other two eigenvalues will produce at least one eigenvector, and the resulting set of four eigenvectors will be linearly independent, by Theorem 7. So \( A \) is diagonalizable, by the Diagonalization Theorem. (Note that since one eigenspace is two-dimensional, the other eigenspaces must be one-dimensional, because there could not possibly be \textit{more} than four linearly independent eigenvectors in \( \mathbb{R}^4 \).)

---

**Mastering Linear Algebra Concepts: Eigenvalue, Eigenvector, Eigenspace**

I suggest that you take time now to prepare review sheets for the three terms listed above. Begin with their definitions, of course.

**Eigenvalue:**
- equivalent description Sec. 5.1: Equation (3); Box on page 276
- geometric interpretation Sec. 5.1: Fig. 2
- special cases Theorems 1, 6
- typical computations Sec. 5.1: Exer. 7, 19; Sec. 5.2: Exer. 7, 15
- connections with other concepts The IMT; Theorem 4; Sec. 5.2: Exer. 19
MATLAB  Diagonalization

To practice the diagonalization procedure in this section, you should use nulbasis to produce eigenvectors. For Exercises 33–36, first enter the command ev = eig(A), to provide the eigenvalues. See the MATLAB box for Section 5.1.

In later work, you may automate the diagonalization process. The command [P D] = eig(A) produces two square matrices P and D (or any other names you choose) such that AP = PD, with D diagonal. If P is invertible, then A is diagonalizable. Check whether P*D*inv(P) - A is the zero matrix. In any case, P is likely to be different from what you construct for the homework. The columns of P are scaled so they are “unit” vectors (studied in Chapter 6). Dividing P by one of its entries may make some of the entries recognizable, because the problems in the text usually involve simple numbers.

5.4  EIGENVALUES AND LINEAR TRANSFORMATIONS

This section introduces the matrix of a linear transformation relative to specified bases for vector spaces, and then uses this concept to give a new interpretation of the matrix factorization A = PDP⁻¹.

STUDY NOTES

The exercises will help you learn the definition of the matrix representation of a transformation relative to specified bases, say B and C. Compare this definition with the standard matrix of a transformation from \( \mathbb{R}^n \) into \( \mathbb{R}^m \). What are B and C in this case?

After Example 1 (and Exercises 1–6 and 9–10), the section focuses on the case when \( T:V \rightarrow V \) and \( B = C \). The following algorithm and the solution to Exercise 7 describe two different ways to construct \( [T]_B \).
Algorithm for Finding the $\mathcal{B}$-matrix of $T : V \to V$

1. Compute the images of the basis vectors:
   \[ T(b_1), \ldots, T(b_n) \]
2. Convert these images into $\mathcal{B}$-coordinate vectors:
   \[ [T(b_1)]_\mathcal{B}, \ldots, [T(b_n)]_\mathcal{B} \]
3. Place the $\mathcal{B}$-coordinate vectors into the columns of $[T]_\mathcal{B}$.

The sentence before Theorem 8 summarizes the main idea of the theorem. Studying the proof should help you understand the theorem and review important concepts. The subsection on Similarity of Matrix Representations could have contained more ideas. Take notes carefully if your instructor decides to expand this material somewhat.

**SOLUTIONS TO EXERCISES**

1. \[ T(b_1) = 3d_1 - 5d_2, \quad T(b_2) = -d_1 + 6d_2, \quad T(b_3) = 4d_2 \]

   \[ [T(b_1)]_T = \begin{bmatrix} 3 \\ -5 \end{bmatrix}, \quad [T(b_2)]_T = \begin{bmatrix} -1 \\ 6 \end{bmatrix}, \quad [T(b_3)]_T = \begin{bmatrix} 0 \\ 4 \end{bmatrix} \]

   Matrix for $T$ relative to $\mathcal{B}$ and $\mathcal{D}$:
   \[ \begin{bmatrix} 3 & -1 & 0 \\ -5 & 6 & 4 \end{bmatrix} \]

7. The method of Example 2 works. But, perhaps a faster way is to realize that the information given provides the general form of $T(p)$ as shown in the figure below:

\[ a_0 + a_1 t + a_2 t^2 \]  \[ \rightarrow 3a_0 + (5a_0 - 2a_1) t + (4a_1 + a_2) t^2 \]

The matrix that implements the multiplication along the bottom of the figure is easily filled in by inspection:
\[
\begin{bmatrix}
? & ? & ? \\
? & ? & ? \\
? & ? & ? \\
\end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
a_2 \\
\end{bmatrix}
= \begin{bmatrix}
3a_0 \\
5a_0 - 2a_1 \\
4a_1 + a_2 \\
\end{bmatrix}
\] implies that \([T]_B = \begin{bmatrix}
3 & 0 & 0 \\
5 & -2 & 0 \\
0 & 4 & 1 \\
\end{bmatrix}\]

**Study Tip:** See the Study Guide notes for Section 1.9 (particularly for Exercise 19). This method allows you to find a matrix that implements the mapping without assuming that \(T\) is linear. In fact, this derivation proves that \(T\) is linear, because it is now represented as a matrix transformation. Why not try this method on Example 2?

9. **b.** The transformation \(T\) maps \(\mathbb{P}_2\) into \(\mathbb{R}^3\) by the formula \(T(p) = \begin{bmatrix} p(-1) \\ p(0) \\ p(1) \end{bmatrix}\). Take any \(p\) and \(q\) in \(\mathbb{P}_2\) and any scalar \(c\). Then \(T\) is linear because

\[
T(p + q) = \begin{bmatrix} (p + q)(-1) \\ (p + q)(0) \\ (p + q)(1) \end{bmatrix} = \begin{bmatrix} p(-1) + q(-1) \\ p(0) + q(0) \\ p(1) + q(1) \end{bmatrix} = T(p) + T(q), \quad \text{and}
\]

\[
T(cp) = \begin{bmatrix} (cp)(-1) \\ (cp)(0) \\ (cp)(1) \end{bmatrix} = c \begin{bmatrix} p(-1) \\ p(0) \\ p(1) \end{bmatrix} = cT(p)
\]

13. Start by diagonalizing \(A = \begin{bmatrix} 0 & 1 \\ -3 & 4 \end{bmatrix}\). The characteristic polynomial is

\[
\lambda^2 - 4\lambda + 3 = (\lambda - 1)(\lambda - 3), \quad \text{so the eigenvalues are 1 and 3.}
\]

**For \(\lambda = 1\):** \(A - I = \begin{bmatrix} -1 & 1 \\ -3 & 3 \end{bmatrix}\). The equation \((A - I)x = 0\) amounts to \(-x_1 + x_2 = 0\). So \(x_1 = x_2\), with \(x_2\) free. As a basis vector, take \(u_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\).

**For \(\lambda = 3\):** \(A - 3I = \begin{bmatrix} -3 & 1 \\ -3 & 1 \end{bmatrix}\). The equation \((A - I)x = 0\) amounts to \(-3x_1 + x_2 = 0\). So \(x_1 = x_2/3\), with \(x_2\) free. Take \(u_2 = \begin{bmatrix} 1 \\ 3 \end{bmatrix}\) as a basis vector. The vectors \(u_1, u_2\) can form the columns of a matrix \(P\) that diagonalizes \(A\). By Theorem 8, the basis \(B = \{u_1, u_2\}\) has the property that the \(B\)-matrix of the transformation \(x \mapsto Ax\) is a diagonal matrix.

19. If \(A\) is similar to \(B\), then there exists an invertible matrix \(P\) such that \(P^{-1}AP = B\). Thus \(B\) is invertible because it is the product of invertible matrices. By a theorem about inverses of products, \(B^{-1} = P^{-1}A^{-1}(P^{-1})^{-1} = P^{-1}A^{-1}P\), which shows that \(A^{-1}\) is similar to \(B^{-1}\).
21. By hypothesis, there exist invertible $P$ and $Q$ such that $P^{-1}BP = A$ and $Q^{-1}CQ = A$. Then $P^{-1}BP = Q^{-1}CQ$. Left-multiply by $Q$ and right-multiply by $Q^{-1}$ to obtain $QP^{-1}BPQ^{-1} = QQ^{-1}CQQ^{-1}$. So $C = QP^{-1}BPQ^{-1} = (PQ^{-1})^{-1}B(PQ^{-1})$, which shows that $B$ is similar to $C$.

23. If $Ax = \lambda x$, $x \neq 0$, then $P^{-1}Ax = \lambda P^{-1}x$. If $B = P^{-1}AP$, then

$$B(P^{-1}x) = P^{-1}AP(P^{-1}x) = P^{-1}Ax = \lambda P^{-1}x$$

by the first calculation. Note that $P^{-1}x \neq 0$, because $x \neq 0$ and $P^{-1}$ is invertible. Hence (*) shows that $P^{-1}x$ is an eigenvector of $B$ corresponding to $\lambda$. (Of course, $\lambda$ is an eigenvalue of both $A$ and $B$ because the matrices are similar, by Theorem 4 in Section 5.2.)

25. If $A = PBP^{-1}$, then

$$\text{tr}(A) = \text{tr}((PB)P^{-1}) = \text{tr}(P^{-1}(PB))$$

By the trace property

$$= \text{tr}(P^{-1}PB) = \text{tr}(IB) = \text{tr}(B)$$

If $B$ is diagonal, then the diagonal entries of $B$ must be the eigenvalues of $A$, by the Diagonalization Theorem (Theorem 5 in Section 5.3). So $\text{tr} A = \text{tr} B = \{\text{sum of the eigenvalues of } A\}$.

Study Tip: It can be shown that for any square matrix $A$, the trace of $A$ is the sum of the eigenvalues of $A$, counted according to multiplicities. You can use this fact to provide a quick check on your eigenvalue calculations. The sum of the eigenvalues must match the sum of the diagonal entries in $A$ (even if $A$ is not diagonalizable).

29. If $B = \{b_1, \ldots, b_n\}$, then the $B$-coordinate vector of $b_j$ is $e_j$, the standard basis vector for $\mathbb{R}^n$. For instance,

$$b_j = 1 \cdot b_j + 0 \cdot b_2 + \cdots + 0 \cdot b_n$$

Thus $[I(b_j)]_B = [b_j]_B = e_j$, and

$$[I]_B = \begin{bmatrix} [I(b_1)]_B & \cdots & [I(b_n)]_B \end{bmatrix} = \begin{bmatrix} e_1 & \cdots & e_n \end{bmatrix} = I$$

31. See the subsection, “Similarity of Matrix Representations,” which points out that the matrix $D$ in Theorem 8 need not be diagonal. If $P$ is the matrix whose columns come from $B$, then the $B$-matrix of the transformation $x \mapsto Ax$ is $D = P^{-1}AP$. From the data in the text,

$$A = \begin{bmatrix} -7 & -48 & -16 \\ 1 & 14 & 6 \\ -3 & -45 & -19 \end{bmatrix}, \quad P = \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 \\ -3 & -3 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} -7 & -48 & -16 \\ -3 & -2 & 3 \\ -3 & -3 & 0 \end{bmatrix}$$

Copyright © 2016 Pearson Education, Inc.
If the characteristic equation of an $n \times n$ real matrix $A$ has a complex eigenvalue $\lambda$ and if $\mathbf{v}$ is a nonzero vector in $\mathbb{C}^n$ such that $A\mathbf{v} = \lambda\mathbf{v}$, then both $\lambda$ and $\mathbf{v}$ provide useful information about $A$.

**STUDY NOTES**

Only matrices with real entries are considered here. If $\lambda$ is a complex eigenvalue of $A$, with $\mathbf{v}$ a corresponding eigenvector, then $\bar{\lambda}$ is also an eigenvalue of $A$, with $\bar{\mathbf{v}}$ an eigenvector. Find out if you should know how to prove this fact.

Example 6 describes the prototype for all $2 \times 2$ matrices with a complex eigenvalue $\lambda$. Only $\lambda$ is needed if you want to know the angle $\varphi$ of rotation and the scale factor $|\lambda|$, but an associated eigenvector $\mathbf{v}$ is also needed if you want to factor $A$ as $PCP^{-1}$, as in Example 7.

**SOLUTIONS TO EXERCISES**

1. $A = \begin{bmatrix} 1 & -2 \\ 1 & 3 \end{bmatrix}$, $A - \lambda I = \begin{bmatrix} 1-\lambda & -2 \\ 1 & 3-\lambda \end{bmatrix}$

   $\det(A - \lambda I) = (1 - \lambda)(3 - \lambda) - (-2) = \lambda^2 - 4\lambda + 5$

   Use the quadratic formula to find the eigenvalues: $\lambda = \frac{4 \pm \sqrt{16 - 20}}{2} = 2 \pm i$. Example 2 gives a shortcut for finding one eigenvector, and Example 5 shows how to write the other eigenvector with no effort.

   For $\lambda = 2 + i$: $A - (2 + i)I = \begin{bmatrix} -1-i & -2 \\ 1 & 1-i \end{bmatrix}$. The equation $(A - \lambda I)x = 0$ gives

   $(-1-i)x_1 - 2x_2 = 0$

   $x_1 + (1-i)x_2 = 0$

   As in Example 2, the two equations are equivalent—each determines the same relation between $x_1$ and $x_2$. So use the second equation to obtain $x_1 = -(1-i)x_2$, with $x_2$ free. The general solution is $x_2 \begin{bmatrix} -1+i \\ 1 \end{bmatrix}$, and the vector $\mathbf{v}_1 = \begin{bmatrix} -1+i \\ 1 \end{bmatrix}$ provides a basis for the eigenspace.

   For $\lambda = 2 - i$: Let $\mathbf{v}_2 = \mathbf{v}_1 = \begin{bmatrix} -1-i \\ 1 \end{bmatrix}$. The remark prior to Example 5 shows that $\mathbf{v}_2$ is automatically an eigenvector for $2 + i$. In fact, calculations similar to those above would show that $\{\mathbf{v}_2\}$ is a basis for the eigenspace. (In general, for a real matrix $A$, it can be shown that the set of complex conjugates of the vectors in a basis of the eigenspace for $\lambda$ is a basis of the eigenspace for $\bar{\lambda}$.)
7. \( A = \begin{bmatrix} \sqrt{3} & -1 \\ 1 & \sqrt{3} \end{bmatrix} \). The eigenvalues are \( \sqrt{3} \pm i \). Ask your instructor if you are permitted to write down the eigenvalues of \( \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \) from memory, or if you are expected to find them via the characteristic equation. Note that the eigenvectors are easy to remember, too. See the Practice Problem.

The scale factor associated with the transformation \( x \mapsto Ax \) is simply \( r = |\lambda| = \left( (\sqrt{3})^2 + 1^2 \right)^{1/2} = 2 \). For the angle of the rotation, plot the point \((a, b) = (\sqrt{3}, 1)\) in the \(xy\)-plane and use trigonometry.

![Diagram](image)

\( \phi = \pi/6 \) radians

13. From Exercise 1, \( \lambda = 2 \pm i \), and the eigenvector \( v = \begin{bmatrix} -1 - i \\ 1 \end{bmatrix} \) corresponds to \( \lambda = 2 - i \). Since \( \text{Re } v = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \) and \( \text{Im } v = \begin{bmatrix} -i \\ 0 \end{bmatrix} \), take \( P = \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix} \). Then compute

\[
C = P^{-1}AP = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} 1 & -2 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} -1 & -1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} -3 & -1 \\ 2 & -1 \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ 1 & 2 \end{bmatrix}
\]

Actually, Theorem 9 gives the formula for \( C \). Note that the eigenvector \( v \) corresponds to \( a - bi \) instead of \( a + bi \). If, for instance, you use the eigenvector for \( 2 + i \), your \( C \) will be \( \begin{bmatrix} 2 & 1 \\ -1 & 2 \end{bmatrix} \). The imaginary part of the eigenvalue is the \((1, 2)\)-entry in \( C \).

So, there are two possible choices for \( C \) (depending on the vector used to produce \( P \)). On an exam, if you are not sure of the form of \( C \), you can always compute it quickly from the formula \( C = P^{-1}AP \), as in the solution above.

Note: Because there are two possibilities for \( C \) in the factorization of a \( 2 \times 2 \) matrix as in Exercise 13, the measure of rotation \( \phi \) associated with the transformation \( x \mapsto Ax \) is determined only up to a change of sign. The “orientation” of the angle is determined by the change of variable \( x = Pu \). See Fig. 4 in the text.

19. \( A = \begin{bmatrix} 1.52 & -0.7 \\ 0.56 & 0.4 \end{bmatrix} \), \( \det(A - \lambda I) = \lambda^2 - 1.92\lambda + 1 \)

Use the quadratic formula to solve \( \lambda^2 - 1.92\lambda + 1 = 0 \):
\[ \lambda = \frac{1.92 \pm \sqrt{-0.3136}}{2} = 0.96 \pm 0.28i \]

To find the eigenvector for \( \lambda = 0.96 - 0.28i \), solve
\[
\begin{align*}
(1.52 - 0.96 + 0.28i)x_1 & = -0.7x_2 \\
0.56x_1 + (0.4 - 0.96 + 0.28i)x_2 & = 0
\end{align*}
\]

There is a nonzero solution (because \( 0.96 - 0.28i \) is an eigenvalue), so you can use either equation to find the solution. From the second equation,
\[
x_1 = \frac{0.56 - 0.28i}{0.56} x_2 = (1 - 0.5i)x_2
\]

Setting \( x_2 = 2 \) produces the (complex) eigenvector \( v = \begin{bmatrix} 2 - i \\ 2 \end{bmatrix} \).

Since \( \text{Re } v = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \) and \( \text{Im } v = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \), take \( P = \begin{bmatrix} 2 & -1 \\ 2 & 0 \end{bmatrix} \). Finally, compute
\[
P^{-1}AP = \frac{1}{2} \begin{bmatrix} 0 & 1 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} 1.52 & -0.7 \\ 0.56 & 0.4 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ 2 & 0 \end{bmatrix} = \begin{bmatrix} 0.96 & -0.28 \\ 0.28 & 0.96 \end{bmatrix}
\]

This final matrix, which has the proper form, is \( C \).

25. Write \( x = \text{Re } x + i(\text{Im } x) \), so that \( Ax = A(\text{Re } x) + iA(\text{Im } x) \). Since \( A \) is real, so are \( A(\text{Re } x) \) and \( A(\text{Im } x) \). Thus \( A(\text{Re } x) \) is the real part of \( Ax \) and \( A(\text{Im } x) \) is the imaginary part of \( Ax \).

**MATLAB Complex Eigenvalues**

The command \([V \ D] = \text{eig}(A)\) (mentioned in Section 5.3) works for matrices with complex eigenvalues. In this case \( V \) and the diagonal matrix \( D \) have some complex entries. For a \( 2 \times 2 \) real matrix with a complex eigenvalue, MATLAB tends to place the eigenvalue \( a - bi \) (where \( b > 0 \)) as the \((2, 2)\)-entry of \( D \). MATLAB does not produce matrices for a factorization \( A = PCP^{-1} \) of the sort described in this section.

For any matrix \( V \), the commands \( \text{real}(V) \) and \( \text{imag}(V) \) produce the real and imaginary parts of the entries in \( V \), displayed as matrices the same size as \( V \).
5.6 DISCRETE DYNAMICAL SYSTEMS

This section presents the climax to a crescendo of ideas that began in Section 1.10 and flowed through parts of Chapters 4 and 5. You need not have read all the material to appreciate the interesting applications in this section, but you will profit from a review of page 271 and Example 5 in Section 5.2.

KEY IDEAS

A solution of a first order homogeneous difference equation
\[ x_{k+1} = Ax_k \quad (k = 0, 1, 2, \ldots) \]  

is a sequence \{x_k\} that satisfies (1) and is described by a formula for each \(x_k\) that does not depend on the preceding terms in the sequence other than the initial term \(x_0\). In Section 5.1, you saw how a solution can be constructed when \(x_0\) is an eigenvector. When \(x_0\) is not an eigenvector, look for an eigenvector decomposition of \(x_0\):
\[ x_0 = c_1v_1 + \cdots + c_nv_n \]  

Each \(v_i\) is an eigenvector. (2)

To make (2) possible for any \(x_0\) in \(\mathbb{R}^n\), the section assumes that the \(n \times n\) matrix \(A\) has \(n\) linearly independent eigenvectors. If \(x_k = A^kx_0\), then
\[ x_k = c_1(\lambda_1)^k v_1 + \cdots + c_n(\lambda_n)^k v_n \]  

When \{\(x_k\)\} describes the “state” of a system at discrete times (denoted by \(k = 0, 1, 2, \ldots\)), the long-term behavior of this dynamical system is a description of what happens to \(x_k\) as \(k \to \infty\). The text focuses on the following important situation.

Let \(A\) be an \(n \times n\) matrix with \(n\) linearly independent eigenvectors, corresponding to eigenvalues such that \(|\lambda_1| \geq |\lambda_j| \) for \(j = 2, \ldots, n\). If \(x_0\) is given by (2) with \(c_1 \neq 0\), then for all sufficiently large \(k\),
\[ x_{k+1} \approx \lambda_1x_k \]  

Each entry in \(x_k\) grows by a factor of \(\lambda_1\).
\[ x_k \approx c_1(\lambda_1)^k v_1 \]  

\(x_k\) is approximately a multiple of \(v_1\), and so the ratio between any two entries in \(x_k\) is nearly the same as the corresponding ratio for \(v_1\).

STUDY NOTES

When the two approximations above are true in an application, the eigenvalue \(\lambda_1\) and the eigenvector \(v_1\) have interesting physical interpretations. Make sure you can describe these on an exam. (See the last four sentences in the solution of Example 1, for instance.)

The predator-prey model is rather primitive and provides only a starting point for more refined models. Still, you might enjoy considering what the model in Example 1 predicts if \(x_0\)
happens to be a multiple of \( v_2 = (5, 1) \), or if initially there are more than 5 owls for every 1 thousand rats, assuming \( p = .104 \).

The graphical descriptions of solutions to difference equations should help you understand what can happen to \( x_k \) as \( k \to \infty \). I hope you enjoy studying the figures even if your class does not have time to cover this part of the section. Only the simplest cases are shown, but these cases form the foundation for studying nonlinear dynamical systems which are widely used (but require calculus techniques not covered here). Even for nonlinear systems, eigenvalues and eigenvectors of certain matrices play an important role.

**SOLUTIONS TO EXERCISES**

1. **a.** The eigenvectors \( v_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \) and \( v_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \) form a basis for \( \mathbb{R}^2 \). To find the action of \( A \) on \( x_0 = \begin{bmatrix} 9 \\ 1 \end{bmatrix} \) express \( x_0 \) in terms of \( v_1 \) and \( v_2 \). That is, find \( c_1, c_2 \) such that \( x_0 = c_1v_1 + c_2v_2 \):

\[
\begin{bmatrix} 1 \\ -1 \\ 9 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 5 \\ 0 \\ 1 \\ -4 \end{bmatrix} \Rightarrow x_0 = 5v_1 - 4v_2
\]

Since \( v_1, v_2 \) are eigenvectors (for the eigenvalues 3 and 1/3):

\[
x_1 = Ax_0 = 5Av_1 - 4Av_2 = 5 \cdot 3v_1 - 4 \cdot (1/3)v_2
\]

\[
= \begin{bmatrix} 15 \\ 15 \end{bmatrix} - \frac{4}{3} \begin{bmatrix} 49/3 \\ 41/3 \end{bmatrix}
\]

**b.** Each time \( A \) acts on a linear combination of \( v_1 \) and \( v_2 \), the \( v_1 \) term is multiplied by the eigenvalue 3 and the \( v_2 \) term is multiplied by the eigenvalue 1/3.

\[
x_2 = Ax_1 = A[5(3)v_1 - 4(1/3)v_2] = 5(3)^2v_1 - 4(1/3)^2v_2
\]

In general, \( x_k = 5(3)^kv_1 - 4(1/3)^kv_2 \) for \( k \geq 0 \).

7. **a.** The matrix \( A \) in Exercise 1 has eigenvalues 3 and 1/3. Since \( |3| > 1 \) and \( |1/3| < 1 \), the origin is a saddle point.

**b.** The direction of greatest attraction is determined by \( v_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \), the eigenvector corresponding to the eigenvalue with absolute value less than 1. The direction of greatest repulsion is determined by \( v_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \), the eigenvector corresponding to the eigenvalue greater than 1.

**c.** The drawing below shows: (1) lines through the eigenvectors and the origin, (2) arrows toward the origin (showing attraction) on the line through \( v_2 \) and arrows away from the origin (showing repulsion) on the line through \( v_1 \), (3) several typical trajectories (with arrows) that show the general flow of points. No specific points other than \( v_1 \) and \( v_2 \)
were computed. This type of drawing is about all that one can make without using a computer to plot points.

**Remark:** Sketching trajectories for a dynamical system in which the origin is an attractor or a repellor is more difficult than the sketch in Exercise 7. There has been no discussion of the direction in which the trajectories “bend” as they move toward or away from the origin. For instance, if you rotate Figure 1 of Section 5.6 through a quarter-turn and relabel the axes so that $x_1$ is on the horizontal axis, then the new figure corresponds to the matrix $A$ with the diagonal entries .8 and .64 interchanged. In general, if $A$ is a diagonal matrix, with positive diagonal entries $a$ and $d$, unequal to 1, then the trajectories lie on the axes or on curves whose equations have the form $x_2 = r(x_1)^s$, where $s = (\ln d)/(\ln a)$ and $r$ depends on the initial point $x_0$. (See *Encounters with Chaos*, by Denny Gulick, New York: McGraw-Hill, 1992, pp. 147–150.)

**Study Tip:** If your instructor wants you to graph trajectories when the origin is an attractor or repellor, there will need to be some class discussion of exactly how to do this.

13. $A = \begin{bmatrix} .8 & .3 \\ -1.4 & 1.5 \end{bmatrix}$. First find the eigenvalues:

$$\begin{align*}
\det(A - \lambda I) &= (.8 - \lambda)(1.5 - \lambda) - (.3)(-1.4) = \lambda^2 - 2.3 + 1.32 \\
&= (\lambda - 1.2)(\lambda - 1.1) \\
&= 0
\end{align*}$$

Since both eigenvalues, 1.2 and 1.1, are greater than 1, the origin is a repellor. For the direction of greatest repulsion, find the eigenvector for the larger eigenvalue, 1.2:

$$[(A - 1.2I) \mathbf{0}] = \begin{bmatrix} -1.4 & .3 & 0 \\ -1.4 & .3 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & -3/4 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{x} = x_3 \begin{bmatrix} 3/4 \\ 1 \end{bmatrix}$$

Any multiple of $\begin{bmatrix} 3/4 \\ 1 \end{bmatrix}$, such as $\begin{bmatrix} 3 \\ 4 \end{bmatrix}$, determines the direction of greatest repulsion.
MATLAB Plotting Trajectories

Given a vector \( x \), the command \( x = A \cdot x \) will compute the “next” point on the trajectory. Use the up-arrow (↑) and <Enter> to repeat the command, over and over.

The following steps create a “trajectory” matrix \( T \) whose columns are the points \( x \), \( Ax \), \( A^2x \), ..., \( A^{15}x \). (Change 15 to any integer you wish.)

\[
T = x \\
\mathbf{for} \ j = 1:15 \\
\quad x = A \cdot x \\
\quad T = [T \ x] \\
\mathbf{end}
\]

After you type the line beginning with “for,” MATLAB will suspend all calculations (while you type additional lines) until you type “end” and press <Enter>.

If you want MATLAB itself to plot the points in \( T \), use the commands:

\[
\text{plot}(T(1,:), T(2,:), 'ob'), \text{grid}
\]

The ‘o’ produces a small circle at each point on the trajectory. The ‘b’ makes the circle blue. If you have the data for another trajectory stored in a matrix \( S \), you can plot both trajectories on the same graph:

\[
\text{plot}(T(1,:), T(2,:), 'ob', S(1,:), S(2,:), 'g'), \text{grid}
\]

Each new plot command erases the previous graph. If you want the new graph added to the previous graph, issue the command hold on before the next plot command.

For Exercise 17, the following commands will produce a graph of the first entry in each of the first nine columns of the matrix \( T \) constructed above.

\[
k = 0:8 \\
\text{plot}(k, T(1,1:9), '-'), \text{grid}
\]

To graph the sum of the two entries in the first nine columns of \( T \), with \( k \) already defined, enter

\[
\text{plot}(k, T(1,1:9)+T(2,1:9), '-'), \text{grid}
\]

The command ./ between two vectors of equal lengths divides each entry in the first vector by the corresponding entry in the second vector. Thus, to plot the quotient of the two entries in each column of \( T \), use

\[
\text{plot}(k, T(1,1:9)./ T(2,1:9), '-'), \text{grid}
\]
5.7  APPLICATIONS TO DIFFERENTIAL EQUATIONS

If you plan to take a course in differential equations, the material in this section will be a valuable reference. If you have already studied differential equations, you may gain new understanding as you work through this section.

KEY IDEAS

A basic solution of the differential equation \( x' = Ax \) is an eigenfunction \( x(t) = ve^{\lambda t} \), where \( \lambda \) is an eigenvalue of \( A \) and \( v \) is a corresponding eigenvector. In all examples and exercises in this section, every solution of \( x' = Ax \) is a linear combination of eigenfunctions. (This is because \( A \) is diagonalizable. The general case is usually handled in a full course in differential equations.) An initial condition, \( x(0) = x_0 \), determines the weights for the linear combination of eigenfunctions.

The eigenvalues of \( A \) determine the nature of the origin for the dynamical system described by \( x' = Ax \). Most of the discussion involves the case when \( A \) is \( 2 \times 2 \). If one eigenvalue is negative and one is positive, the origin is a saddle point. If the real parts of the eigenvalues are negative (this includes the case when both eigenvalues are real and negative), the origin is an attractor of the dynamical system. If the real parts are positive, the origin is a repellor. If an eigenvalue is complex, then the trajectory of a corresponding eigenfunction forms a spiral—either toward the origin, away from the origin, or on an ellipse around the origin, depending on the real part of the eigenvalue.

Study Tip: Note that conditions on eigenvalues here differ from those in Section 5.6. For differential equations, the real parts of the eigenvalues determine the nature of the trajectories; for difference equations, the absolute values of the eigenvalues are important. You can remember this if you note that a basic solution \( ve^{\lambda t} \) of \( x' = Ax \) tends to \( 0 \) (as \( t \to \infty \)) only if the real part of \( \lambda \) is negative. In contrast, a basic solution \( \lambda^k v \) of \( x_{k+1} = Ax_k \) tends to \( 0 \) (as \( k \to \infty \)) only if the absolute value of \( \lambda \) is less than 1.

SOLUTIONS TO EXERCISES

1. The eigenfunctions for \( x' = Ax \) are \( v_1e^{4t} \) and \( v_2e^{2t} \). The general solution of \( x' = Ax \) has the form

\[
\begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} -3 & 1 \\ 1 & 1 \end{pmatrix} e^{4t} + \begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix} e^{2t}
\]

The initial condition \( x(0) = (-6, 1) \) determines \( c_1 \) and \( c_2 \):

\[
\begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} -3 & 1 \\ 1 & 1 \end{pmatrix} e^{4(0)} + \begin{pmatrix} c_1 & c_2 \end{pmatrix} \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix} e^{2(0)} = \begin{pmatrix} -6 & 1 \\ 1 & 1 \end{pmatrix}
\]

\[
\begin{pmatrix} -3 & -6 \\ 1 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 1 \\ -3 & -1 & -6 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 5/2 \\ 0 & 1 & -3/2 \end{pmatrix}
\]
Thus \( c_1 = 5/2 \), \( c_2 = -3/2 \), and \( \mathbf{x}(t) = \frac{5}{2} \begin{bmatrix} -3 \\ 1 \end{bmatrix} e^{4t} - \frac{3}{2} \begin{bmatrix} -1 \\ 1 \end{bmatrix} e^{2t} \).

**Checkpoint:** Let \( A \) be the matrix \( \begin{bmatrix} 1 & 3/2 \\ -1/2 & -1 \end{bmatrix} \), obtained by dividing the matrix in Exercise 3 by 2. Now the eigenvalues of \( A \) are .5 and -.5. Is the origin an attractor or a saddle point for the equation \( \mathbf{x}' = A\mathbf{x} \)?

7. Use the eigenvectors \( \mathbf{v}_1 = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \) and \( \mathbf{v}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \) (found in Exercise 5) to create \( P = [\mathbf{v}_1 \ \mathbf{v}_2] \). Match the eigenvectors with the eigenvalues in \( D = \begin{bmatrix} 4 & 0 \\ 0 & 6 \end{bmatrix} \). The details of the substitution of \( \mathbf{x} = Py \) into \( \mathbf{x}' = A\mathbf{x} \) are given in the answer section of the text.

**Helpful Hint:** The idea of changing variables to uncouple a differential equation is fairly common in engineering texts. Exercises 7 and 8 test your understanding of the value of a diagonalization. (You might see such a question on an exam.)

13. An eigenvalue of \( A \) is \( \lambda = 1 + 3i \), with eigenvector \( \mathbf{v} = (1 + i, 2) \). The complex eigenfunctions \( \mathbf{v} e^{\lambda t} \) and \( \overline{\mathbf{v}} e^{\lambda \bar{t}} \) provide a basis for the solution space of all complex solutions of \( \mathbf{x}' = A\mathbf{x} \). The general (complex) solution is

\[
c_1 \begin{bmatrix} 1 + i \\ 2 \end{bmatrix} e^{(1+3i)t} + c_2 \begin{bmatrix} 1 - i \\ 2 \end{bmatrix} e^{(1-3i)t} \quad (c_1 \text{ and } c_2 \text{ are complex})
\]

Use the real and imaginary parts of \( \mathbf{v} e^{(1+3i)t} \) to build the general real solution. Rewrite \( \mathbf{v} e^{(1+3i)t} \) as:

\[
\begin{bmatrix} 1 + i \\ 2 \end{bmatrix} e^{(1+3i)t} = \begin{bmatrix} 1 + i \\ 2 \end{bmatrix} (\cos 3t + i \sin 3t) e^t
\]

\[
= \begin{bmatrix} \cos 3t - \sin 3t \\ 2 \cos 3t \end{bmatrix} e^t + i \begin{bmatrix} \sin 3t + \cos 3t \\ 2 \sin 3t \end{bmatrix} e^t
\]

The general real solution has the form

\[
c_1 \begin{bmatrix} \cos 3t - \sin 3t \\ 2 \cos 3t \end{bmatrix} e^t + c_2 \begin{bmatrix} \sin 3t + \cos 3t \\ 2 \sin 3t \end{bmatrix} e^t \quad (c_1 \text{ and } c_2 \text{ are real})
\]

The trajectories are spirals because the eigenvalues are complex. The spirals tend away from the origin because the real parts of the eigenvalues are positive.

19. Substitute \( R_1 = 1/5 \), \( R_2 = 1/3 \), \( C_1 = 4 \), and \( C_2 = 3 \) into the formula for \( A \) given in Example 1, and use a matrix program to find the eigenvalues and eigenvectors:
A = \begin{bmatrix} -2 & 3/4 \\ 1 & -1 \end{bmatrix}, \quad \lambda_1 = -0.5: \quad v_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad \lambda_2 = -2.5: \quad v_2 = \begin{bmatrix} -3 \\ 2 \end{bmatrix}

General solution: \quad x(t) = c_1 \begin{bmatrix} 1 \\ 2 \end{bmatrix} e^{-0.5t} + c_2 \begin{bmatrix} -3 \\ 2 \end{bmatrix} e^{-2.5t}.

The condition \( x(0) = \begin{bmatrix} 4 \\ 4 \end{bmatrix} \) implies that \( \begin{bmatrix} 1 & -3 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix} \). By a matrix program, \( c_1 = 5/2 \) and \( c_2 = -1/2 \), so that

\[
\begin{bmatrix} v_1(t) \\ v_2(t) \end{bmatrix} = x(t) = \frac{5}{2} \begin{bmatrix} 1 \\ 2 \end{bmatrix} e^{-0.5t} - \frac{1}{2} \begin{bmatrix} -3 \\ 2 \end{bmatrix} e^{-2.5t}
\]

Helpful Hint: (for 21 and 22) Find the general real solution before you use the initial condition to find the constants \( c_1 \) and \( c_2 \). Otherwise, your \( c_1 \) and \( c_2 \) will probably be complex, and you will have to do unnecessary complex arithmetic to write the solution using only real scalars.

Answer to Checkpoint: One eigenvalue is positive and one is negative, so the origin is a saddle point. If you consider the difference equation \( x_{k+1} = Ax_k \) (with the same matrix \( A \)), then the origin is an attractor, because both eigenvalues are less than 1 in absolute value. Be careful if you have a test that covers both Sections 5.6 and 5.7.

MATLAB Solutions of Differential Equations

For the eigenvalues of \( A \), use \( \text{ev} = \text{eig}(A) \). The eigenvectors shown in the text’s answers were produced using commands such as

\[ \mathbf{v} = \text{nulbasis}(A - \text{ev}(1) \cdot \text{eye}(3)) \]

If the eigenvalue \( \text{ev}(1) \) is complex, the eigenvector \( \mathbf{v} \) will be complex. The real and imaginary parts of \( \mathbf{v} \) are \( \text{real}(\mathbf{v}) \) and \( \text{imag}(\mathbf{v}) \).

If you use the command \( [\mathbf{P} \ \mathbf{D}] = \text{eig}(A) \), your eigenvectors should be multiples of those in the text’s answer (when the eigenspaces are one-dimensional). To test whether a vector \( \mathbf{v} \) is a multiple of a vector \( \mathbf{w} \), compute \( \mathbf{v} ./ \mathbf{w} \). This divides each entry in \( \mathbf{v} \) by the corresponding entry in \( \mathbf{w} \). If \( \mathbf{v} \) is a multiple of \( \mathbf{w} \), the result of \( \mathbf{v} ./ \mathbf{w} \) should be a vector whose entries are all equal.

5.8 ITERATIVE ESTIMATES FOR EIGENVALUES

The algorithms in this section illustrate another use of the eigenvector decomposition described in Section 5.6 (on page 301). Other methods for eigenvalue estimation were mentioned in Section 5.2 (on page 279).
STUDY NOTES

Throughout the section, we suppose that the initial vector \( \mathbf{x}_0 \) can be written as 
\[ \mathbf{x}_0 = c_1 \mathbf{v}_1 + \cdots + c_n \mathbf{v}_n, \]
where \( \mathbf{v}_1, \ldots, \mathbf{v}_n \) are eigenvectors of \( A \) and \( c_1 \neq 0 \). (In practice, you will not know \( c_1 \mathbf{v}_1, \ldots, c_n \mathbf{v}_n \). This eigenvector decomposition is used only to explain why the power method works.)

**The Power Method:** Assume the eigenvalue \( \lambda_1 \) for \( \mathbf{v}_1 \) is a strictly dominant eigenvalue (so that \( |\lambda_1| > |\lambda_j| \) for \( j = 2, \ldots, n \)). Then, for large \( k \), the line through \( A^k \mathbf{x}_0 \) and \( \mathbf{0} \) nearly coincides with the line through \( \mathbf{v}_1 \) and \( \mathbf{0} \). The vector \( A^k \mathbf{x}_0 \) itself may never approach a multiple of \( \mathbf{v}_1 \) (see Exercise 21), but if each \( A^k \mathbf{x}_0 \) is scaled so its largest entry is 1, then the scaled vectors approach an eigenvector (a multiple of \( \mathbf{v}_1 \)) as \( k \to \infty \).

**The Inverse Power Method:** You must start with an initial estimate \( \alpha \) for a particular eigenvalue, say \( \lambda_2 \), and \( \alpha \) must be closer to \( \lambda_2 \) than to any other eigenvalue of \( A \). In this case, \( 1/(\lambda_2 - \alpha) \) is a strictly dominant eigenvalue of the matrix \( B = (A - \alpha I)^{-1} \). The inverse power method avoids computing \( B \). Instead of multiplying \( \mathbf{x}_k \) by \( B \) to get \( \mathbf{x}_{k+1} \) (suitably scaled), you solve the equation \( (A - \alpha I)\mathbf{y}_k = \mathbf{x}_k \) for \( \mathbf{y}_k \) and then scale \( \mathbf{y}_k \) to produce \( \mathbf{x}_{k+1} \).

### SOLUTIONS TO EXERCISES

1. The vectors in the sequence 
\[ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ .25 \end{bmatrix}, \begin{bmatrix} 1 \\ .3158 \end{bmatrix}, \begin{bmatrix} 1 \\ .3298 \end{bmatrix}, \begin{bmatrix} 1 \\ .3326 \end{bmatrix} \]
approach an eigenvector \( \mathbf{v}_1 \). Of these vectors, the last one, \( \mathbf{x}_4 \), is probably the best estimate of \( \mathbf{v}_1 \). To compute an estimate of \( \lambda_1 \), multiply one of the vectors by \( A \) and examine its entries. Again, the best information probably comes from \( A \mathbf{x}_4 = \begin{bmatrix} 4.9978 \\ 1.6652 \end{bmatrix} \) whose entries are approximately \( \lambda_1 \) times the entries in \( \mathbf{x}_4 \). From the first entry, the estimate of \( \lambda_1 \) is 4.9978.

   The computed value of \( A \mathbf{x}_4 \) can be used as an estimate of the direction of the eigenspace.
Study Tip: Exercises 1–6 make good exam questions because they test your understanding of the power method without requiring extensive calculation.

7. The data in the table below and the tables in Exercise 19 were produced by MATLAB, which carried more decimal places than shown here.

<table>
<thead>
<tr>
<th>( k )</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_k )</td>
<td>[ 1 ]</td>
<td>[ .75 ]</td>
<td>[ 1 ]</td>
<td>[ .9932 ]</td>
<td>[ 1 ]</td>
<td>[ .9998 ]</td>
</tr>
<tr>
<td></td>
<td>[ 0 ]</td>
<td>[ 1 ]</td>
<td>[ .9565 ]</td>
<td>[ 1 ]</td>
<td>[ .9990 ]</td>
<td>[ 1 ]</td>
</tr>
<tr>
<td>( Ax_k )</td>
<td>[ 0 ]</td>
<td>[ 11.5 ]</td>
<td>[ 12.70 ]</td>
<td>[ 12.959 ]</td>
<td>[ 12.9927 ]</td>
<td>[ 12.9990 ]</td>
</tr>
<tr>
<td></td>
<td>[ 8 ]</td>
<td>[ 11.0 ]</td>
<td>[ 12.78 ]</td>
<td>[ 12.946 ]</td>
<td>[ 12.9948 ]</td>
<td>[ 12.9987 ]</td>
</tr>
<tr>
<td>( \mu_k )</td>
<td>[ 8 ]</td>
<td>[ 11.5 ]</td>
<td>[ 12.78 ]</td>
<td>[ 12.959 ]</td>
<td>[ 12.9948 ]</td>
<td>[ 12.9990 ]</td>
</tr>
</tbody>
</table>

The exact eigenvalues are 13 and –2. The subspaces determined by \( A^k x \) are lines whose slopes alternate above and below the slope of the eigenspace. (The eigenspace is the line \( x_2 = x_1 \).)

13. If the eigenvalues close to 4 and –4 have different absolute values, then one of these eigenvalues is a strictly dominant eigenvalue, so the power method will work. But the power method depends on powers of the quotients \( \lambda_2/\lambda_1 \) and \( \lambda_3/\lambda_1 \) going to zero. If \( |\lambda_2/\lambda_1| \) is close to 1, its powers will go to zero slowly.

15. Suppose \( Ax = \lambda x \), with \( x \neq 0 \). For any \( \alpha \), \( Ax - \alpha I x = (\lambda - \alpha) x \), and \( (A - \alpha I) x = (\lambda - \alpha) x \). If \( \alpha \) is not an eigenvalue of \( A \), then \( A - \alpha I \) is invertible and \( \lambda - \alpha \) is not 0; hence

\[ x = (A - \alpha I)^{-1} (\lambda - \alpha) x \] and \( (\lambda - \alpha)^{-1} x = (A - \alpha I)^{-1} x \)

This last equation shows that \( x \) is an eigenvector of \( (A - \alpha I)^{-1} \) corresponding to the eigenvalue \( (\lambda - \alpha)^{-1} \).

19. a. The data in the table on the next page show that \( \mu_6 = 30.2887 = \mu_7 \) to four decimal places. Actually, to six places, the largest eigenvalue is 30.288685, with eigenvector (.957629, .688937, 1, .943782).
b. The inverse power method (with $\alpha = 0$) produces $v_1 = \mu_1^{-1} = .010141$, and $v_2 = .0101501$, which seems to be accurate to at least four places. Actually, $v_2$ is accurate to six places, $v_3$ is accurate to eight places, and $v_4$ is accurate to ten places. The convergence is so rapid because the next-to-smallest eigenvalue is near .85, which is much farther away from 0 than .0101501. The vector $x_4$ gives an estimate for the eigenvector that is accurate to seven places in each entry.

<table>
<thead>
<tr>
<th>$k$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_k$</td>
<td>1</td>
<td>1</td>
<td>.99</td>
<td>.961</td>
<td>.9581</td>
<td>.9577</td>
<td>.957637</td>
<td>.957630</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>.7</td>
<td>.71</td>
<td>.691</td>
<td>.6893</td>
<td>.6890</td>
<td>.688942</td>
<td>.688938</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>.8</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>.7</td>
<td>.93</td>
<td>.942</td>
<td>.9436</td>
<td>.9438</td>
<td>.943778</td>
<td>.943781</td>
</tr>
<tr>
<td>$Ax_k$</td>
<td>10</td>
<td>26.2</td>
<td>29.4</td>
<td>29.05</td>
<td>29.01</td>
<td>29.006</td>
<td>29.0054</td>
<td>29.0053</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>18.8</td>
<td>21.1</td>
<td>20.90</td>
<td>20.87</td>
<td>20.868</td>
<td>20.8671</td>
<td>20.8670</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>26.5</td>
<td>30.6</td>
<td>30.32</td>
<td>30.29</td>
<td>30.289</td>
<td>30.2887</td>
<td>30.2887</td>
</tr>
<tr>
<td>$\mu_k$</td>
<td>10</td>
<td>26.5</td>
<td>30.6</td>
<td>30.32</td>
<td>30.29</td>
<td>30.2892</td>
<td>30.2887</td>
<td>30.2887</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$k$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_k$</td>
<td>1</td>
<td>-.6098</td>
<td>-.60401</td>
<td>-.603973</td>
<td>-.6039723</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>-.2439</td>
<td>-.25105</td>
<td>-.251134</td>
<td>-.2511351</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>.1463</td>
<td>.14890</td>
<td>.148953</td>
<td>.1489534</td>
</tr>
<tr>
<td>$Ax_k$</td>
<td>25</td>
<td>-59.56</td>
<td>-59.5041</td>
<td>-59.5044</td>
<td>-59.50438</td>
</tr>
<tr>
<td></td>
<td>-41</td>
<td>98.61</td>
<td>98.5211</td>
<td>98.5217</td>
<td>98.52170</td>
</tr>
<tr>
<td>$\mu_k$</td>
<td>-.024</td>
<td>.010141</td>
<td>.0101501</td>
<td>.010150059</td>
<td>.0101500484</td>
</tr>
</tbody>
</table>
MATLAB  Power Method and Inverse Power Method

Use \texttt{format long} to display 15 decimal digits in your data. The algorithms below assume that $A$ has a strictly dominant eigenvalue, and the initial vector is $x$, with largest entry 1 (in magnitude). (If your initial vector is called $x0$, rename it by entering $x = x0$.)

\textbf{The Power Method} When the following sequence of commands is performed over and over, the values of $x$ approach (in many cases) an eigenvector for a strictly dominant eigenvalue:

\begin{align*}
  y &= A \cdot x \quad (1) \\
  [t \ r] &= \max(\text{abs}(y)); \ mu = y(r) \quad \text{mu = estimate for eigenvalue} \quad (2) \\
  x &= y/y(r) \quad \text{Estimate for the eigenvector} \quad (3)
\end{align*}

In (2), $t$ is the absolute value of the largest entry in $y$ and $r$ is the index of that entry. As these commands are repeated, the numbers that appear in $y(r)$ are the $\mu_k$ that approach the dominant eigenvalue.

Recall that MATLAB commands can be recalled by the up-arrow key ($\uparrow$). After entering (1) – (3), your keystrokes can be

\begin{align*}
  \uparrow \uparrow \uparrow \quad \text{<Enter>} \quad \uparrow \uparrow \uparrow \quad \text{<Enter>} \quad \uparrow \uparrow \uparrow \quad \text{<Enter>}
\end{align*}

and so on. Alternatively, you could enclose lines (1) – (3) in a loop (See the MATLAB box after section 5.6.)

\textbf{The Inverse Power Method} Store the initial estimate of the eigenvalue in the variable $a$, and enter the command $C = A - a \cdot \text{eye}(n)$, where $n$ is the number of columns of $A$. Then enter the commands

\begin{align*}
  y &= C \cdot x \quad \text{Solves} \ (A - aI)y = x \quad (1) \\
  [t \ r] &= \max(\text{abs}(y)); \ nu = a + 1/y(r) \quad \text{nu = estimated eigenvalue} \quad (2) \\
  x &= y/y(r) \quad \text{Estimate for the eigenvector} \quad (3)
\end{align*}

As these commands are repeated (using $\uparrow \uparrow \uparrow \quad \text{<Enter>}$ each time), lines (2) and (3) produce the sequences $\{v_k\}$ and $\{x_k\}$ described in the text.

\textbf{Displaying Data} If your computer screen displays only 24 or 25 lines, vectors in the sequence $\{x_k\}$ tend to scroll off the screen soon after you compute them. To see more vectors at once, and to compare their entries more easily, you can display them as row vectors. Change (1) to $y = A \cdot x$; $y'$ (power method) or $y = C \cdot x$; $y'$ (inverse power), and for both methods, change (3) to $x = y/y(r); \ x'$.

For even more data on your screen, use the command \texttt{format compact}, which removes extra lines between data displays. The simple command \texttt{format} returns everything to normal.
Chapter 5  GLOSSARY CHECKLIST

Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

**algebraic multiplicity**: The multiplicity of an eigenvalue as . . .

**attractor** (of a dynamical system in \( \mathbb{R}^2 \)): The origin of \( \mathbb{R}^2 \) when all trajectories tend . . .

**\( B \)-matrix** (for \( T \)): A matrix \([T]_B\) for a linear transformation \( T:V \rightarrow V \) relative to a basis \( B \) for \( V \), with the property that . . .

**characteristic equation** (of \( A \)): . . .

**characteristic polynomial** (of \( A \)): . . .

**companion matrix**: A special form of matrix whose characteristic . . . is . . .

**complex eigenvalue**: A nonreal root of the characteristic equation of an \( n \times n \) matrix \( A \), when . . .

**complex eigenvector**: A nonzero vector \( x \) in \( \mathbb{C}^n \) such that . . ., where . . .

**decoupled system**: A difference equation \( y_{k+1} = Ay_k \), or a differential equation, in which \( A \) is a . . .

**determinant** (of a square matrix \( A \)): A number \( \det A \) computed from \( A \); equal to . . .

**diagonalizable** (matrix): A matrix that may be written in factored form as . . .

**difference equation** (or **linear recurrence relation**): An equation of the form . . . whose solution is . . .

**discrete linear dynamical system** (or briefly, a **dynamical system**): A difference equation of the form . . . that describes . . .

**eigenfunction** (of the equation \( x'(t) = Ax(t) \)): A function of the form . . .

**eigenspace** (of \( A \) corresponding to \( \lambda \)): The set of . . . solutions of . . .

**eigenvalue** (of \( A \)): A scalar \( \lambda \) such that . . .

**eigenvector** (of \( A \)): A . . . vector \( x \) such that . . .

**eigenvector basis**: A basis consisting entirely of . . .

**eigenvector decomposition** (of \( x \)): An equation \( x = . . .

**fundamental set of solutions** (for \( x' = Ax \)): A basis for . . .

**\( \text{Im } x \)**: The vector in \( \mathbb{R}^n \) formed from . . .

**invariant subspace** (for \( A \)): A subspace \( H \) such that . . .

**inverse power method**: An algorithm for estimating . . .

Copyright © 2016 Pearson Education, Inc.
matrix for $T$ relative to bases $B$ and $C$: A matrix $M$ for a linear transformation $T: V \rightarrow W$ with the property that . . . . When $W = V$ and $C = B$, the matrix $M$ is called . . . and is denoted by . . . .

power method: An algorithm for estimating . . . .

repellor (of a dynamical system in $\mathbb{R}^2$): The origin in $\mathbb{R}^2$ when all trajectories . . . tend . . . .

Rayleigh quotient: $R(x) = . . .$. An estimate of . . . .

$\Re x$: The vector in $\mathbb{R}^n$ formed from . . . .

saddle point (of a dynamical system in $\mathbb{R}^2$): The origin in $\mathbb{R}^2$ when . . . .

similar (matrices): Matrices $A$ and $B$ such that . . . .

spiral point (of a dynamical system in $\mathbb{R}^2$): The origin in $\mathbb{R}^2$ when . . . .

stage-matrix model: A difference equation $x_{k+1} = Ax_k$ where $x_k$ lists . . . .

strictly dominant eigenvalue: An eigenvalue $\lambda_1$ of a matrix $A$ with the property that . . . .

trace (of a square matrix $A$): The . . . , denoted by $\text{tr} A$.

trajectory: The graph of a solution $\{x_0, x_1, x_2, \ldots\}$ of a . . . .

Also, the graph of $x(t)$ for $t \geq 0$, when . . . .
6.1 INNER PRODUCT, LENGTH, AND ORTHOGONALITY

The concepts of length, distance, and orthogonality introduced in this section are essential for many geometric descriptions in the rest of the text.

STUDY NOTES

The first half of the section is computational and easily learned. The second half, however, requires more attention. Read it carefully. The concepts of orthogonality and orthogonal complements are the foundation for the rest of the chapter. In fact, Theorem 3 is sometimes called the Fundamental Theorem of Linear Algebra.

SOLUTIONS TO EXERCISES

1. \[ \mathbf{u} = \begin{bmatrix} -1 \\ 2 \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} 4 \\ 6 \end{bmatrix}, \quad \mathbf{u} \cdot \mathbf{u} = (-1)^2 + 2^2 = 5, \]
   \[ \mathbf{v} \cdot \mathbf{u} = 4(-1) + 6(2) = 8, \quad \frac{\mathbf{v} \cdot \mathbf{u}}{\mathbf{u} \cdot \mathbf{u}} = \frac{8}{5} \]

7. \[ \mathbf{w} = (3, -1, -5), \quad ||\mathbf{w}||^2 = \mathbf{w} \cdot \mathbf{w} = 3^2 + (-1)^2 + (-5)^2 = 35. \quad \text{So } ||\mathbf{w}|| = \sqrt{35}. \]

13. \[ \mathbf{x} = (10, -3), \quad \mathbf{y} = (-1, -5) \]
   \[ ||\mathbf{x} - \mathbf{y}||^2 = [10 - (-1)]^2 + [-3 - (-5)]^2 = 121 + 4 = 125 \]
   \[ \text{dist}(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}|| = \sqrt{125}, \text{ or } 5\sqrt{5} \]

19. a. See the definition of $||\mathbf{v}||$.
   b. See Theorem I(c).
   c. See the discussion of Fig. 5.
d. Think about a $2 \times 2$ matrix of zeros and ones, and see Theorem 3 for statements that are always true.

e. See the box following Example 6.

21. Theorem l(b): $(u + v) \cdot w = (u + v)^T w = (u^T + v^T)w = u^T w + v^T w = u \cdot w + v \cdot w$. The second and third equalities used Theorems 3(b) and 2(c), respectively, from Section 2.1. Theorem l(c): $(cu) \cdot v = (cu)^T v = (cu^T)v = c(u^T v) = c(u \cdot v)$, by Theorems 3(c) and 2(d) in Section 2.1. Also, $u \cdot (cv) = u^T (cv) = cu^T v = c(u \cdot v)$.

25. When $v = \begin{bmatrix} a \\ b \end{bmatrix}$, the set $H$ of all vectors $\begin{bmatrix} x \\ y \end{bmatrix}$ that are orthogonal to $v$ is the subspace of vectors whose entries satisfy $ax + by = 0$. If $a \neq 0$, then $x = -(b/a)y$, with $y$ a free variable. Then $H$ is a line through the origin. A natural choice for a basis for this subspace is $\begin{bmatrix} -b \\ a \end{bmatrix}$. If $a = 0$ and $b \neq 0$, then the vectors in $H$ satisfy $by = 0$. Since $b$ is nonzero, $y = 0$ and $x$ is free. A basis for $H$ is $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$. Note, however, that $\begin{bmatrix} -b \\ a \end{bmatrix}$ is also a basis, since $a = 0$ and $b \neq 0$.

Finally, if $a$ and $b$ are both zero, then $H$ is $\mathbb{R}^2$ itself, because the equation $0x + 0y = 0$ places no restrictions on $x$ or $y$.

27. If $y$ is orthogonal to $u$ and $v$, then $y \cdot u = 0$ and $y \cdot v = 0$, and hence by a property of the inner product, $y \cdot (u + v) = y \cdot u + y \cdot v = 0 + 0 = 0$. So $y$ is orthogonal to $u + v$.

29. Take a typical vector $w = c_1 v_1 + \cdots + c_p v_p$ in $W$. If $x$ is orthogonal to each $v_j$, then using the linearity of the inner product (Theorem l(b) and l(c)), $w \cdot x = (c_1 v_1 + \cdots + c_p v_p) \cdot x = c_1 v_1 \cdot x + \cdots + c_p v_p \cdot x = 0$. So $x$ is orthogonal to each $w$ in $W$.

31. Suppose $x$ is in $W$ and $W^\perp$. Then, since $x$ is in $W^\perp$, $x$ is orthogonal to every vector in $W$, including $x$ itself. So $x \cdot x = 0$. This is true only if $x = 0$. This problem shows that $W \cap W^\perp$ is the zero subspace. (See Exercise 32 in Section 4.1.)

MATLAB The inner product of real column vectors $u$ and $v$ is $u^*v$ (and $v^*u$); the length of $v$ is $\text{norm}(v)$. See the MATLAB note for Section 2.1.
6.2 ORTHOGONAL SETS

Orthogonal sets and orthogonal bases are used throughout the chapter. The “orthogonal projection” discussed in this section is an important special case of the orthogonal projections studied in Section 6.3.

STUDY NOTES

The proofs of Theorems 4 and 5 are worth studying because they involve a calculation you will see and use several times.

The subsection entitled An Orthogonal Projection is simple but extremely important. Also, the geometric interpretation of Theorem 5 on page 341 will be helpful when you study Theorem 8 in the next section.

The attention paid to Theorems 6 and 7 will depend on what your instructor plans to do later in the chapter. In some cases, an instructor may discuss Theorems 6 and 7 only for square matrices. The $m \times n$ case is needed later, for Theorems 10, 12 and 15. Remember: the term orthogonal matrix applies only to a square matrix. Also, the columns of an orthogonal matrix must be orthonormal, not simply orthogonal.

SOLUTIONS TO EXERCISES

1. $u = \begin{bmatrix} -1 \\ 4 \\ -3 \end{bmatrix}$, $v = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix}$, $w = \begin{bmatrix} 3 \\ -4 \\ -7 \end{bmatrix}$, $u \cdot v = -5 + 8 - 3 = 0$, $u \cdot w = -3 - 16 + 21 = 2 \neq 0$. The set $\{u, v, w\}$ is not orthogonal. There is no need to check $v \cdot w$.

7. $u_1 = \begin{bmatrix} 2 \\ -3 \end{bmatrix}$, $u_2 = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$, $x = \begin{bmatrix} 9 \\ -7 \end{bmatrix}$. $u_1 \cdot u_2 = 12 - 12 = 0$, so $\{u_1, u_2\}$ is an orthogonal set.

Since the vectors are nonzero, $u_1$ and $u_2$ are linearly independent, by Theorem 4. But two such vectors in $\mathbb{R}^2$ automatically form a basis for $\mathbb{R}^2$. So $\{u_1, u_2\}$ is an orthogonal basis for $\mathbb{R}^2$. By Theorem 5,

$$x = \frac{x \cdot u_1}{u_1 \cdot u_1} u_1 + \frac{x \cdot u_2}{u_2 \cdot u_2} u_2 = \frac{18 + 21}{4 + 9} u_1 + \frac{54 - 28}{36 + 16} u_2 = \begin{bmatrix} 3 \\ -3 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 6 \\ -3 \end{bmatrix}$$

13. $y = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$, $u = \begin{bmatrix} 4 \\ -7 \end{bmatrix}$. The orthogonal projection of $y$ onto $u$ is

$$\hat{y} = \frac{y \cdot u}{u \cdot u} u = \frac{8 - 21}{16 + 49} u = -\frac{13}{65} u = -\frac{1}{5} \begin{bmatrix} 4 \\ -7 \end{bmatrix} = \begin{bmatrix} -4/5 \\ 7/5 \end{bmatrix}$$

The component of $y$ orthogonal to $u$ is $y - \hat{y} = \begin{bmatrix} 2 \\ 3 \end{bmatrix} - \begin{bmatrix} -4/5 \\ 7/5 \end{bmatrix} = \begin{bmatrix} 14/5 \\ 8/5 \end{bmatrix}$.  

Copyright © 2016 Pearson Education, Inc.
Thus, $y = \hat{y} + (y - \hat{y}) = \begin{bmatrix} -4/5 \\ 7/5 \end{bmatrix} + \begin{bmatrix} 14/5 \\ 8/5 \end{bmatrix}.$

19. $u = \begin{bmatrix} -.6 \\ .8 \end{bmatrix}, v = \begin{bmatrix} .8 \\ .6 \end{bmatrix}, u \cdot v = -.48 + .48 = 0,$ so $\{u, v\}$ is an orthogonal set. Also, $\|u\|^2 = u \cdot u = (-.6)^2 + (.8)^2 = .36 + .64 = 1.$ Similarly, $\|v\|^2 = v \cdot v = 1.$ Thus $\{u, v\}$ is an orthonormal set.

23. a. See Example 3, for instance. b. See Theorem 5.
   c. See the paragraph after Example 5. d. See the paragraph before Example 7.
   e. See Example 4.

25. $(Ux) \cdot (Uy) = (Ux)^T(Uy) = x^T U^T U y = x^T y = x \cdot y$ (because $U^T U = I$). If $y = x,$ Theorem 7(b) says that $\|Ux\|^2 = \|x\|^2,$ which implies part (a).
   Part (c) of Theorem 7 follows immediately from part (b).

**Study Tip:** If your instructor emphasizes orthogonal matrices, work Exercises 27–29. (They make good test questions.) In each case, mention explicitly how you use the fact that the matrices are square. Don’t read the solutions below until you have first written your own solution.

27. If $U$ has orthonormal columns, then $U^T U = I,$ by Theorem 6. If $U$ is also square, then the equation $U^T U = I$ implies that $U$ is invertible, by the Invertible Matrix Theorem.

29. Since $U$ and $V$ are orthogonal, each is invertible. By Theorem 6 in Section 2.2, $UV$ is invertible and $(UV)^{-1} = V^{-1} U^{-1} = V^T U^T = (UV)^T$ (by Theorem 3 in Section 2.1). Thus $UV$ is an orthogonal matrix.

31. The full solution is in the text.

---

**Mastering Linear Algebra Concepts: Orthogonal Basis**

To the review sheet(s) you have on “basis”, add the concepts of an orthogonal basis and an orthonormal basis for a subspace. You need to know what special properties they posses.

- **basic definitions**
  Pages 339 and 342
- **equivalent descriptions**
  Theorems 4 and 6
- **geometric interpretation**
  Figs. 1 and 6
- **special cases**
  Matrix with orthonormal columns
- **examples and counterexamples**
  Examples 2 and 5
- **algorithms and computations**
  Example 2
- **connections with other concepts**
  Orthogonal matrix

---

Copyright © 2016 Pearson Education, Inc.
MATLAB Orthogonality

In Exercises 1–10 and 17–22, the fastest way (counting the keystrokes) in MATLAB to test a set such as \{u_1, u_2, u_3\} for orthogonality is to use a matrix $U = [u_1 \ u_2 \ u_3]$ whose columns are the vectors from the set, and test whether $U^T U$ is a diagonal matrix. See the proof of Theorem 6.

For column vectors $y$ and $u$, the orthogonal projection of $y$ onto $u$ is

$$(y^T u) / (u^T u) * u$$

The parentheses (and the final *) are essential. MATLAB computes the scalar quotient $(y^T u)/(u^T u)$ and then multiplies $u$ by this scalar.

6.3 ORTHOGONAL PROJECTIONS

A familiar idea in Euclidean geometry is to construct a line segment perpendicular to a line or plane. This section treats an analogous situation in $\mathbb{R}^n$, namely, the orthogonal projection of a vector (a point in $\mathbb{R}^n$) onto a subspace. The case when the subspace is a line through the origin was already examined in Section 6.2.

KEY IDEAS

If $y$ is in $\mathbb{R}^n$ and if $W$ is a subspace of $\mathbb{R}^n$, then the orthogonal projection of $y$ onto $W$, denoted by $\hat{y}$ or $\text{proj}_W y$, has two important properties:

(i) $y - \hat{y}$ is orthogonal to $W$ (so $y$ is the sum of a vector $\hat{y}$ in $W$ and a vector $y - \hat{y}$ in $W^\perp$), and

(ii) $\hat{y}$ is the closest point in $W$ to $y$.

Properties (i) and (ii) are described in the Orthogonal Decomposition Theorem and the Best Approximation Theorem. You should learn the statements of both theorems. (By now you probably know that whenever a theorem has an official name, an instructor has an easy time asking test questions about it.) When you need one of these theorems in a discussion (homework or test question), you should mention the theorem by name.

If your class covers Theorem 10, then the paragraph following the theorem will help you understand the difference between an orthogonal matrix (which must be square) and a rectangular matrix with orthonormal columns.
SOLUTIONS TO EXERCISES

1. \[
\begin{bmatrix}
0 \\
1 \\
-1
\end{bmatrix},
\begin{bmatrix}
3 \\
1 \\
1
\end{bmatrix},
\begin{bmatrix}
1 \\
0 \\
-4
\end{bmatrix},
\begin{bmatrix}
5 \\
-3 \\
1
\end{bmatrix},
\begin{bmatrix}
10 \\
-8 \\
2
\end{bmatrix}
\]
You could calculate all the inner products in the decomposition:

\[
x = \frac{x \cdot u_1}{u_1 \cdot u_1} u_1 + \frac{x \cdot u_2}{u_2 \cdot u_2} u_2 + \frac{x \cdot u_3}{u_3 \cdot u_3} u_3 + \frac{x \cdot u_4}{u_4 \cdot u_4} u_4
\]

in \text{Span}\{u_1, u_2, u_3\} in \text{Span}\{u_4\}

However, once you know the vector in \text{Span}\{u_4\}, the vector in \text{Span}\{u_1, u_2, u_3\} is determined completely by (1). So all you need is

\[
\frac{x \cdot u_4}{u_4 \cdot u_4} u_4 = 50 + 24 - 2 + 0 \over 25 + 9 + 1 + 1 = 2 u_4 = \begin{bmatrix} 10 \\ -6 \\ -2 \\ 2 \end{bmatrix}
\]

The vector in \text{Span}\{u_1, u_2, u_3\} is \(x - 2u_4 = \begin{bmatrix} 10 \\ -8 \\ -6 \\ -2 \\ 2 \\ -2 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \\ -2 \\ 4 \\ 0 \\ -2 \end{bmatrix}.

**Study Tip:** One way to check whether \(\text{proj}_W y\) is computed correctly is to verify that \(y - \text{proj}_W y\) is orthogonal to each vector in the orthogonal basis \(\{u_1, ..., u_p\}\) for \(W\). A faster check that will catch most errors (but not all) is to verify that \(y - \text{proj}_W y\) is orthogonal to \(\text{proj}_W y\).

7. \[
y = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix},
\begin{bmatrix} 1 \\ 3 \\ -2 \end{bmatrix},
\begin{bmatrix} 5 \\ 1 \\ 4 \end{bmatrix}
\]
First, make sure that \(\{u_1, u_2\}\) is an orthogonal basis for \(\text{Span}\{u_1, u_2\}\). This is easy, since \(u_1\) and \(u_2\) are nonzero and \(u_1 \cdot u_2 = 0\). Next, by the Orthogonal Decomposition Theorem, \(y\) is the sum of \(\text{proj}_W y\) and \(y - \text{proj}_W y\), where \(W = \text{Span}\{u_1, u_2\}\).
6.3 • Orthogonal Projections

\[ \text{proj}_W y = \frac{y \cdot u_1}{u_1 \cdot u_1} u_1 + \frac{y \cdot u_2}{u_2 \cdot u_2} u_2 = \frac{1 + 9 - 10}{1 + 9 + 4} u_1 + \frac{5 + 3 + 20}{25 + 1 + 16} u_2 \]

\[ = 0 u_1 + \frac{2}{3} u_2 \begin{bmatrix} 10/3 \\ 2/3 \\ 8/3 \end{bmatrix} \]

and

\[ y - \text{proj}_W y = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix} - \begin{bmatrix} 10/3 \\ 2/3 \\ 8/3 \end{bmatrix} = \begin{bmatrix} -7/3 \\ 7/3 \\ 7/3 \end{bmatrix} \]

As a check, scale \( y - \text{proj}_W y = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \), and observe that the scaled vector is obviously orthogonal to \( u_1 \) and \( u_2 \). Thus \( y - \text{proj}_W y \) is in \( W^\perp \), as it should be.

**Warning:** The formula for \( \text{proj}_W y \) applies only if \( \{u_1, \ldots, u_p\} \) is an orthogonal basis for \( W \). That's why you should check orthogonality, as in Exercise 7, if you are not sure that the basis is orthogonal. If an orthogonal basis is not available, then other methods can be used to compute \( \hat{y} \). (See Exercise 23 in Section 6.5, for example.)

13. \( z = \begin{bmatrix} 3 \\ -7 \\ 2 \end{bmatrix}, v_1 = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}, v_2 = \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} \). Note that \( v_1 \) and \( v_2 \) are orthogonal. By the Best Approximation Theorem, the closest point in \( \text{Span}\{v_1, v_2\} \) to \( z \) is the orthogonal projection \( \hat{z} \), where

\[ \hat{z} = \frac{z \cdot v_1}{v_1 \cdot v_1} v_1 + \frac{z \cdot v_2}{v_2 \cdot v_2} v_2 = \frac{10}{15} v_1 + \frac{2}{3} v_2 = \begin{bmatrix} 2 \\ -1 \\ -3 \end{bmatrix} \begin{bmatrix} 1 \\ 7 \\ 3 \end{bmatrix} = \begin{bmatrix} -1 \\ -3 \end{bmatrix} \]

\[ \begin{bmatrix} 2 \\ -1 \\ -3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ -3 \end{bmatrix} \]

Check: \( z - \hat{z} = \begin{bmatrix} -4 \\ 4 \\ 0 \end{bmatrix} \). The vector \( \begin{bmatrix} 4 \\ -4 \\ 0 \end{bmatrix} \) is orthogonal to both \( v_1 \) and \( v_2 \).
19. By the Orthogonal Decomposition Theorem, \( \mathbf{u}_3 \) is the sum of a vector in \( W = \text{Span}\{\mathbf{u}_1, \mathbf{u}_2\} \) and a vector \( \mathbf{v} \) orthogonal to \( W \). First,

\[
\text{proj}_{W} \mathbf{u}_3 = \frac{-2}{6} \mathbf{u}_1 + \frac{2}{30} \mathbf{u}_2 = \begin{bmatrix} -2/6 \\ -2/6 \\ 4/6 \\ 4/6 \end{bmatrix} + \begin{bmatrix} 10/30 \\ -2/30 \\ 4/30 \\ 4/30 \end{bmatrix} = \begin{bmatrix} -2/5 \\ 0 \\ 4/5 \\ 4/5 \end{bmatrix}
\]

Then

\[
\mathbf{v} = \mathbf{u}_3 - \text{proj}_{W} \mathbf{u}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} -2/5 \\ 0 \\ 4/5 \\ 4/5 \end{bmatrix} = \begin{bmatrix} 2/5 \\ 0 \\ 0 \\ 1 \end{bmatrix}
\]

Not only is \( \mathbf{v} \) orthogonal to \( W \), but also any multiple of \( \mathbf{v} \) is in \( W^\perp \).

**Study Tip:** It would be a good idea to try Exercise 20 and compare the result with Exercise 19. Then think about the following problem: Suppose that \( \{\mathbf{u}_1, \mathbf{u}_2\} \) is an orthogonal set of nonzero vectors in \( \mathbb{R}^3 \). How would you find an orthogonal basis of \( \mathbb{R}^3 \) that contains \( \mathbf{u}_1 \) and \( \mathbf{u}_2 \)? You might discuss this with your instructor.

21. a. See the calculations for \( z_2 \) in Example 1 or the box after Example 6 in Section 6.1.

b. See the Orthogonal Decomposition Theorem.

c. See the the second paragraph after the statement of Theorem 9.

d. See the box before The Best Approximation Theorem.

e. See the paragraph after Theorem 10.

23. By the Orthogonal Decomposition Theorem, each \( \mathbf{x} \) in \( \mathbb{R}^n \) can be written uniquely as \( \mathbf{x} = \mathbf{p} + \mathbf{u} \), with \( \mathbf{p} \) in Row \( A \) and \( \mathbf{u} \) in (Row \( A \))^\perp. By Theorem 3 in Section 6.1, \( \mathbf{u} \) is in Nul \( A \).

Next, suppose that \( A\mathbf{x} = \mathbf{b} \) is consistent. Let \( \mathbf{x} \) be a solution, and write \( \mathbf{x} = \mathbf{p} + \mathbf{u} \), as above. Then

\[
A\mathbf{p} = A(\mathbf{x} - \mathbf{u}) = A\mathbf{x} - A\mathbf{u} = \mathbf{b} - 0 = \mathbf{b}.
\]

So the equation \( A\mathbf{x} = \mathbf{b} \) has at least one solution \( \mathbf{p} \) in Row \( A \).

Finally, suppose that \( \mathbf{p} \) and \( \mathbf{p}_1 \) are both in Row \( A \) and satisfy \( A\mathbf{x} = \mathbf{b} \). Then \( \mathbf{p} - \mathbf{p}_1 \) is in Nul \( A \) because

\[
A(\mathbf{p} - \mathbf{p}_1) = A\mathbf{p} - A\mathbf{p}_1 = \mathbf{b} - \mathbf{b} = 0
\]

The equations \( \mathbf{p} = \mathbf{p}_1 + (\mathbf{p} - \mathbf{p}_1) \) and \( \mathbf{p} = \mathbf{p} + 0 \) both decompose \( \mathbf{p} \) as the sum of a vector in Row \( A \) and a vector in (Row \( A \))^\perp. By the uniqueness of the orthogonal decomposition (Theorem 8), \( \mathbf{p}_1 = \mathbf{p} \), so \( \mathbf{p} \) is unique.

25. From Exercise 36 of Section 6.2, \( U \) should have orthonormal columns, because \( U \) is formed by normalizing the columns of the matrix \( A \) in Exercise 35 whose columns are orthogonal. Verify this by computing \( U^TU \). The result should be the \( 4 \times 4 \) identity matrix.

The closest point to \( \mathbf{y} \) in Col \( U \) is the orthogonal projection of \( \mathbf{y} \) onto Col \( U \). By Theorem 10, this closest point is \( UU^T\mathbf{y} \). The MATLAB command is \( U*U'*\mathbf{y} \). The result of this computation should be the (column) vector \( (1.2, .4, 1.2, 1.2, .4, 1.2, .4, .4) \).
Warning:  I had to work hard to make the arithmetic simple in the exercises for this section, to avoid distractions for you and to save you time. You might not be so lucky on an exam. Even if a problem is designed to be numerically simple, there is always a chance that a minor error will make the calculations messy. In such a case, don’t despair. Carry out the arithmetic as best you can, showing the details of your work (patterned after the solutions in this Study Guide). Chances are that you will get substantial credit for showing that you understand the concepts.

MATLAB  Orthogonal Projections

The orthogonal projection of \( y \) onto a single vector was described in the MATLAB note for Section 6.2. The orthogonal projection onto the set spanned by an orthogonal set of vectors is the sum of the one-dimensional projections. Another way to construct this projection is to normalize the orthogonal vectors, place them in the columns of a matrix \( U \), and use Theorem 10. For instance, if \( \{y_1, y_2, y_3\} \) is an orthogonal set of nonzero vectors, then the matrix

\[
U = \begin{bmatrix}
y_1/\text{norm}(y_1) & y_2/\text{norm}(y_2) & y_3/\text{norm}(y_3) \\
\end{bmatrix}
\]

has orthonormal columns, and \( U^* (U^* y) \) produces the orthogonal projection of \( y \) onto the subspace spanned by \( \{y_1, y_2, y_3\} \). (The parentheses around \( U^* y \) speeds up the computation of \( U^* U^* y \) by avoiding a matrix-matrix product.)

6.4  THE GRAM-SCHMIDT PROCESS  

This section has a nice geometric appeal. The Gram-Schmidt process is well-liked by students and faculty because it is easily learned. Although the process is seldom used in practical computations, it has important generalizations to spaces other than \( \mathbb{R}^n \) (to be discussed briefly in Section 6.7).

KEY IDEAS

When the Gram-Schmidt process is applied to \( \{x_1, \ldots, x_p\} \), the first step is to set \( v_1 = x_1 \). For \( k = 2, \ldots, n \), the \( k \)th step consists of subtracting from \( x_k \) its projection onto the subspace spanned by the previous \( x \)’s. At each step the projection is easy to compute because an orthogonal basis for the appropriate subspace has already been constructed.

The QR factorization of a matrix \( A \) encapsulates the result of applying the Gram-Schmidt process to the columns of \( A \), just as the LU factorization of a matrix encodes the row operations that reduce a matrix to echelon form. Also, just as the LU factorization can be implemented via multiplication by elementary matrices, so can the QR factorization be constructed via multiplication by orthogonal matrices.
SOLUTIONS TO EXERCISES

1. \( \mathbf{x}_1 = \begin{bmatrix} 3 \\ 0 \\ -1 \end{bmatrix}, \mathbf{x}_2 = \begin{bmatrix} 8 \\ 5 \\ -6 \end{bmatrix} \).

Set \( \mathbf{v}_1 = \mathbf{x}_1 \) and compute

\[
\mathbf{v}_2 = \mathbf{x}_2 - \frac{\mathbf{x}_2 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 = \begin{bmatrix} 8 \\ 5 \\ -6 \end{bmatrix} - \frac{30}{10} \begin{bmatrix} 3 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 5 \\ -3 \end{bmatrix}
\]

Check: \( \mathbf{v}_2 \cdot \mathbf{v}_1 = -3 + 0 + 3 = 0 \). So an orthogonal basis is \( \left\{ \begin{bmatrix} 3 \\ 0 \\ -1 \end{bmatrix}, \begin{bmatrix} 0 \\ 5 \\ -3 \end{bmatrix} \right\} \).

7. \( \mathbf{x}_1 = \begin{bmatrix} 2 \\ -5 \\ 1 \end{bmatrix}, \mathbf{x}_2 = \begin{bmatrix} 4 \\ -1 \\ 2 \end{bmatrix} \).

From Exercise 3, use \( \mathbf{v}_1 = \begin{bmatrix} 2 \\ -5 \\ 1 \end{bmatrix} \) and \( \mathbf{v}_2 = \begin{bmatrix} 3/2 \\ 1/2 \end{bmatrix} \) as an orthogonal basis for \( W = \text{Span}\{\mathbf{x}_1, \mathbf{x}_2\} \). Scale \( \mathbf{v}_2 \) to \( (2, 1, 1) \) before normalizing, and then obtain

\[
\mathbf{u}_1 = \frac{1}{\sqrt{30}} \begin{bmatrix} 2 \\ -5 \\ 1 \end{bmatrix} = \frac{2}{\sqrt{30}} \begin{bmatrix} 1/30 \\ -5/30 \\ 1/30 \end{bmatrix}, \quad \mathbf{u}_2 = \frac{1}{\sqrt{6}} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \frac{2}{\sqrt{6}} \begin{bmatrix} 1/6 \\ 1/6 \end{bmatrix}
\]

Study Tip: If you need to normalize a vector by hand, first consider scaling the entries in the vector to make them small integers, if possible.

13. \( \mathbf{A} = \begin{bmatrix} 5 & 9 \\ 1 & 7 \\ -3 & -5 \\ 1 & 5 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 5/6 & -1/6 \\ 1/6 & 5/6 \\ -3/6 & 1/6 \\ 1/6 & 3/6 \end{bmatrix} \).

Let

\[
\mathbf{R} = \mathbf{Q}^\top \mathbf{A} = \begin{bmatrix} 5/6 & 1/6 & -3/6 & 1/6 \\ -1/6 & 5/6 & 1/6 & 3/6 \end{bmatrix} \begin{bmatrix} 5 & 9 \\ 1 & 7 \\ -3 & -5 \\ 1 & 5 \end{bmatrix} = \begin{bmatrix} 36/6 & 72/6 \\ 0 & 36/6 \end{bmatrix} = \begin{bmatrix} 6 & 12 \\ 0 & 6 \end{bmatrix}
\]

As a check, compute \( \mathbf{QR} = \begin{bmatrix} 5/6 & -1/6 \\ 1/6 & 5/6 \\ -3/6 & 1/6 \\ 1/6 & 3/6 \end{bmatrix} \begin{bmatrix} 6 & 12 \\ 0 & 6 \end{bmatrix} = \begin{bmatrix} 5 & 54/6 \\ 1 & 42/6 \\ -3 & -30/6 \\ 1 & 30/6 \end{bmatrix} = \mathbf{A} \).
Remark: The reason the $R$ in Exercise 13 works is that the columns of $Q$ form an orthonormal basis for $\text{Col } A$ (since they were obtained by the Gram-Schmidt process). Thus $QQ^TA = y$ for all $y$ in $\text{Col } A$, by Theorem 10 in Section 6.3. In particular, $QQ^TA = A$. So if $R$ is $Q^TA$, then $QR = Q(Q^TA) = A$.

17. a. See the remark after Example 5 in Section 6.2, and the reference there to Exercise 32.
   b. See (1) in the statement of Theorem 11.
   c. See the solution of Example 4.

19. The full solution is in the text.

21. The solution in the text is complete, except for the details of extending an orthonormal basis for $\text{Span}\{q_1, \ldots, q_n\}$ to an orthonormal basis for $\mathbb{R}^m$. Here is one algorithm. Let $\{e_1, \ldots, e_m\}$ be the standard basis for $\mathbb{R}^m$. Let $f_1$ be the first vector in this basis that is not in $W_n = \text{Span}\{q_1, \ldots, q_n\}$, and let $u_1 = f_1 - \text{proj}_{W_n} f_1$. Then $\{q_1, \ldots, q_n, u_1\}$ is an orthogonal basis for $W_{n+1} = \text{Span}\{q_1, \ldots, q_n, u_1\}$. Let $f_2$ be the first vector in $\{e_1, \ldots, e_m\}$ that is not in $W_{n+1}$.
   (Of course $f_2$ occurs later than $f_1$ in the list $e_1, \ldots, e_m$.) Form $u_2 = f_2 - \text{proj}_{W_{n+1}} f_2$ and $W_{n+2} = \text{Span}\{q_1, \ldots, q_n, u_1, u_2\}$. This process will continue until $m - n$ vectors have been added to the original $n$ vectors. Normalizing the new vectors produces an orthonormal basis for $\mathbb{R}^m$.

23. Use the definition of matrix multiplication. When $A = QR$, the first $p$ columns of $A$ are determined by the action of $Q$ on the first $p$ columns of $R$. So, if $A = [A_1 \quad A_2]$, make the same column-partition of $R$ as $[R_1 \quad R_2]$, where $R_1$ has $p$ columns. Then

\[ A = QR = [QR_1 \quad QR_2] = [A_1 \quad A_2] \]

Is $QR$ a QR factorization of $A_1$? Unfortunately, no. The second factor in a QR factorization should be square and upper triangular with positive entries on the diagonal. Since $R$ has those properties, its first $p$ columns have zeros in rows $p + 1$ to $n$. (This is a key observation.) So, partition $R_1$ into two blocks, $R_1 = \begin{bmatrix} R_{11} \\ 0 \end{bmatrix}$, where $R_{11}$ is square and upper triangular. The entries on the diagonal of $R_{11}$ are positive because they come from $R$. Then $A_1 = QR_1 = Q \begin{bmatrix} R_{11} \\ 0 \end{bmatrix}$. You might consider left-multiplying $R_{11}$ by $Q$, but partitioned matrix multiplication does not work that way. $QR_{11}$ is not defined, because $Q$ has more columns than $R_{11}$ has rows. (Why?)

The final idea needed is to view the product $QR_1$ as a product of block matrices, by partitioning the columns of $Q$ to match the row partition of $R_1$. Write $Q = [Q_1 \quad Q_2]$, where $Q_1$ consists of the first $p$ columns of $Q$. The matrix $Q_1$ has orthonormal columns, because the columns come from $Q$ and so the columns are unit vectors and are pairwise orthogonal. Finally,

\[ A_1 = QR_1 = [Q_1 \quad Q_2] \begin{bmatrix} R_{11} \\ 0 \end{bmatrix} = Q_1 R_{11} + Q_2 0 = Q_1 R_{11} \]
This concludes the construction, because the properties of \( Q_1 \) and \( R_{11} \) have already been discussed. This solution has followed a path that a good student might find. Now that you see how things fit together, you should be able to write a short proof that begins with appropriate partitions of \( Q \) and \( R \). I encourage you to try it. The shorter proof is hidden at the end of the solutions for Section 6.5.

**MATLAB The Gram-Schmidt Process**

If \( A \) has only two columns, then the Gram-Schmidt process is

\[
\begin{align*}
v_1 &= A(:,1) \\
v_2 &= A(:,2) - (A(:,2)'*v_1)/(v_1' * v_1)*v_1
\end{align*}
\]

If \( A \) has three columns, add the command

\[
v_3 = A(:,3) - (A(:,3)'*v_1)/(v_1'*v_1)*v_1 - (A(:,3)'*v_2)/(v_2'*v_2)*v_2
\]

You should use these commands for a while, to learn the general procedure. After that, you can use the Laydata5 command `proj(x,V)`, which computes the projection of a vector \( x \) onto the subspace spanned by the columns of a matrix (or vector) \( V \). For example,

\[
\begin{align*}
v_2 &= A(:,2) - \text{proj}(A(:,2),v_1) & \text{Here } v_1 & \text{ is } V \\
v_3 &= A(:,3) - \text{proj}(A(:,3),[v_1 \ v_2]) & \text{Here } [v_1 \ v_2] & \text{ is } V
\end{align*}
\]

The columns of \( v' \) in \( \text{proj}(x,V) \) need not be orthogonal for the command to work, but if they are, the entries in \( \text{proj}(x,V) \) will usually agree with those computed via Theorem 10 in Section 6.3, to twelve or more decimal places. Enter `help proj` to learn more about `proj`.

To check your work or to save time, enter \( Q = \text{gs}(A) \), which uses the Gram-Schmidt process to construct the columns of \( Q \). See `help gs`.

Although you should construct the QR factorization of a matrix using the approach in the text, you might like to see what MATLAB does. The command \( [Q_1 R_1] = \text{qr}(A) \) creates a modified QR factorization of an \( m \times n \) matrix \( A \) as described in Exercise 21 in the text. If rank \( A = r \), then the first \( r \) rows of \( R_1 \) are nonzero and the first \( r \) columns of \( Q_1 \) form an orthonormal basis for \( \text{Col } A \).
6.5 LEAST-SQUARES PROBLEMS

The basic geometric principles in this section provide the foundation for all the applications in Sections 6.6–6.8.

KEY IDEAS

A least-squares solution of $Ax = b$ is any vector $x$ that makes $Ax$ as close as possible to $b$. (Learn the formal definition, too.) If the columns of $A$ are linearly dependent, then there are many least-squares solutions of $Ax = b$. They can all be found by row reducing the augmented matrix for the normal equations $A^T A x = A^T b$ (Theorem 13).

If the columns of $A$ are linearly independent, then there is only one least-squares solution. To find it, solve the normal equations or compute $(A^T A)^{-1} A^T b$ (Theorem 14). If a QR factorization of $A$ is available, say $A = QR$, solve the equation $Rx = Q^T b$ (Theorem 15 and the Numerical Note).

STUDY NOTES

The material up to and including Figure 2 needs to be read carefully several times, so you understand what the term “least-squares solution” means. Be careful to distinguish between $\hat{x}$ and $b$. A common mistake is to think that $\hat{x}$ itself somehow has the least-squares norm or is the closest point to $b$. Look at Fig. 2 again. The vector closest to $b$ is $A\hat{x}$, not $\hat{x}$.

Theorem 13 provides a common way to find least-squares solutions. One way to remember the normal equations is to observe that they look the same as $Ax = b$ with $A^T$ left-multiplied on each side of the equation. It is completely wrong, however, to try to derive the normal equations from $Ax = b$ via left-multiplication by $A^T$. If the equation $Ax = b$ has no solution, then the equation itself is a false statement about every vector $x$. Matrix algebra on such a false statement is meaningless.

SOLUTIONS TO EXERCISES

1. $A = \begin{bmatrix} -1 & 2 \\ 2 & -3 \\ -1 & 3 \end{bmatrix}$, $b = \begin{bmatrix} 4 \\ 1 \\ 2 \end{bmatrix}$

$A^T A = \begin{bmatrix} -1 & 2 & -1 \\ 2 & -3 & 3 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} -1 \\ 2 \\ -3 \\ -1 \\ 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 6 & -11 \\ -11 & 22 \end{bmatrix}$, $A^T b = \begin{bmatrix} 4 \\ 1 \\ 2 \end{bmatrix}$

a. The normal equations: $\begin{bmatrix} 6 & -11 \\ -11 & 22 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -4 \\ 11 \end{bmatrix}$

b. Since $A^T A$ is only $2 \times 2$, $(A^T A)^{-1}$ is easy to compute, and
\[ \hat{x} = \begin{bmatrix} 6 & -11 \\ -11 & 22 \end{bmatrix}^{-1} \begin{bmatrix} -4 \\ 11 \end{bmatrix} = \frac{1}{11} \begin{bmatrix} 22 & 11 \\ 11 & 6 \end{bmatrix} \begin{bmatrix} -4 \\ 11 \end{bmatrix} = \frac{1}{11} \begin{bmatrix} 33 \\ 22 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix} \]

**Warning:** It is important to distinguish between the normal equations \( A^T A \hat{x} = A^T b \) and the formula \( \hat{x} = (A^T A)^{-1} A^T b \). Both equations describe \( \hat{x} \) (implicitly or explicitly), but the formula for \( \hat{x} \) holds only when \( A \) has linearly independent columns. Note that the expression \((A^T A)^{-1}A^T\) cannot be simplified when \( A \) is not invertible.

7. \[ A = \begin{bmatrix} 1 & -2 \\ -1 & 2 \\ 0 & 3 \\ 2 & 5 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 1 \\ -4 \\ 2 \end{bmatrix}, \quad A^T A = \begin{bmatrix} 1 & -1 & 0 & 2 \\ -2 & 2 & 3 & 5 \\ 0 & 3 & 2 & 5 \end{bmatrix} = \begin{bmatrix} 6 & 6 \\ 6 & 42 \end{bmatrix} \]

\[ A^T b = \begin{bmatrix} 1 & -1 & 0 & 2 \\ -2 & 2 & 3 & 5 \end{bmatrix} \begin{bmatrix} 3 \\ 1 \\ -4 \\ 2 \end{bmatrix} = \begin{bmatrix} 6 \\ -6 \end{bmatrix} \]

The normal equations: \[ \begin{bmatrix} 6 & 6 \\ 6 & 42 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ -6 \end{bmatrix} \]

The particular numbers in \( A^T A \) suggest that the normal equations might be solved easily via row operations:

\[ \begin{bmatrix} 6 & 6 & 6 \\ 6 & 42 & -6 \end{bmatrix} \sim \begin{bmatrix} 6 & 6 & 6 \\ 0 & 36 & -12 \end{bmatrix} \sim \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & -1/3 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 4/3 \\ 0 & 1 & -1/3 \end{bmatrix} \]

Thus \( \hat{x} = \begin{bmatrix} 4/3 \\ -1/3 \end{bmatrix} \). The least-squares error is \( \| A\hat{x} - b \| \), so compute

\[ A\hat{x} - b = \begin{bmatrix} 1 & -2 \\ -1 & 2 \\ 0 & 3 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} 4/3 \\ -1/3 \\ -4 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ 3 \\ -1 \end{bmatrix} \]

\[ \| A\hat{x} - b \|^2 = 1 + 9 + 9 + 1 = 20, \text{ and } \| A\hat{x} - b \| = \sqrt{20} = 2\sqrt{5} \]

**Study Tip:** A good way to check your work in Exercises 1–8 is to verify, that \( A\hat{x} - b \) is orthogonal to each column of \( A \).

**Warning:** The matrices in Exercises 9–12 are special—their columns are orthogonal. That is why these exercises are not difficult. See Example 4. In general, if the columns of \( A \) are not orthogonal, finding the orthogonal projection of \( b \) onto Col(\( A \)) takes more work. See Exercise 23.
13. \[ Au = \begin{bmatrix} 3 & 4 \\ -2 & 1 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 5 \\ 11 \\ 11 \end{bmatrix} = \begin{bmatrix} 11 \\ -9 \\ 5 \end{bmatrix} = b - Au \begin{bmatrix} 11 \\ -11 \\ 11 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ -6 \end{bmatrix}, \| b - Au \| = \sqrt{40} \]

\[ Av = \begin{bmatrix} 3 & 4 \\ -2 & 1 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 5 \\ 7 \\ 7 \\ -2 \end{bmatrix} = \begin{bmatrix} 7 \\ -12 \\ 7 \end{bmatrix} = b - Av \begin{bmatrix} 11 \\ -9 \\ 5 \end{bmatrix} = \begin{bmatrix} 7 \\ -12 \\ 7 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \\ -2 \end{bmatrix}, \| b - Av \| = \sqrt{29} \]

Obviously, \( Au \) is not the closest point of \( \text{Col} \ A \) to \( b \), because \( Av \) is closer. Hence \( u \) is not the least-squares solution of \( Ax = b \).

17. a. See the beginning of the section.
   b. See the comments about equation (1).
   c. Read the definition of a least-squares solution.
   d. See Theorem 13.
   e. See Theorem 14.

19. The full solution is in the text.

21. a. If \( A \) has linearly independent columns, then the equation \( Ax = 0 \) has only the trivial solution. By Exercise 17, \( A^T Ax = 0 \) also has only the trivial solution. Since \( A^T A \) is square, it must be invertible, by the Invertible Matrix Theorem.
   b. Since the \( n \) linearly independent columns of \( A \) belong to \( \mathbb{R}^m \), \( m \) could not be less than \( n \).
   c. The \( n \) linearly independent columns of \( A \) form a basis for \( \text{Col} \ A \), so the rank of \( A \) is \( n \).

23. (This solution is for Section 6.4.) Given \( A = QR \), partition \( A = [A_1 \quad A_2] \), where \( A_1 \) has \( p \) columns. Consider the following partitions of \( Q \) and \( R \), which are possible because \( R \) is upper triangular:

\[ A = \begin{bmatrix} A_1 & A_2 \end{bmatrix} = QR = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix} = \begin{bmatrix} Q_1R_{11} & Q_1R_{12} + Q_2R_{22} \end{bmatrix} \]

where \( Q_1 \) has \( p \) columns and \( R_{11} \) is a \( p \times p \) matrix. Then \( A_1 = Q_1R_{11} \). The matrix \( Q_1 \) has orthonormal columns, because the columns come from \( Q \) and so the columns are unit vectors and pairwise orthogonal. The matrix \( R_{11} \) is square and upper triangular, because of its position inside \( R \). The diagonal entries of \( R_{11} \) are positive because they are diagonal entries of \( R \). Thus \( Q_1R_{11} \) is a QR factorization of \( A_1 \).
MATLAB The Backslash Command

When $A$ has linearly dependent columns, you can write the general description of all least-squares solutions on paper after you row reduce the augmented matrix for the normal equations: \texttt{ref}([\texttt{A’*A} \ A’*\texttt{b}]). When $A$ has linearly independent columns, enter the MATLAB “backslash” command

$$x = (A’*A) \backslash (A’*b)$$

to solve the normal equations. You can also enter \texttt{inv}(A’*A)*(A’*b) ; or use \texttt{ref}, as above. For Exercises 15 and 16, see the Numerical Note in Section 6.5 in the text and use the backslash command $R \backslash (Q’*b)$ to solve $Rx = Q’b$.

When is $A$ is not square but has linearly independent columns, this procedure of first forming $Q$ and $R$ and then solving $Rx = Q’b$ is exactly what MATLAB does (internally) when the MATLAB command $A \backslash b$ is used to solve $Ax = b$. You should use the normal equations or QR for computations here, instead of just using the backslash. This will give you a solid conceptual background for applying least-squares techniques later in your career. For more about the backslash, see the MATLAB box for Section 2.5.

For Exercise 26, the command $A = [A1; A2]$ creates a (partitioned) matrix whose top block is $A1$ and bottom block is $A2$. Of course, $A1$ and $A2$ must have the same number of columns.

6.6 APPLICATIONS TO LINEAR MODELS

This section of the text will be a valuable reference for any person who works with data that requires statistical analysis. Many graduate fields require such work, often in connection with doctoral research. Even most undergraduates will take a course where least-squares lines are used.

KEY IDEA

Linear algebra unifies the study of many problems in statistics and data analysis. All the examples in this section, from ordinary linear regression (using a least-squares line) to multiple regression, concern just one idea: find a least-squares solution of $X\beta = y$. Only the design matrix $X$ varies. The exercises help you practice choosing $X$. The least-squares solution $\hat{\beta}$ always satisfies the normal equations $X^T X \hat{\beta} = X^T y$. 

Copyright © 2016 Pearson Education, Inc.
STUDY NOTES

Don’t confuse the least-squares line in Fig.1 with the lines and planes in Section 6.5 onto which we projected various vectors \( b \). The line is nothing more than a special case of the curves in Figures 2–5. In each case, the “linearity” of the model lies not in the curve, but rather in the fact that the unknown parameters (or weights) \( \beta_0, \beta_1, \ldots \) occur linearly in the formula for the curve, just as the variables \( x_1, x_2, \ldots \) occur in an ordinary linear equation.

Any \( 4 \times 4 \) submatrix of the design matrix in Example 3 is called a Vandermonde matrix. Using Exercise 11 on page 160, one can show that if at least four of the values \( x_1, \ldots, x_n \) are distinct, then the least-squares solution \( \hat{\beta} \) will be unique, by Theorem 14 in Section 6.5.

FURTHER READING

An important generalization of the discussion here is to multivariate analysis, which involves several \( y \) vectors rather than just one. In this case the basic equation is \( XB = Y \), where each column of \( Y \) is a data set for one dependent variable, and each column of \( B \) is a set of parameters to be determined. That is, \( X[\beta_1 \cdots \beta_p] = [y_1 \cdots y_p] \). For more information, see the classic text by T. W. Anderson, *An Introduction to Multivariate Statistical Analysis*, John Wiley & Sons, New York, 1984 (and 1958). The preface of the text says, “A knowledge of matrix algebra is a prerequisite [for understanding the text].” Most modern multivariate statistics texts rely heavily on matrix notation and matrix algebra.

SOLUTIONS TO EXERCISES

1. Place the \( x \)-coordinates of the data in the second column of \( X \) and the \( y \)-coordinates in the vector \( y \). So \( X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix} \) and \( y = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \). Compute

\[
\begin{bmatrix}
1 & 1 & 1 & 1 \\
0 & 1 & 2 & 3
\end{bmatrix}
\begin{bmatrix}
1 & 0 \\
1 & 1 \\
1 & 2 \\
1 & 3
\end{bmatrix}
= \begin{bmatrix}
4 & 6 \\
6 & 14
\end{bmatrix}
\begin{bmatrix}
1 \\
0 & 1 & 2 & 3
\end{bmatrix}
= \begin{bmatrix}
1 \\
1 & 1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
6 \\
11
\end{bmatrix}
\]

\[
X^T X = \begin{bmatrix} 4 & 6 & 1 & 1 \\ 6 & 14 & 0 & 1 \\ 1 & 1 & 6 & 2 \\ 1 & 3 & 2 & 3 \end{bmatrix}
\begin{bmatrix}
1 \\
1 \\
1 \\
3
\end{bmatrix}
= \begin{bmatrix}
1 & 6 \\
6 & 11
\end{bmatrix}
\]

The matrix normal equation and its solution are:

\[
\begin{bmatrix} 4 & 6 \\ 6 & 14 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 6 \\ 11 \end{bmatrix}
\]

\[
\begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 4 & 6 \end{bmatrix}^{-1} \begin{bmatrix} 6 \\ 11 \end{bmatrix}
= \frac{1}{20} \begin{bmatrix} 14 & -6 \\ 4 & 11 \end{bmatrix} \begin{bmatrix} 6 \\ 18 \end{bmatrix}
= \begin{bmatrix} 18/20 \\ 8/20 \end{bmatrix}
= \begin{bmatrix} .9 \\ .4 \end{bmatrix}
\]
The least-squares line, \( y = \beta_0 + \beta_1x \), is \( y = .9 + .4x \).

7. a. \( y = X\hat{\beta} + \varepsilon \), where \( y = \begin{bmatrix} 1.8 \\ 2.7 \\ 3.4 \\ 3.8 \\ 3.9 \end{bmatrix}, X = \begin{bmatrix} 1 & 1 \\ 2 & 4 \\ 3 & 9 \\ 4 & 16 \\ 5 & 25 \end{bmatrix}, \hat{\beta} = \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \end{bmatrix}, \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} \).

b. In this problem, \( X^TX \) is invertible. You can use your matrix program to solve the normal equations without explicitly computing the entries in \( X^TX \). For details, see the MATLAB box below or the corresponding box in the appendix for your matrix program. In any case,

\[
\hat{\beta} = \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \end{bmatrix} = \begin{bmatrix} 1.76 \\ -0.20 \end{bmatrix} \text{ (to two decimal places)}
\]

The desired least-squares equation is \( y = 1.76x - 0.20x^2 \).

13. Let \( 1 \) be the vector in \( \mathbb{R}^3 \) with 1 in each entry, let \( t = (0, \ldots, 12) \), and for \( k = 2 \) and 3, let \( t.^k \) denote the vector whose entries are the \( k \)th powers of the entries in \( t \). (See the MATLAB box below.) Then the design matrix is:

\[
X = \begin{bmatrix} 1 & t & t.^2 & t.^3 \end{bmatrix}
\]

The observation vector \( y \) lists the measured positions of the plane.

a. Numerical solution of the normal equations yields

\[
\hat{\beta} = (-.8558, 4.7025, 5.5554, -.0274)
\]

The least-squares polynomial (position of the plane at time \( t \)) is

\[
y = -.8558 + 4.7025t + 5.5554t^2 - .0274t^3
\]

b. The velocity is the derivative of the position function:

\[
v(t) = 4.7025 + 11.1108t - .0822t^2
\]

When \( t = 4.5 \) seconds, \( v(4.5) = 53.0 \) ft/sec.

15. From equation (1) on page 369,

\[
X^TX = \begin{bmatrix} 1 & \cdots & 1 \\ x_1 & \cdots & x_n \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} n & \Sigma x \\ \Sigma x & \Sigma x^2 \end{bmatrix}
\]

\[
X^Ty = \begin{bmatrix} 1 & \cdots & 1 \\ x_1 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} \Sigma y \\ \Sigma xy \end{bmatrix}
\]

The equations (7) in the text follow immediately from the usual matrix normal equation \( X^TX\hat{\beta} = X^Ty \).
16. **Note:** The formulas you should derive are
\[ \hat{\beta}_0 = \frac{(\Sigma x^2)(\Sigma y) - (\Sigma x)(\Sigma xy)}{n\Sigma x^2 - (\Sigma x)^2}, \quad \hat{\beta}_1 = \frac{n\Sigma xy - (\Sigma x)(\Sigma y)}{n\Sigma x^2 - (\Sigma x)^2} \]

Some statistics texts present other equivalent formulas for \( \hat{\beta}_0 \) and \( \hat{\beta}_1 \).

19. The equation to be proved is \( ||y||^2 = ||X\hat{\beta}||^2 + ||y - X\hat{\beta}||^2 \). This follows from the Pythagorean Theorem (in Section 6.1) and the figure below.

---

**Appendix: The Geometry of a Linear Model**

The column space of the design matrix \( X \) is sometimes called the **design subspace**. If \( \hat{\beta} \) is the least-squares solution of \( y = X\hat{\beta} \), then the residual vector \( \epsilon = y - X\hat{\beta} \), is orthogonal to the design subspace, and the equation \( y = X\hat{\beta} - \epsilon \) is an orthogonal decomposition of the observed \( y \) into the sum of the least-squares predicted \( \hat{y} \) and the residual vector \( \epsilon \).

---

**MATLAB  Least-squares Solutions and Functions of Vectors**

Once you create the design matrix \( X \) and the observation vector \( y \), your computations for least-squares solutions here are the same as those described in the MATLAB box for Section 6.5. Here, \( A \) and \( b \) are replaced by \( X \) and \( y \), respectively. The MATLAB command

\[ \text{ref}([X'*X \ X'*y]) \]

leads to the general description of all least-squares solutions. When \( X \) has linearly independent columns, the command \( (X' *X) \backslash (X' *y) \) creates the least-squares solution. In
subsequent courses, you may choose simply to use \( \mathbf{x} \backslash \mathbf{y} \), which also produces a least-squares solution, except when \( X \) is square and singular (or nearly singular).

To construct the design matrix for an exercise in this section, you may need MATLAB’s ability to compute functions of vectors. If \( \mathbf{x} \) is a vector and \( k \) is a positive integer, then \( \mathbf{x}^\wedge{k} \) is a vector the same size as \( \mathbf{x} \) whose entries are the \( k \)th powers of the entries in \( \mathbf{x} \). The function \( \cos(\mathbf{x})^\wedge{k} \) was mentioned in the MATLAB box for Section 4.3. The exponential function, \( \exp(\mathbf{x}) \), and natural logarithm function, \( \log(\mathbf{x}) \), also act on each entry in \( \mathbf{x} \). The entries in the vector \( \exp(-0.02*\mathbf{x}) \), for example, are computed by applying the function \( e^{-0.02x} \) to the corresponding entries in \( \mathbf{x} \).

6.7 INNER PRODUCT SPACES

Three examples of inner product spaces are described here, in Examples 1, 2, and 7. Corresponding applications appear in the next section. Material in Sections 6.7 and 6.8 will be useful for many careers, particularly science, engineering, and mathematics. If your course does not cover this now, the text and Study Guide can help you learn it later on your own.

KEY IDEAS

The concepts of length and orthogonality in \( \mathbb{R}^n \) have analogues in a number of other vector spaces. The definition of an inner product identifies the basic properties needed for a theory that parallels the familiar theory for \( \mathbb{R}^n \). Two useful facts, the Cauchy-Schwarz inequality and the triangle inequality, were not developed earlier, but they are important for applications both in \( \mathbb{R}^n \) and in other inner product spaces. Every mathematics major will need to know these facts in other undergraduate courses.

The inner product in Example 1 is used in Section 6.8 to describe weighted least-squares problems. The inner product in Examples 2–6 provides a more sophisticated approach to the least-squares curve fitting discussed in Section 6.6. See the “trend analysis” in Section 6.8.

Be sure to read the paragraph preceding Example 6. The idea of “best approximation” to a function is of fundamental importance in mathematics. The most common applications of best approximation (such as Fourier series, introduced in Section 6.8) involve the inner product in Example 7.

SOLUTIONS TO EXERCISES

1. The inner product is \( \langle \mathbf{x}, \mathbf{y} \rangle = 4x_1y_1 + 5x_2y_2 \). Let \( \mathbf{x} = (1, 1) \), \( \mathbf{y} = (5, -1) \).
   a. \( ||\mathbf{x}||^2 = 4 \cdot 1 \cdot 1 + 5 \cdot 1 \cdot 1 = 9 \), \( ||\mathbf{x}|| = 3 \)
   \( ||\mathbf{y}||^2 = 4 \cdot 5 \cdot 5 + 5(-1)(-1) = 105 \), \( ||\mathbf{y}|| = \sqrt{105} \)
   \( ||\langle \mathbf{x}, \mathbf{y} \rangle||^2 = |4 \cdot 1 \cdot 5 + 5 \cdot 1(-1)|^2 = |15|^2 = 225 \)
b. A vector \( z = (z_1, z_2) \) is orthogonal to \( y \) if and only if \( \langle z, y \rangle = 0 \), that is,

\[
4 \cdot z_1 \cdot 5 + 5 \cdot z_2 \cdot (-1) = 0, \quad 20z_1 - 5z_2 = 0, \quad \text{and} \quad z_2 = 4z_1
\]

Thus \( (z_1, z_2) \) is orthogonal to \( y \) if and only if \( z_2 = 4z_1 \).

7. Given \( p(t) = 4 + t \) and \( q(t) = 5 - 4t^2 \). The orthogonal projection \( \hat{q} \) of \( q \) onto the subspace spanned by \( p \) is \( [\langle q, p \rangle / \langle p, p \rangle]p \). The notation of Example 5 organizes the calculations nicely:

The inner product \( \langle q, p \rangle \) equals the (standard) inner product of the two corresponding vectors in \( \mathbb{R}^3 \): \( \langle q, p \rangle = 3 \cdot 1 + 4 \cdot 5 + 5 \cdot 1 = 28 \). Similarly, \( \langle p, p \rangle = 3^2 + 4^2 + 5^2 = 50 \). Thus

\[
\hat{q}(t) = \frac{28}{50}(4 + t) = \frac{56}{25} + \frac{14}{25}t
\]

13. Suppose \( A \) is invertible and \( \langle u, v \rangle = (Au) \cdot (Av) \), for \( u, v \) in \( \mathbb{R}^n \). Note that \( \langle u, v \rangle \) is in \( \mathbb{R} \), and check each axiom in the definition on page 376:

i. \( \langle u, v \rangle = (Au) \cdot (Av) = (Av) \cdot (Au) \quad \text{Property of dot product} \)

\[
= \langle v, u \rangle
\]

ii. \( \langle u + v, w \rangle = [A(u + v)] \cdot (Aw) = [Au + Av] \cdot (Aw) \quad \text{Matrix multiplication} \)

\[
= (Au) \cdot (Aw) + (Av) \cdot (Aw) \quad \text{Property of dot product} \)

\[
= \langle u, w \rangle + \langle v, w \rangle
\]

iii. \( \langle cu, v \rangle = [A(cu)] \cdot (Av) = [c(Au)] \cdot (Av) \quad \text{Matrix multiplication} \)

\[
= c(Au) \cdot (Av) \quad \text{Property of dot product} \)

\[
= c\langle u, v \rangle
\]

iv. \( \langle u, u \rangle = (Au) \cdot (Au) = ||Au||^2 \geq 0 \), and this quantity is zero if and only if the vector \( Au \) is \( 0 \). But \( Au = 0 \) if and only if \( u = 0 \), because \( A \) is invertible.

Another method for verifying the axioms is to use properties of the transpose operation. The calculations are similar. However, for (i), you need to use the fact that the transpose of a scalar (which is a \( 1 \times 1 \) matrix) is the scalar itself: \( \langle u, v \rangle = \langle u, v \rangle^T = [(Au)^T(Av)]^T = (Av)^T(Au)^T = (Av)(Au) = \langle v, u \rangle \).

17. \( \| u + v \|^2 = \langle u + v, u + v \rangle = \langle u, u \rangle + \langle v, v \rangle + \langle u, v \rangle + \langle v, u \rangle \quad \text{Axiom ii} \)

\[
= \langle u + v, u \rangle + \langle u + v, v \rangle \quad \text{Axiom i} \)

\[
= \langle u, u \rangle + \langle v, v \rangle + \langle u, v \rangle + \langle v, u \rangle \quad \text{Axiom ii} \)

\[
= \langle u, u \rangle + 2\langle u, v \rangle + \langle v, v \rangle \quad \text{Axiom i} \)

Copyright © 2016 Pearson Education, Inc.
Next, replace \( v \) by \(-v\) and use the fact that \( u - v = u + (-1)v \).

\[
\| u - v \|^2 = \langle u, u \rangle + 2 \langle u, -v \rangle + \langle -v, -v \rangle \\
= \langle u, u \rangle + 2 \langle u, v \rangle - (-1)^2 \langle v, v \rangle \\
\]

Replacing \( v \) above by \(-v\) and using Axiom iii and Exercise 15,

Subtracting, \( \|u + v\|^2 - \|u - v\|^2 = 2 \langle u, v \rangle - (-2 \langle u, v \rangle) = 4 \langle u, v \rangle \). Division by 4 gives the desired identity.

19. The full solution is in the text.

25. In the space \( C[-1, 1] \) with the integral inner product, the polynomials \( t \) and 1 are orthogonal, because

\[
\langle t, 1 \rangle = \int_{-1}^{1} t \cdot 1 \, dt = \frac{1}{2} t^2 \bigg|_{-1}^{1} = \frac{1}{2} (1)^2 - \frac{1}{2} (-1)^2 = 0
\]

So 1 and \( t \) can be in an orthogonal basis for \( \text{Span}\{1, t, t^2\} \). Next, compute \( \text{proj}_W t^2 \), the orthogonal projection of the vector \( t^2 \) onto the subspace \( W \) spanned by 1 and \( t \).

\[
\langle t^2, 1 \rangle = \int_{-1}^{1} t^2 \cdot 1 \, dt = \frac{1}{3} t^3 \bigg|_{-1}^{1} = \frac{1}{3} (1)^3 - \frac{1}{3} (-1)^3 = \frac{2}{3}
\]

\[
\langle t^2, t \rangle = \int_{-1}^{1} t^2 \cdot t \, dt = \frac{1}{4} t^4 \bigg|_{-1}^{1} = \frac{1}{4} (1)^4 - \frac{1}{4} (-1)^4 = 0
\]

There is no need to compute \( \langle t, t \rangle \), because \( t^2 \) is orthogonal to \( t \). Thus

\[
\text{proj}_W t^2 = \frac{\langle t^2, 1 \rangle}{\langle 1, 1 \rangle} + \frac{\langle t^2, t \rangle}{\langle t, t \rangle} t = \frac{2/3}{2} 1 + 0 = \frac{1}{3}
\]

A polynomial orthogonal to \( W \) is \( t^2 - \text{proj}_W t^2 = t^2 - \frac{1}{3} \). Another choice is this polynomial scaled by 3, namely, \( 3t^2 - 1 \). Thus, the polynomials, 1, \( t^2 \), and \( 3t^2 - 1 \) form an orthogonal basis for \( \text{Span}\{1, t, t^2\} \).

Can you find the next Legendre polynomial, a cubic polynomial that is orthogonal to each of the first three Legendre polynomials?
6.8 APPLICATIONS OF INNER PRODUCT SPACES

Of the three applications in this section, the discussion of Fourier series is by far the most important. Such series have great practical value, particularly in mathematics, engineering and the physical sciences. Calculations with Fourier series are simple because sine and cosine functions are orthogonal. This fact is often overlooked in undergraduate courses that do not assume a linear algebra background.

KEY IDEAS

The text gives the normal equations for the weighted least-squares solution of \( Ax = y \). When applied to a least-squares line problem, the most common situation, the normal equations are usually written as \((WX)^{T}WX\hat{\beta} = (WX)^{T}y\) where \( W \) is the (diagonal) weighting matrix, \( X \) is the design matrix, \( \hat{\beta} \) is the least-squares parameter vector, and \( y \) is the observation vector.

Trend analysis is really a least-squares regression problem of the type described in Section 6.6, with data points \((x_{1}, y_{1}), \ldots, (x_{n}, y_{n})\) fitted by a curve of the form

\[
y = \beta_{0}f_{0}(x) + \beta_{1}f_{1}(x) + \cdots + \beta_{k}f_{k}(x)
\]

where the functions \( f_{0}, \ldots, f_{k} \) are polynomials that are orthogonal with respect to an inner product on \( P_{n-1} \) defined by

\[
\langle p, q \rangle = \int p(x)q(x) \, dx.
\]

Usually, \( x_{1}, \ldots, x_{n} \) are arranged to be evenly spaced and sum to zero, and the functions \( f_{1}, \ldots, f_{k} \) are of degree 3 or 4 or less.

In \( C[0, 2\pi] \) with the integral inner product, the set

\[
\{1, \cos t, \cos 2t, \ldots, \cos nt, \sin t, \sin 2t, \ldots, \sin nt\}
\]

is orthogonal. The \( n \)th order Fourier approximation to some \( f \) in \( C[0, 2\pi] \) is simply the orthogonal projection of \( f \) onto the subspace \( W \) of trigonometric polynomials spanned by the functions in (*).

The Fourier coefficients of \( f \) are the weights in the usual formula for the orthogonal projection of \( f \) onto \( W \).

If an application involves an interval \([0, T]\) instead of \([0, 2\pi]\), then the inner product requires an integral over \([0, T]\), and the appropriate orthogonal set is obtained by replacing \( t \) in each function in (*) with \( 2\pi t/T \).

SOLUTIONS TO EXERCISES

1. For the data \((-2, 0), (-1, 0), (0, 2), (1, 4), (2, 4)\), construct

\[
X = \begin{bmatrix}
1 & -2 \\
1 & -1 \\
1 & 0 \\
1 & 1 \\
1 & 2
\end{bmatrix}
\]

Design matrix, \( \beta = \begin{bmatrix} \beta_{0} \\ \beta_{1} \end{bmatrix} \) Parameter vector, \( y = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 4 \\ 4 \end{bmatrix} \) Observation vector.

Since the first and last data points are about half as reliable as the other points, a suitable weighting matrix is

Copyright © 2016 Pearson Education, Inc.
The remaining calculations are the same as in ordinary least-squares, except that the weighted design matrix $WX$ and the weighted observation vector $Wy$ appear in place of $X$ and $y$, respectively.

The normal equations and solution are

$$
\begin{bmatrix}
14 & 0 \\
0 & 16
\end{bmatrix}
\begin{bmatrix}
\beta_0 \\
\beta_1
\end{bmatrix} =
\begin{bmatrix}
28 \\
24
\end{bmatrix},
\begin{bmatrix}
\beta_0 \\
\beta_1
\end{bmatrix} =
\begin{bmatrix}
1/14 & 0 \\
0 & 1/16
\end{bmatrix}
\begin{bmatrix}
28 \\
24
\end{bmatrix} =
\begin{bmatrix}
2 \\
3/2
\end{bmatrix}
$$

The equation of the least-squares line is $y = 2 + (3/2)x$.

7. $\|\cos kt\|^2 = \int_0^{2\pi} \cos kt \cdot \cos kt \, dt = \int_0^{2\pi} \frac{1 + \cos 2kt}{2} \, dt$

$$
= \left(\frac{1}{2}t + \frac{\sin 2kt}{4k}\right)\bigg|_0^{2\pi} = \left(\frac{1}{2} \cdot 2\pi + 0\right) - 0 = \pi \quad (\text{if } k \neq 0)
$$

$\|\sin kt\|^2 = \int_0^{2\pi} \sin kt \cdot \sin kt \, dt = \int_0^{2\pi} \frac{1 - \cos 2kt}{2} \, dt$

$$
= \left(\frac{1}{2}t + \frac{\sin 2kt}{4k}\right)\bigg|_0^{2\pi} = \left(\frac{1}{2} \cdot 2\pi + 0\right) - 0 = \pi \quad (\text{if } k \neq 0)
$$

9. $f(t) = 2\pi - t$. The definite integrals of $t \cos kt$ and $t \sin kt$, shown below, were computed in Example 4. The Fourier coefficients of $f$ are:
The third-order Fourier approximation to \( f \) is
\[
\pi + 2\sin t + \sin 2t + \frac{2}{3}\sin 3t
\]

13. Take \( f \) and \( g \) in \( C[0, 2\pi] \) and let \( m \) be a nonnegative integer. Then, the linearity of the inner product shows that
\[
\langle (f + g), \cos mt \rangle = \langle f, \cos mt \rangle + \langle g, \cos mt \rangle
\]

Dividing each term in this equality by \( \langle \cos mt, \cos mt \rangle \), we conclude that the Fourier coefficient \( a_m \) of \( f + g \) is the sum of the corresponding Fourier coefficients of \( f \) and of \( g \). Similarly, the Fourier coefficient \( b_m \) of \( f + g \) is the sum of the corresponding Fourier coefficients of \( f \) and of \( g \).

---

**Appendix: The Linearity of an Orthogonal Projection**

The argument for Exercise 13 is a special case of a general principle. In any inner product space, the mapping \( y \mapsto \frac{\langle y, u \rangle}{\langle u, u \rangle}u \) is linear, for any nonzero \( u \). To verify this, take any \( x \) and \( y \) in the space and any scalar \( c \). Then
\[
\frac{\langle x + y, u \rangle}{\langle u, u \rangle}u = \frac{\langle x, u \rangle}{\langle u, u \rangle}u + \frac{\langle y, u \rangle}{\langle u, u \rangle}u, \text{ and } \frac{c\langle x, u \rangle}{\langle u, u \rangle}u = c\frac{\langle x, u \rangle}{\langle u, u \rangle}u
\]

Similarly, if \( u_1, \ldots, u_p \) are any nonzero vectors, then the mapping
\[
y \mapsto \frac{\langle y, u_1 \rangle}{\langle u_1, u_1 \rangle}u_1 + \cdots + \frac{\langle y, u_p \rangle}{\langle u_p, u_p \rangle}u_p
\]
is a linear transformation. Thus, if \( \{u_1, \ldots, u_p\} \) is an orthogonal basis for a subspace \( W \), then the mapping \( y \mapsto \text{proj}_W y \) is a linear transformation.

In particular, if \( W \) is the vector space of trigonometric polynomials of order at most \( n \), and if \( f \) and \( g \) are in \( C[0, 2\pi] \), then

\[
\text{proj}_W(f + g) = \text{proj}_W f + \text{proj}_W g
\]

That is, the \( n \)th order Fourier approximation to \( f + g \) is the sum of the \( n \)th order Fourier approximations to \( f \) and to \( g \). Can you use the linearity of the mapping \( f \mapsto \text{proj}_W f \) and the final result of Example 4, to produce (with practically no work) the answer to Exercise 9? [Hint: The \( n \)th order Fourier approximation to a constant function is the function itself.]

---

**MATLAB Graphing Functions**

After you find \( f_4 \) and \( f_5 \) by hand computations, you can use `plot` to graph them. For instance, to plot \( f(t) = \sin t + \sin 3t \), you can write

```matlab
% MATLAB code

t = linspace(0,2*pi);
y = sin(t) + sin(3*t);
plot(t,y)
```

See the MATLAB box for Section 4.1 for more details.

---

**CHAPTER 6 GLOSSARY CHECKLIST**

Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

- **angle** (between nonzero vectors \( u \) and \( v \) in \( \mathbb{R}^2 \) or \( \mathbb{R}^3 \)): The angle \( \theta \) between the . . . . Related to the scalar product by: \( u \cdot v = . . . . \)
- **best approximation**: The closest point . . . .
- **Cauchy-Schwarz inequality**: . . . for all \( u, v \).
- **component of \( y \) orthogonal to \( u \) (for \( u \neq 0 \))**: The vector . . . .
- **design matrix**: The matrix \( X \) in the linear model . . . , where the columns of \( X \) are determined in some way by . . . .
- **distance between \( u \) and \( v \)**: . . . , denoted by \( \text{dist}(u, v) \).
- **Fourier approximation (of order \( n \))**: The closest point in . . . to . . . .
- **Fourier coefficients**: The weights used to make . . . .
Fourier series: An infinite series that . . . in $C[0, 2\pi]$, with the inner product given by a definite integral.

**fundamental subspaces** (of $A$): The . . . space and . . . space of $A$, and the . . . space and . . . space of $A^T$, with $\text{Col } A^T$ commonly called the . . .

**general least-squares problem**: Given an $m \times n$ matrix $A$ and a vector $b$ in $\mathbb{R}^m$, find . . . such that . . . for all . . .

**Gram-Schmidt process**: An algorithm for producing . . .

**inner product**: The scalar . . . , usually written as $\mathbf{u} \cdot \mathbf{v}$, where $\mathbf{u}$ and $\mathbf{v}$ are vectors in $\mathbb{R}^n$ viewed as . . . Also called the . . . of $\mathbf{u}$ and $\mathbf{v}$. In general, a function on a vector space that assigns to each pair of vectors $\mathbf{u}$ and $\mathbf{v}$ a number . . ., subject to certain axioms.

**inner product space**: A vector space on which is defined . . .

**least-squares line**: The line . . . that minimizes . . . in the equation . . .

**least-squares solution** (of $Ax = b$): A vector . . . such that . . .

**length** (of $v$): The scalar $||v|| = . . .$; also called the . . . of $v$.

**linear model** (in statistics): Any equation of the form . . ., where $X$ and $y$ are known and $\beta$ is to be chosen to minimize . . .

**mean-deviation form** (of a vector): A vector whose entries . . .

**multiple regression**: A linear model involving . . . variables and . . .

**normal equations**: The system of equations represented by . . ., whose solution yields all . . . solutions of $Ax = b$. In statistics, a common notation is . . .

**normalizing** (a vector $v$): The process of creating a . . . vector $u$ that . . .

**observation vector**: The vector . . . in the linear model $y = X\beta + \epsilon$, where the entries in . . . are the observed values of . . .

**orthogonal basis**: A basis that . . .

**orthogonal complement** (of $W$): The set $W^\perp$ of . . .

**orthogonal matrix**: A . . . matrix $U$ such that . . .

**orthogonal projection of $y$ onto $u$** (or onto the line through $u$ and the origin, for $u \neq 0$): The vector $\hat{y}$ defined by . . .

**orthogonal projection of $y$ onto $W$**: The unique vector $\hat{y}$ such that . . .

   Notation: $\hat{y} = \text{proj}_W y$.

**orthogonal set**: A set $S$ of vectors such that . . . for . . .

**orthogonal to $W$**: Orthogonal to every . . .

**orthonormal basis**: A basis that is . . .

**orthonormal set**: An . . . set of . . .

**parameter vector**: The unknown vector . . . in the linear model . . .
regression coefficients: The coefficients ... in the ...

residual vector: The quantity ... that appears in the general linear model: ..., the difference between ... and the ... values (of $y$).

QR factorization: A factorization of an $m \times n$ matrix $A$ with linearly independent columns, $A = QR$, where $Q$ is an ... matrix whose ..., and $R$ is an ... matrix.

same direction (as a vector $v$): A vector that is ... .

scale (a vector): Multiply a vector by ... .

trend analysis: The use of ... to fit data, with the inner product ...

triangle inequality: ...

trigonometric polynomial: A linear combination of ... and ... functions such as ...

unit vector: A vector $v$ such that ...

weighted least squares: Least-squares problems with a ... inner product such as $\langle x, y \rangle = ...$
7.1  DIAGONALIZATION OF SYMMETRIC MATRICES

To prepare for this section, review Section 5.3. Focus on the Diagonalization Theorem, Example 3, and Theorem 6. Also, review Example 3 in Section 6.2.

STUDY NOTES

If a symmetric matrix has distinct eigenvalues, as in Example 2, then the ordinary diagonalization process produces a matrix $P$ with orthogonal columns, because eigenvectors from different eigenspaces are automatically orthogonal. However, the $P$ you need here must have orthonormal columns. Forgetting to normalize the columns of $P$ is the main error students make in this section.

The statements in Theorem 3 (The Spectral Theorem), together with the general approach used in Example 3, lead to the following outline for orthogonally diagonalizing any symmetric matrix.

**Procedure for Orthogonally Diagonalizing a Symmetric Matrix $A$**

1. Find the eigenvalues of $A$.
2. For each eigenvalue of $A$, construct an orthonormal basis for the eigenspace.
   a. If the eigenspace has a basis \{v\}, normalize $v$ to produce a unit vector $u$.
   b. If the eigenspace has a basis \{v$_1$, v$_2$\}, first produce an orthogonal basis \{v$_1$, z$_2$\}, where
      \[ z_2 = v_2 - \frac{v_2 \cdot v_1}{v_1 \cdot v_1} v_1 \]
Then normalize \( \mathbf{v}_1 \) and \( \mathbf{z}_2 \) to produce an orthonormal basis \( \{ \mathbf{u}_1, \mathbf{u}_2 \} \) for the eigenspace.

c. If the eigenspace has a basis \( \{ \mathbf{v}_1, ..., \mathbf{v}_k \} \), use the Gram-Schmidt process to construct an orthonormal basis \( \{ \mathbf{u}_1, ..., \mathbf{u}_k \} \) for the eigenspace.

3. The union of the bases for all the eigenspaces is an orthonormal basis for \( \mathbb{R}^n \) (when \( A \) is \( n \times n \)). Use these vectors as the columns of an orthogonal matrix \( P \).

4. Construct \( D \) from the eigenvalues, in an order corresponding to the columns of \( P \). Repeat each eigenvalue according to the dimension of the eigenspace.

5. Finally, \( A = PDP^{-1} = PD^TP \).

The exercises in this section have been constructed so that mastery of the Gram-Schmidt process is not needed, because some courses may omit Section 6.4. However, you do need to understand the calculations in Step 2(b) above.

**SOLUTIONS TO EXERCISES**

1. \( A = \begin{bmatrix} 3 & 5 \\ 5 & -7 \end{bmatrix} = A^T \), because the (1,2) and (2,1) entries match. The entries on the main diagonal of \( A \) can have any values.

7. \( P = \begin{bmatrix} .6 & .8 \\ .8 & -.6 \end{bmatrix} = [\mathbf{p}_1 \ \mathbf{p}_2] \). To show that \( P \) is orthogonal by hand calculations, show that its columns are orthonormal: \( \mathbf{p}_1 \cdot \mathbf{p}_2 = .48 - .48 = 0 \), \( \|\mathbf{p}_1\|^2 = (.6)^2 + (.8)^2 = 1 \), and similarly, \( \|\mathbf{p}_2\|^2 = 1 \). Since \( P \) is square, \( P \) is an orthogonal matrix.

13. \( A = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} \). Characteristic polynomial: \( (3 - \lambda)^2 - 1 = \lambda^2 - 6\lambda + 8 = (\lambda - 4)(\lambda - 2) \). So the eigenvalues are 4 and 2.

For \( \lambda = 4 \): \( [A - 4I \ 0] = \begin{bmatrix} -1 & 1 & 0 \\ 1 & -1 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \), \( x_1 = x_2 \)

Take \( x_2 = 1 \) to get a basis for the eigenspace: \( \begin{bmatrix} 1 \\ 1 \end{bmatrix} \). Then normalize to get a unit vector:

\( \mathbf{u}_1 = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} \). (Don’t forget this step.)

For \( \lambda = 2 \): \( [A - 2I \ 0] = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \), \( x_1 = -x_2 \)

\( x_2 \) is free
Take \( x_2 = 1 \) to get a basis for the eigenspace: \[
\begin{bmatrix}
-1 \\
1
\end{bmatrix}.
\]
Then normalize to get a unit vector:

\[
\begin{bmatrix}
-1/\sqrt{2} \\
1/\sqrt{2}
\end{bmatrix}.
\]

Set \( P = [u_1 \ u_2] = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\
1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \). The corresponding \( D \) is \[
\begin{bmatrix}
4 & 0 \\
0 & 2
\end{bmatrix}.
\]

**Study Tip:** The fact that eigenvectors for distinct eigenvalues are orthogonal gives you a check on your work. After you find \( u_2 \) in Exercise 13, verify that \( u_2 \cdot u_1 = 0 \). Actually, when \( u_1 \) and \( u_2 \) are in \( \mathbb{R}^2 \), you can easily guess what \( u_2 \) must be, once you know \( u_1 \). If you do this, you should compute \( Au_2 \) to make sure that \( u_2 \) is indeed an eigenvector. Ask your instructor how much work you should show on a test question similar to Exercise 13.

19. Be sure to work this problem before reading the solution. Use Exercises 12–24 to sharpen your skills. They are critical for the rest of the chapter. Here, \( A = \begin{bmatrix} 3 & -2 & 4 \\
-2 & 6 & 2 \\
4 & 2 & 3 \end{bmatrix} \), and the eigenvalues are given: 7 and –2.

For \( \lambda = 7 \):

\[
[A - 7I \ 0] = \begin{bmatrix}
-4 & -2 & 4 & 0 \\
-2 & -1 & 2 & 0 \\
4 & 2 & -4 & 0
\end{bmatrix} \sim \begin{bmatrix} 1 & .5 & -1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

Thus, \( x_1 = -.5x_2 + x_3 \), with \( x_2 \) and \( x_3 \) free. Instead of describing all vectors in the eigenspace, you can produce a basis quickly by choosing two linearly independent solutions of \( (A - 7I)x = 0 \). The natural choices are the vector corresponding to \( x_2 = 1 \) and \( x_3 = 0 \) and the vector for \( x_2 = 0 \) and \( x_3 = 1 \). However, in this particular problem, the coefficient \(-.5\) of \( x_2 \) suggests that a better choice for the first vector is to take \( x_2 = 2 \) and \( x_3 = 0 \). In this case, the two vectors are

\[
v_1 = \begin{bmatrix}
-1 \\
2 \\
0
\end{bmatrix} \quad \text{and} \quad v_2 = \begin{bmatrix}
1 \\
0 \\
1
\end{bmatrix}.
\]

This basis for the eigenspace is not orthogonal. Keep \( v_1 \) and subtract from \( v_2 \) its orthogonal projection onto \( v_1 \). The new vector, \( z_2 \), is an eigenvector for the eigenvalue 7 because it is a linear combination of the vectors \( v_2 \) and \( v_1 \) in the eigenspace for \( \lambda = 7 \):

\[
z_2 = v_2 - \frac{v_2 \cdot v_1}{v_1 \cdot v_1}v_1 = \begin{bmatrix}
1 \\
0 \\
1
\end{bmatrix} - \frac{-1}{5} \begin{bmatrix}
-1 \\
2 \\
0
\end{bmatrix} = \begin{bmatrix}
4/5 \\
2/5 \\
1
\end{bmatrix}
\]
Instead of \( z \), use \( z' = \begin{bmatrix} 4 \\ 2 \\ 5 \end{bmatrix} \), which is easier to normalize. Check that \( z' \cdot v_i = 0 \). An orthonormal basis for the eigenspace is \( u_1 = \begin{bmatrix} -1/\sqrt{5} \\ 2/\sqrt{5} \\ 0 \end{bmatrix} \), \( u_2 = \begin{bmatrix} 4/\sqrt{45} \\ 0 \\ 5/\sqrt{45} \end{bmatrix} \).

For \( \lambda = -2 \):

\[
\begin{bmatrix} 5 & -2 & 4 & 0 \\ 4 & 2 & 5 & 0 \\ 1 & 4 & -1 & 0 \\ 0 & 1 & 1/2 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & -4 & -1 & 0 \\ 4 & 2 & 5 & 0 \\ 0 & 1 & -1/2 & 0 \end{bmatrix} \sim \begin{bmatrix} 1 & -4 & -1 & 0 \\ 0 & 1 & 1/2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}
\]

\( x_1 = -x_3 \), \( x_2 = -(1/2)x_3 \), \( x_3 \) is free.

Take \( x_3 = 2 \) to get a basis for the eigenspace, \( \begin{bmatrix} -2 \\ 1 \\ 2 \end{bmatrix} \), and normalize to obtain \( u_3 = \begin{bmatrix} -2/3 \\ -1/3 \\ 2/3 \end{bmatrix} \).

Finally, set \( P = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix} = \begin{bmatrix} -1/\sqrt{5} & 4/\sqrt{45} & -2/3 \\ 2/\sqrt{5} & 2/\sqrt{45} & -1/3 \\ 0 & 5/\sqrt{45} & 2/3 \end{bmatrix} \) and \( D = \begin{bmatrix} 7 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & -2 \end{bmatrix} \).

What other answers might someone produce? If the vectors \( v_1 \) and \( v_2 \) are interchanged, the first two columns of \( P \) probably will be \( \begin{bmatrix} 1/\sqrt{2} \\ 0 \\ 1/\sqrt{2} \end{bmatrix} \) and \( \begin{bmatrix} -1/\sqrt{18} \\ 4/\sqrt{18} \\ 1/\sqrt{18} \end{bmatrix} \). If the entries in \( D \) are rearranged, the columns of \( P \) must be rearranged to correspond to the new entries in \( D \).

**Study Tip:** The matrix in Exercise 20 has only two distinct eigenvalues (according to the text’s information), so one or both eigenspaces will be at least two-dimensional. You will have to construct an orthonormal basis for such an eigenspace. Exercises 23 and 24 are good models for exam questions because they give you information from which you can orthogonally diagonalize \( A \) without extensive computations.

**25.**

**a.** See Theorem 2 and the paragraph preceding the theorem.

**b.** See Theorem 1.

**c.** See Example 3.

**d.** See the paragraph following formula (2) and Exercise 35.
29. By hypothesis, \( A = PDP^{-1} \), where \( P \) is orthogonal and \( D \) is diagonal. Since \( A \) is invertible, 0 is not an eigenvalue and \( D \) is invertible. Then \( A^{-1} = (PDP^{-1})^{-1} = (P^{-1})^{-1}DP^{-1} = PD^{-1}P^{-1} \).

Since \( D^{-1} \) is diagonal, \( A^{-1} \) is orthogonally diagonalizable.

A second argument: By Theorem 2, \( A \) is symmetric. Since \( A \) is invertible, a property of transposes shows that \( (A^{-1})^T = (A^T)^{-1} = A^{-1} \), so \( A^{-1} \) is symmetric. By Theorem 2, again, \( A^{-1} \) is orthogonally diagonalizable.

31. The solution is in the text.

35. a. The matrix \( B = uu^T \) is an outer product, or a rank 1 matrix. Given \( x \) in \( \mathbb{R}^n \), \( Bx = (uu^T)x = u(u^Tx) = (u^Tx)u \), because \( u^Tx \) is a scalar. Using dot products, \( Bx = (x \cdot u)u \). Since \( u \) is a unit vector, this is the orthogonal projection of \( x \) onto \( u \). See Section 6.2.

b. \( B \) is symmetric, because \( B^T = (uu^T)^T = u^Tu^T = uu^T = B \). Also, \( B^2 = (uu^T)(uu^T) = u(u^Tu)^Tu^T = uu^T = B \), because \( u^Tu = 1 \).

MATLAB Orthogonal Diagonalization

Use \texttt{eig(A)} for eigenvalues and \texttt{nulbasis} to obtain eigenvectors, as in Section 5.3.

If you encounter a two-dimensional eigenspace with a basis \( \{v_1, v_2\} \), use the command

\[
v2 = v2 - (v2'*v1)/(v1'*v1)*v1
\]

or

\[
v2 = v2 - \text{proj}(v2,v1)
\]

to make the new eigenvector \( v_2 \) orthogonal to \( v_1 \). See the MATLAB note for Section 6.4. After you normalize the eigenvectors and create \( P \), check that \( P^T P = I \), to verify that \( P \) is indeed an orthogonal matrix. For practice, you might use MATLAB to work Exercise 19.

7.2 QUADRATIC FORMS

Sections 7.1 and 7.2 provide the foundation for the rest of the chapter.

KEY IDEAS

The main point here is to learn how a change of variable, \( x = Py \), with \( P \) an orthogonal matrix, can transform a quadratic form into a new quadratic form with no cross-product terms.

The equation \( x = Py \) expresses \( x \) as a linear combination of the columns of \( P \), using the entries in \( y \) as weights. If the columns of \( P \) are used as a basis \( \mathcal{B} \) for \( \mathbb{R}^n \), then the entries in \( y \) are the coordinates of \( x \) relative to the basis \( \mathcal{B} \). See Section 4.4 (or Section 2.9). As Section 7.2 shows, the columns of \( P \) are eigenvectors of the matrix \( A \) of the quadratic form \( x^TAx \).
If you were fortunate enough to study Section 5.6 or 5.7, you saw the same equation \( x = Py \), with \( P \) constructed from eigenvectors of \( A \). The key difference here is that \( P \) must be an orthogonal matrix as well as a matrix that diagonalizes \( A \). Nothing less will do. The change of variable \( x = Py \) will work only if \( P^T = P^{-1} \) (and \( A = PDP^{-1} \)).

If your course covers the various classes of quadratic forms (or, equivalently, classes of symmetric matrices), you should learn both the definitions and the characterizations (in Theorem 5) of these classes. Exercise 24 describes another useful way to characterize quadratic forms, often used in multivariable calculus courses. (The \( 2 \times 2 \) case can be generalized to \( n \times n \) matrices.)

### SOLUTIONS TO EXERCISES

1. a. \[ x^TAx = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 5 & 1/3 \\ 1/3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \begin{bmatrix} 5x_1 + (1/3)x_2 \\ (1/3)x_1 + x_2 \end{bmatrix} = 5x_1^2 + (2/3)x_1x_2 + x_2^2 \]

   b. When \( x = (6, 1) \), \( x^TAx = 5(6)^2 + (2/3)(6)(1) + (1)^2 = 185 \).

   c. When \( x = (1, 3) \), \( x^TAx = 5(1)^2 + (2/3)(1)(3) + (3)^2 = 16 \).

7. The matrix of the quadratic form is \( A = \begin{bmatrix} 1 & 5 \\ 5 & 1 \end{bmatrix} \). The characteristic polynomial is \( \lambda^2 - 2\lambda - 24 = (\lambda - 6)(\lambda + 4) \); eigenvalues are 6 and –4.

   For \( \lambda = 6 \): an eigenvector is \( \begin{bmatrix} 1 \\ 1 \end{bmatrix} \), normalized: \( u_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \)

   For \( \lambda = -4 \): an eigenvector is \( \begin{bmatrix} -1 \\ 1 \end{bmatrix} \), normalized: \( u_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 \\ 1 \end{bmatrix} \)

   Thus \( A = PDP^{-1} \) and \( D = P^{-1}AP = P^TAP \), when \( P = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \) and \( D = \begin{bmatrix} 6 & 0 \\ 0 & -4 \end{bmatrix} \).

   The desired change of variable is \( x = Py \), so that

   \[ x^TAx = (Py)^T A(Py) = y^TP^TAPy = y^TDy = 6y_1^2 - 4y_2^2 \]  

\((*)\)

**Study Tip:** To make the “change of variable” requested in Exercise 7, you should: (1) write the equation \( x = Py \) and specify \( P \); (2) show the matrix algebra in (*) that produces the new quadratic form; and (3) include the new quadratic form. Find out how much of this information you should supply if a problem like Exercise 7 were to appear on an exam.

13. The matrix of the quadratic form is \( A = \begin{bmatrix} 1 & -3 \\ -3 & 9 \end{bmatrix} \). The characteristic polynomial is \( \lambda^2 - 10\lambda = \lambda(\lambda - 10) \); the eigenvalues are 10 and 0. Thus the quadratic form is positive semidefinite. To find the change of variable, proceed as in Exercise 7:
For $\lambda = 10$: an eigenvector is \[
\begin{bmatrix}
1 \\
3 \\
-3
\end{bmatrix}, \quad \text{normalized: } u_1 = \frac{1}{\sqrt{10}} \begin{bmatrix} 1 \\ 3 \\ -3 \end{bmatrix}
\]

For $\lambda = 0$: an eigenvector is \[
\begin{bmatrix}
3 \\
1 \\
1
\end{bmatrix}, \quad \text{normalized: } u_2 = \frac{1}{\sqrt{10}} \begin{bmatrix} 3 \\ 1 \\ 1 \end{bmatrix}
\]

Take $P = \frac{1}{\sqrt{10}} \begin{bmatrix} 1 & 3 \\ -3 & 1 \end{bmatrix}$ and $D = \begin{bmatrix} 10 & 0 \\ 0 & 0 \end{bmatrix}$. Since $P$ orthogonally diagonalizes $A$, the desired change of variable is $x = Py$, and

\[
x^TAx = (Py)^TA(Py) = y^TP^TAPy = y^TDy = 10y_1^2
\]

The new quadratic form is $10y_1^2$.

19. Because 8 is larger than 5, you should make the $x_2^2$ term as large as possible. The constraint $x_1^2 + x_2^2 = 1$ keeps $x_2$ from exceeding 1. When $x_1 = 0$ and $x_2 = 1$, the value of the quadratic form is $5(0) + 8(1) = 8$.

21. a. See the definition before Example 1.
    b. See the paragraph following Example 3.
    c. See the Principal Axes Theorem and the Diagonalization Theorem (in Section 5.3).
    d. Check the definition.
    e. See Theorem 5.
    f. See the Numerical Note after Example 6.

25. The text’s answer showed that $x^TB^TBx \geq 0$ for all $x$. To show $B^TB$ is positive definite, suppose that $x^TB^TBx = 0$. Then $(Bx)^TBx = 0$, so that $\|Bx\|^2 = 0$ and $Bx = 0$. If $B$ is invertible, then $x = 0$, which shows that, in this case, the form $x^TB^TBx$ is positive definite.

26. The quadratic forms $x^TAx$ and $x^TBx$ are both positive definite, by Theorem 5, because all eigenvalues of $A$ and $B$ are positive. Then for any nonzero $x$, $x^T(A + B)x = x^TAx + x^TBx > 0$, so the quadratic form $x^T(A + B)x$ is positive definite. Also, the matrix $A + B$ is symmetric, because $(A + B)^T = A^T + B^T = A + B$. By Theorem 5, $A + B$ has positive eigenvalues.

**Mastering Linear Algebra Concepts: Diagonalization and Quadratic Forms**

Since the end of the course draws near, I recommend that you prepare a review sheet that contrasts the Diagonalization Theorem in Section 5.3 with Theorem 2 in Section 7.1. You might begin by copying the statements of these theorems, and then use the following questions to guide your review:

- What special properties does an orthogonal diagonalization have that are not present in all diagonalizations? Consider the eigenvalues, the eigenspaces, the eigenvectors, and the matrix $P$ in $PDP^{-1}$.
• Suppose $A$ is symmetric, and all eigenspaces are one-dimensional. What differences are there between a general diagonalization and an orthogonal diagonalization of $A$? What differences are there when $A$ is symmetric and one eigenspace is two-dimensional?

• Why is an orthogonal diagonalization needed to simplify a quadratic form? Make the change of variable $x = Py$ in $x^TAx$ and show the algebra involved.

• If you studied Section 5.6 or 5.7, compare Figure 4 in Section 5.6 with Figure 3 in Section 5.7. How do the general shapes of the trajectories differ? Why do they differ? Could Figure 5 in Section 5.6 or Figure 5 in Section 5.7 be associated with a symmetric matrix?

7.3 CONSTRAINED OPTIMIZATION

This section is important in its own right, since constrained optimization problems arise in many mathematical problems and applications. The main results of the section are also used in the following two sections.

KEY IDEAS

Theorem 6 gives the main idea. The maximum value of a quadratic form $x^TAx$ over the set of all unit vectors can be computed by finding the greatest eigenvalue of $A$; this maximum value is attained at a corresponding eigenvector. The key step in the proof is to diagonalize $A$ by $P$, substitute $x = Py$, and use the fact that in this case, $x$ and $y$ have the same norm.

Example 6 presents a topic that is widely discussed in elementary economics texts.

SOLUTIONS TO EXERCISES

1. We are given an equality of two quadratic forms:

$$5x_1^2 + 6x_2^2 + 7x_3^2 + 4x_1x_2 - 4x_2x_3 = 9y_1^2 + 6y_2^2 + 3y_3^2$$

The matrix of the left quadratic form is $A = \begin{bmatrix} 5 & 2 & 0 \\ 2 & 6 & -2 \\ 0 & -2 & 7 \end{bmatrix}$.

The equality between the two quadratic forms indicates that the eigenvalues of $A$ are 9, 6, 3.

(Proof: The diagonal matrix $D$ of the quadratic form $9y_1^2 + 6y_2^2 + 3y_3^2$ obviously has eigenvalues 9, 6, 3. Since $A$ is similar to $D$, $A$ has the same eigenvalues as $D$.) The standard calculations produce a unit eigenvector for each eigenvalue. Don’t forget to normalize each eigenvector.
\[ \lambda = 9: \quad u_1 = \begin{bmatrix} 1/3 \\ 2/3 \\ -2/3 \end{bmatrix} \; ; \; \lambda = 6: \quad u_2 = \begin{bmatrix} 2/3 \\ 1/3 \\ 2/3 \end{bmatrix} \; ; \; \lambda = 3: \quad u_3 = \begin{bmatrix} -2/3 \\ 2/3 \\ 1/3 \end{bmatrix} \]

These eigenvectors are mutually orthogonal because they correspond to distinct eigenvalues. So the desired change of variable is

\[ x = Py, \text{ where } P = \begin{bmatrix} 1/3 & 2/3 & -2/3 \\ 2/3 & 1/3 & 2/3 \\ -2/3 & 2/3 & 1/3 \end{bmatrix} \]

**Study Tip:** Review the matrix algebra that leads from \( x^T A x \) to \( y^T D y \) when \( x = Py \). Also, be sure you can show that \( ||x|| = ||y|| \) when \( P \) is an orthogonal matrix.

7. The matrix of \( Q(x) = -2x_1^2 - x_2^2 + 4x_1x_2 + 4x_2x_3 \) is \[ A = \begin{bmatrix} -2 & 2 & 0 \\ 2 & -1 & 2 \\ 0 & 2 & 0 \end{bmatrix} \].

The hint in the exercise lists 2, –1, and –4 as the eigenvalues. The greatest eigenvalue is 2, not –4, because “greatest” here refers to the eigenvalue that is farthest to the right on the real line. The maximum value of \( Q(x) \) (for \( x \) a unit vector) is attained at a unit eigenvector for \( \lambda = 2 \). Standard calculations produce the eigenvector:

\[ v_i = \begin{bmatrix} 1/2 \\ 1 \\ 1 \end{bmatrix}, \text{ scaled to } \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}, \text{ and normalized to } u_i = \begin{bmatrix} 1/3 \\ 2/3 \\ 2/3 \end{bmatrix}. \]

**Warning:** Exercise 7 illustrates the potential error of selecting –4 instead of 2 as the greatest eigenvalue.

12. This exercise can be done by using a theorem, but try to do it by direct computation, using the hint in the text.

13. If \( m = M \) and \( x \) is the unit eigenvector \( u_1 \), then \( x^T A x = u_1^T A u_1 = u_1^T (m u_1) = m \). Otherwise, set \( \alpha = (t - m)/(M - m) \). The hint in the text shows that \( 0 \leq \alpha \leq 1 \) when \( m \leq t \leq M \). For such \( \alpha \), let \( x = \sqrt{1 - \alpha u_n} + \sqrt{\alpha} u_1 \). Then the vectors \( \sqrt{1 - \alpha u_n} \) and \( \sqrt{\alpha} u_1 \) are orthogonal because they are eigenvectors for different eigenvalues (or one of the vectors is \( 0 \)). By the Pythagorean Theorem,
\[ x^T x = \| x \|^2 = \| \sqrt{1 - \alpha} u \|^2 + \| \sqrt{\alpha} u_1 \|^2 \]
\[ = |1 - \alpha| \| u_n \|^2 + |\alpha| \| u_1 \|^2 \]
\[ = (1 - \alpha) + \alpha = 1 \]
because \( u_n \) and \( u_1 \) are unit vectors and \( 0 \leq \alpha \leq 1 \). Also, using the fact that \( u_n \) and \( u_1 \) are orthogonal, compute
\[ x^T Ax = (\sqrt{1 - \alpha} u_n + \sqrt{\alpha} u_1)^T A (\sqrt{1 - \alpha} u_n + \sqrt{\alpha} u_1) \]
\[ = (\sqrt{1 - \alpha} u_n + \sqrt{\alpha} u_1)^T (m \sqrt{1 - \alpha} u_n + M \sqrt{\alpha} u_1) \]
\[ = |1 - \alpha| mu_n^T u_n + |\alpha| Mu_1^T u_1 = (1 - \alpha)m + \alpha M = t \]
Thus the quadratic form \( x^T Ax \) assumes every value between \( m \) and \( M \) for a suitable unit vector \( x \).

7.4 THE SINGULAR VALUE DECOMPOSITION

This section is the capstone of the text. It completes the story of the linear transformation \( x \mapsto Ax \) for a general \( m \times n \) matrix \( A \) and, in so doing, gives you an opportunity to review many basic concepts from Chapters 4–7. In addition, this section opens the door into the modern world of applied linear algebra. An understanding of the singular value decomposition is essential for advanced work in science and engineering that requires matrix computations.

KEY IDEAS

The first singular value \( \sigma_1 \) of an \( m \times n \) matrix \( A \) is the maximum of \( \| Ax \| \) over all unit vectors. This maximum value is attained at a unit eigenvector \( v_1 \) of \( A^T A \) corresponding to the greatest eigenvalue \( \lambda_1 \) of \( A^T A \). The second singular value is the maximum of \( \| Ax \| \) over all unit vectors orthogonal to \( v_1 \). The following algorithm produces the singular value decomposition for \( A \). (As mentioned in the text, other more reliable methods are used in professional software.)

**Procedure for Computing a Singular Value Decomposition**

1. Find an orthonormal basis \( \{v_1, \ldots, v_n\} \) for \( \mathbb{R}^n \) consisting of eigenvectors of \( A^T A \), arranged so that the associated eigenvalues satisfy \( \lambda_1 \geq \cdots \geq \lambda_r > 0 \) and \( \lambda_{r+1} = \cdots = 0 \), where \( r = \text{rank} \ A \).
2. Construct the \( n \times n \) orthogonal matrix \( V = [v_1 \cdots v_n] \).
3. Let \( \sigma_j = \sqrt{\lambda_j} \) (1 \( \leq j \leq n \)), and construct the \( m \times n \) diagonal matrix \( \Sigma \) whose \( (j,j) \)-entry is \( \sigma_j \) (1 \( \leq j \leq n \)) and has zeros elsewhere.
4. The set \{A\vec{v}_1, \ldots, A\vec{v}_r\} is orthogonal and $\sigma_j = \|A\vec{v}_j\|$. Compute $\vec{u}_j = (1/\sigma_j)A\vec{v}_j$ for $1 \leq j \leq r$.

5. Extend \{\vec{u}_1, \ldots, \vec{u}_r\} to an orthonormal basis \{\vec{u}_1, \ldots, \vec{u}_m\} for $\mathbb{R}^m$. Write the $m \times m$ orthogonal matrix $U = [\vec{u}_1 \ \cdots \ \vec{u}_m]$.

6. $A = U\Sigma V^T$.

The diagram below illustrates how the SVD splits the action of $A$ into first multiplication by $V^T$ (which amounts to an orthogonal change of basis in $\mathbb{R}^n$), then a scaling by $\Sigma$ in the directions of the standard basis vectors $\vec{e}_1, \ldots, \vec{e}_n$, and finally multiplication by $U$ (an orthogonal change of basis in $\mathbb{R}^m$).

The Singular Value Decomposition: $A = U\Sigma V^T$

Note that the so-called left singular vectors of $A$ are the columns of $U$ (because $U$ appears on the left in the factorization of $A$), even though $\vec{u}_1, \ldots, \vec{u}_r$ appear on the right side of the diagram above.

**SOLUTIONS TO EXERCISES**

1. $A = \begin{bmatrix} 1 & 0 \\ 0 & -3 \end{bmatrix}$, $A^T A = \begin{bmatrix} 1 & 0 \\ 0 & 9 \end{bmatrix}$. Eigenvalues and eigenvectors of $A^T A$ are:

   $\lambda_1 = 9: \vec{v}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$; $\lambda_2 = 1: \vec{v}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

   (Remember to arrange the eigenvalues in decreasing order.) Thus

   $V = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
The singular values are $\sigma_1 = \sqrt{9} = 3$ and $\sigma_2 = 1$. The matrix $\Sigma$ is the same shape as $A$, and

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$$

Next, compute

$$A\mathbf{v}_1 = \begin{bmatrix} 1 & 0 \\ 0 & -3 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -3 \end{bmatrix}, \quad A\mathbf{v}_2 = \begin{bmatrix} 1 & 0 \\ 0 & -3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and normalize:

$$\mathbf{u}_1 = \frac{1}{\sigma_1} A\mathbf{v}_1 = \frac{1}{3} \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad \mathbf{u}_2 = \frac{1}{\sigma_2} A\mathbf{v}_2 = \frac{1}{1} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Finally, $\{\mathbf{u}_1, \mathbf{u}_2\}$ is already a basis for $\mathbb{R}^2$, so the basis for $\mathbb{R}^2$ is complete, and

$$U = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad A = U\Sigma^{T} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$$

7. $A = \begin{bmatrix} 2 & -1 \\ 2 & 2 \end{bmatrix}$. Find the eigenvalues of $A^TA$ from the characteristic equation.

$$0 = \lambda^2 - 13\lambda + 36 = (\lambda - 9)(\lambda - 4); \quad \lambda_1 = 9, \quad \lambda_2 = 4$$

Corresponding unit eigenvectors for $A^TA$ (calculations omitted) are:

$$\lambda_1 = 9: \quad \mathbf{v}_1 = \begin{bmatrix} 2/\sqrt{5} \\ 1/\sqrt{5} \end{bmatrix}; \quad \lambda_2 = 4: \quad \mathbf{v}_2 = \begin{bmatrix} -1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$$

Take

$$V = \begin{bmatrix} 2/\sqrt{5} & -1/\sqrt{5} \\ 1/\sqrt{5} & 2/\sqrt{5} \end{bmatrix}$$

The singular values are $\sigma_1 = \sqrt{9} = 3$ and $\sigma_2 = \sqrt{4} = 2$. The matrix $\Sigma$ is the same shape as $A$ and $\Sigma = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}$. Next, compute

$$A\mathbf{v}_1 = \begin{bmatrix} 2 & -1 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} 2/\sqrt{5} \\ 1/\sqrt{5} \end{bmatrix} = \begin{bmatrix} 3/\sqrt{5} \\ 6/\sqrt{5} \end{bmatrix}, \quad A\mathbf{v}_2 = \begin{bmatrix} 2 & -1 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} -1/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix} = \begin{bmatrix} -4/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}$$

To check your work at this point, verify that $A\mathbf{v}_1$ and $A\mathbf{v}_2$ are orthogonal. (They are.) Then normalize:
\[ u_1 = \frac{1}{\sigma_1} A v_1 = \frac{1}{3} \begin{bmatrix} 3/\sqrt{5} \\ 6/\sqrt{5} \\ 2/\sqrt{5} \end{bmatrix}, \quad u_2 = \frac{1}{\sigma_2} = \frac{1}{2} \begin{bmatrix} -4/\sqrt{5} \\ 2/\sqrt{5} \\ 1/\sqrt{5} \end{bmatrix} \]

Since \( \{u_1, u_2\} \) is a basis for \( \mathbb{R}^2 \), take \( U = \begin{bmatrix} 1/\sqrt{5} & -2/\sqrt{5} \\ 2/\sqrt{5} & 1/\sqrt{5} \end{bmatrix} \). Thus

\[ A = U \Sigma V^T = \begin{bmatrix} 1/\sqrt{5} & -2/\sqrt{5} \\ 2/\sqrt{5} & 1/\sqrt{5} \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2/\sqrt{5} & 1/\sqrt{5} \\ -1/\sqrt{5} & 2/\sqrt{5} \end{bmatrix} \]

(use \( v^T \), not \( V \)).

**Study Tip:** Your answer for a singular value decomposition may differ from that given in the text. To check your work, compute \( AV \) and \( U \Sigma \). If \( AV = U \Sigma \), then \( A = U \Sigma V^T \) and your answer is correct (provided \( U \) and \( V \) truly are orthogonal matrices).

13. The matrix \( A^T A \) is \( 3 \times 3 \). Because the text has not given you practice computing and solving a cubic characteristic equation, the Hint suggests that you consider \( A^T \) instead of \( A \). (You are free to work on \( A \) itself, if you prefer.) Using \( A^T \), compute

\[ (A^T)^T A^T = A A^T = \begin{bmatrix} 3 & 2 & 2 \\ 2 & 3 & -2 \\ 2 & -2 & 2 \end{bmatrix} \begin{bmatrix} 3 & 2 \\ 2 & 3 \\ 2 & -2 \end{bmatrix} = \begin{bmatrix} 17 & 8 \\ 8 & 17 \end{bmatrix} \]

The characteristic equation is

\[ 0 = \lambda^2 - 34\lambda + 225 = (\lambda - 25)(\lambda - 9); \quad \lambda_1 = 25, \quad \lambda_2 = 9 \]

The corresponding unit eigenvectors and the matrix \( V \) are

\[ \lambda_1 = 25: \quad v_1 = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}; \quad \lambda_2 = 9: \quad v_2 = \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}; \quad V = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \]

The singular values are \( \sigma_1 = 5 \) and \( \sigma_2 = 3 \). Thus \( \Sigma = \begin{bmatrix} 5 & 0 \\ 0 & 3 \end{bmatrix} \), the same size as \( A^T \). To get \( u_1 \) and \( u_2 \), compute \( A^T v_1 \) and \( A^T v_2 \),

\[ A^T [v_1 \quad v_2] = \begin{bmatrix} 3 & 2 \\ 2 & 3 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 5/\sqrt{2} & -1/\sqrt{2} \\ 5/\sqrt{2} & 1/\sqrt{2} \\ 0 & -4/\sqrt{2} \end{bmatrix} \]

and normalize:
We need one more vector, orthogonal to \( u_1 \) and \( u_2 \). So write the equations \( u_1^T x = 0 \) and \( u_2^T x = 0 \) and solve for \( x \). Simpler equations are

\[
\sqrt{2} u_1^T x = 0 \quad \text{or} \quad x_1 + x_2 = 0 \\
\sqrt{8} u_2^T x = 0 \quad \text{or} \quad -x_1 + x_2 - 4x_3 = 0
\]

The solution is \( x_1 = -2x_3, \quad x_2 = 2x_3, \quad x_3 \text{ free} \). A suitable unit vector is

\[
u_3 = \begin{bmatrix} -2/3 \\ 2/3 \\ 1/3 \end{bmatrix}
\]

Thus an SVD of \( A^T \) is

\[
A^T = [u_1 \quad u_2 \quad u_3] \begin{bmatrix} 5 & 0 \\ 0 & 3 \end{bmatrix} [v_1 \quad v_2]^T
\]

So an SVD of \( A \) appears by taking transposes:

\[
A = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} 5 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ -1/\sqrt{18} & 1/\sqrt{18} & -4/\sqrt{18} \\ -2/\sqrt{3} & 2/3 & 1/3 \end{bmatrix}
\]

This is an SVD because the outside matrices are orthogonal matrices, and the center matrix is a diagonal matrix of the proper type. Another way to find \( u_3 \) is to realize that \( u_1 \) and \( u_2 \) form an orthonormal basis for \( \text{Row } A \). The remaining \( u_3 \) must be a basis for \( \text{Nul } A^T \). (See Fig. 4 on page 421.) One way to obtain \( u_{r+1}, \ldots, u_n \) is to construct a basis for the solution set of \( A^T x = 0 \) and then perform the Gram-Schmidt process.

**Study Tip:** Exercises 15 and 16 make good exam questions.

19. Let \( A = U\Sigma V^T \). Then

\[
A^T A = (U\Sigma V^T)^T U\Sigma V^T = V\Sigma^T U^T U\Sigma V^T
\]

Because \( U \) and \( V \) are orthogonal

If \( \sigma_1, \ldots, \sigma_r \) are the nonzero diagonal entries in \( \Sigma \), then \( \Sigma^T \Sigma \) is diagonal, with diagonal entries \( \sigma_1^2, \ldots, \sigma_r^2 \) and possibly some zeros. Thus \( V \) diagonalizes \( A^T A \). By the Diagonalization
Theorem in Section 5.3, the columns of $V$ are eigenvectors of $A^T A$, and $\sigma_1^2, \ldots, \sigma_r^2$ are the nonzero eigenvalues of $A^T A$. Hence $\sigma_1, \ldots, \sigma_r$ are the nonzero singular values of $A$. A similar calculation of $AA^T$ shows that the columns of $U$ are eigenvectors of $AA^T$.

23. From the proof of Theorem 10, $U \Sigma = \begin{bmatrix} \sigma_1 u_1 & \cdots & \sigma_r u_r & 0 & \cdots & 0 \end{bmatrix}$. The column-row expansion of a matrix product shows that

$$A = (U \Sigma) V^T = (U \Sigma) \begin{bmatrix} v_1^T \\ \vdots \\ v_n^T \end{bmatrix} = \sigma_1 u_1 v_1^T + \cdots + \sigma_r u_r v_r^T$$

This expansion generalizes the spectral decomposition in Section 7.1.

25. Consider the SVD for the standard matrix of $T$, say, $A = U \Sigma V^T$. Let $B = \{v_1, \ldots, v_n\}$ and $C = \{u_1, \ldots, u_m\}$ be bases constructed from the columns of $V$ and $U$, respectively. Observe that, since the columns of $V$ are orthonormal, $V^T v_j = e_j$, where $e_j$ is the $j$th column of the $n \times n$ identity matrix. To find the matrix of $T$ relative to $B$ and $C$, compute

$$T(v_j) = A v_j = U \Sigma V^T v_j = U \Sigma e_j = U \sigma_j e_j = \sigma_j U e_j = \sigma_j u_j$$

So $[T(v_j)]_c = \sigma_j e_j$. Formula (4) in the discussion at the beginning of Section 5.4 shows that the “diagonal” matrix $\Sigma$ is the matrix of $T$ relative to $B$ and $C$.

**MATLAB The Singular Value Decomposition**

The command $[P \ D] = \text{eig}(A' * A)$ produces an orthogonal matrix $P$ of eigenvectors and a diagonal matrix $D$ of eigenvalues of $A^T A$, but the eigenvalues in $D$ may not be in decreasing order. In such a case, you will have to rearrange things to form $V$ and $\Sigma$ (denoted below by $S$). For instance, if $P$ is $3 \times 3$, the command

$$V = P(:,[1 \ 3 \ 2])$$

interchanges columns 2 and 3 of $P$ to form $V$. The commands

$$S = \text{zeros(size(A))}; \quad S(2,2) = \text{sqrt(D(3,3))}$$

produce a zero matrix for “$\Sigma$” the same size as $A$ and place the square root of the (3,3)-entry of $D$ into the (2,2)-entry of $S$. Other diagonal entries for $S$ can be entered similarly. To form $U$ for the SVD, first normalize the nonzero columns of $A^* V$ and place them in a matrix $U$. If $U$ is square, you are finished. If $U$ is not square, the missing columns must form an orthonormal basis for Null $A^T$. (See Fig. 4 on page 421.) The command $\text{null}(A')$ produces this orthonormal basis. Thus, the square matrix $U$ is given by

$$U = [U \ \text{null}(A')]$$

Copyright © 2016 Pearson Education, Inc.
This construction of the SVD helps you to think about properties of the factorization. In practical work, however, you should use the much faster and more numerically reliable command 

\[ \mathbf{[U S V]} = \text{svd}(\mathbf{A}) \]

7.5 APPLICATIONS TO IMAGE PROCESSING AND STATISTICS

If you find remote sensing or image processing interesting, or if you plan to use multivariate statistics later in your career, then you will want to study this section thoroughly. You may have difficulty finding an elementary explanation of this material elsewhere. The idea for the application to image processing came from a student in my linear algebra class—a geography major who was taking an undergraduate course in remote sensing. The book by Lillesand and Kiefer, referenced in the text, was one of the texts for her course.

KEY IDEAS

The first principal component of the data in the matrix of observations is a unit eigenvector \( \mathbf{u}_1 \) corresponding to the largest eigenvalue of the covariance matrix \( \mathbf{S} \). If \( \mathbf{u}_1 = (c_1, \ldots, c_p) \), then the entries in \( \mathbf{u}_1 \) are weights in a linear combination of the original variables, \( x_1, \ldots, x_p \), that creates a new variable \( y_1 \) (sometimes called a composite score or index):

\[ y_1 = \mathbf{u}_1^\top \mathbf{X} = c_1 x_1 + \cdots + c_p x_p \]

The variance of the values of this index is the largest possible among all indices whose coefficients \( c_1, \ldots, c_p \) form a unit vector. (The variance of \( y_1 \) is the largest eigenvalue of \( \mathbf{S} \).) The second principal component is the unit eigenvector corresponding to the second largest eigenvalue of \( \mathbf{S} \). The entries in the second principal component determine the index with greatest variance among all possible indices (determined by a unit vector) that are uncorrelated (in a statistical sense) with \( y_1 \). Additional principal components are defined similarly.

Checkpoints: (1) If the variables \( x_1 \) and \( x_3 \) are uncorrelated, what can you say about the covariance matrix \( \mathbf{S} \)? (2) What is the covariance matrix of the new variables \( y_1, \ldots, y_p \) formed from the principal components of \( \mathbf{S} \)?

SOLUTIONS TO EXERCISES

1. The matrix of observations is 
\[ \mathbf{X} = \begin{bmatrix} 19 & 22 & 6 & 3 & 2 & 20 \\ 22 & 6 & 9 & 15 & 13 & 5 \end{bmatrix} \]

and the sample mean \( \mathbf{M} \) is 
\[ \begin{bmatrix} 12 \\ 10 \end{bmatrix} \]. Subtract \( \mathbf{M} \) from each column of \( \mathbf{X} \) to obtain

Copyright © 2016 Pearson Education, Inc.
The sample covariance matrix is

\[
S = \frac{1}{N-1}BB^T = \frac{1}{5} \begin{bmatrix}
7 & 10 & -6 & -9 & -10 & 8 \\
10 & -4 & -1 & 5 & 3 & -5
\end{bmatrix}
\]

\[
= \frac{1}{5} \begin{bmatrix}
-430 & -135 \\
-135 & 80
\end{bmatrix}
= \begin{bmatrix}
86 & -27 \\
-27 & 16
\end{bmatrix}
\]

Usually, \( S \) contains decimals.

**Study Tip:** Note that the formula for the sample mean involves division by \( N \), but for statistical reasons, the covariance matrix formula involves division by \( N - 1 \).

7. Let \( x_1, x_2 \) denote the variables for the two-dimensional data in Exercise 1. The characteristic equation of the covariance matrix \( S \) from Exercise 1 is \( \lambda^2 - 102\lambda + 647 = 0 \). By the quadratic formula, the roots of this equation are \( \lambda_1 = 95.20 \) and \( \lambda_2 = 6.80 \) (to two decimal places). The first principal component of the data is a unit eigenvector corresponding to \( \lambda_1 \), which turns out to be \((-0.95, 0.32)\), or \((0.95, -0.32)\). The two possible choices for the new variable are \( y_1 = -0.95 x_1 + 0.32 x_2 \) and \( y_1 = 0.95 x_1 - 0.32 x_2 \). The variance of \( y_1 \) is 95.20, while the total variance is 95.20 + 6.80 = 102. Since 95.20/102 = .933, the new variable \( y_1 \) explains about 93.3% of the variance in the data.

11. a. The solution in the text shows that the \( Y_k \) are in mean-deviation form, where \( Y_k = PX_k \) for some \( p \times p \) matrix \( P \).

b. By part (a), the covariance matrix of \( Y_1, \ldots, Y_N \) is

\[
\frac{1}{N-1}[Y_1 \cdots Y_N][Y_1 \cdots Y_N]^T
= \frac{1}{N-1}P^T[X_1 \cdots X_N][P^T[X_1 \cdots X_N]]^T
= P^T\left(\frac{1}{N-1}[X_1 \cdots X_N][X_1 \cdots X_N]^T\right)P
= P^TSP
\]

because \( X_1, \ldots, X_N \) are in mean-deviation form.
13. Let $\mathbf{M}$ be the sample mean of the data, and for $k = 1, \ldots, N$, write $\hat{X}_k$ for $\mathbf{X}_k - \mathbf{M}$. Let $B = \begin{bmatrix} \hat{X}_1 & \cdots & \hat{X}_N \end{bmatrix}$, the matrix of observations in mean deviation form. By the column-row expansion of $BB^T$, the sample covariance matrix is

$$S = \frac{1}{N-1}BB^T = \frac{1}{N-1} \begin{bmatrix} \hat{X}_1 & \cdots & \hat{X}_N \end{bmatrix} \begin{bmatrix} \hat{X}_1^T \\ \vdots \\ \hat{X}_N^T \end{bmatrix}$$

$$= \frac{1}{N-1} \sum_{k=1}^{N} \hat{X}_k \hat{X}_k^T = \frac{1}{N-1} \sum_{k=1}^{N} (\mathbf{X}_k - \mathbf{M})(\mathbf{X}_k - \mathbf{M})^T$$

**Answers to Checkpoints:**  
(1) The $(1, 3)$-entry and $(3, 1)$-entry of $S$ are zero.  
(2) The covariance matrix of $y_1, \ldots, y_p$ is the diagonal matrix formed from the eigenvalues of $S$. This matrix is diagonal because the new variables are pairwise uncorrelated.  

### MATLAB Computing Principal Components

The command `mean(X')` produces a row vector whose $j$th entry lists the average of the $j$th row of $X$, and `diag(mean(X'))` creates a diagonal matrix whose diagonal entries are the row averages of $X$. (Be careful not to use `mean(X)`, which lists the averages of the columns of $X$.) Finally, the command `diag(mean(X'))*ones(size(X))` creates a matrix the size of $X$, whose columns are all the same, each one listing the row averages of $X$. To convert the data in $X$ into mean-deviation form, use

$$B = X - \text{diag(mean(X'))}*\text{ones(size(X))}$$

The sample covariance matrix is produced by

$$S = BB'/\text{(N-1)}$$

The principal component data is produced by

$$[U,D,V] = \text{svd}(B'/\text{sqrt(N-1)})$$

The columns of $V$ are the principal components of the data, and the diagonal entries of $D^2$ list the variances of the new variates.
CHAPTER 7  SUPPLEMENTARY EXERCISES ________________

7. If \( A = R^TR \), where \( R \) is invertible, then \( A \) is positive definite, by Exercise 25 in Section 7.2. Conversely, suppose that \( A \) is positive definite. Then by Exercise 26 in Section 7.2, \( A = B^TB \) for some positive definite matrix \( B \). Since the eigenvalues of \( B \) are positive, 0 is not an eigenvalue and so \( B \) is invertible. In particular, the columns of \( B \) are linearly independent. By Theorem 12 in Section 6.4, \( B = QR \) for some \( n \times n \) matrix \( Q \) with orthonormal columns and some upper triangular matrix \( R \) with positive elements on its diagonal. Since \( Q \) is square, \( Q^TQ = I \). So

\[
A = B^TB = (QR)^T(QR) = R^TQ^TQR = R^TR
\]

and \( R \) has the required properties.

11. Start with an SVD decomposition, \( A = U\Sigma V^T \). Since \( U \) is orthogonal, \( U^TU = I \), and so

\[
A = U\Sigma U^TUV^T = PQ,
\]

where \( P = U\Sigma U^T = U\Sigma U^{-1} \) and \( Q = UV^T \). The matrix \( P \) is symmetric, because \( \Sigma \) is symmetric, and \( P \) has nonnegative eigenvalues because it is similar to \( \Sigma \) (which is diagonal with nonnegative entries). Thus \( P \) is positive semidefinite. The matrix \( Q \) is orthogonal because it is the product of orthogonal matrices.

CHAPTER 7  GLOSSARY CHECKLIST ________________

Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

**condition number** (of \( A \)): The quotient \( \sigma_1/\sigma_n \), where . . .

**covariance** (of variables \( x_i \) and \( x_j \), for \( i \neq j \)): The entry in the covariance matrix \( S \) for a matrix of observations, where \( x_i \) and \( x_j \) vary over the . . . coordinates, respectively of the observation vectors.

**covariance matrix** (or **sample covariance matrix**): The \( p \times p \) matrix \( S \) defined by \( S = \ldots \), where \( B \) is a \( p \times N \) matrix of observations . . . .

**indefinite matrix**: A symmetric matrix \( A \) such that . . .

**indefinite quadratic form**: A quadratic form \( Q \) such that \( Q(x) \ldots \)

**left singular vectors** (of \( A \)): The columns of . . . in the singular value decomposition \( A = \ldots \).

**matrix of observations**: A \( p \times N \) matrix whose columns are . . . , each column listing \( p \) measurements made on . . .

**mean-deviation form** (of a matrix of observations): A matrix whose . . . vectors are . . .

**Moore-Penrose inverse**: See pseudoinverse.

**negative definite matrix**: A symmetric matrix \( A \) such that . . .

**negative definite quadratic form**: A quadratic form \( Q \) such that \( Q(x) \ldots \)
negative semidefinite matrix:  A symmetric matrix $A$ such that . . .
negative semidefinite quadratic form:  A quadratic form $Q$ such that . . .
orthogonally diagonalizable:  A matrix $A$ that admits a factorization, $A = PDP^{-1}$, with $P$ . . . and $D$ . . .
positive definite matrix:  A symmetric matrix $A$ such that . . .
positive definite quadratic form:  A quadratic form $Q$ such that $Q(x)$ . . .
positive semidefinite matrix:  A symmetric matrix $A$ such that . . .
positive semidefinite quadratic form:  A quadratic form $Q$ such that . . .
principal axes (of a quadratic form $x^TAx$):  The orthonormal columns of an orthogonal matrix $P$
such that . . .
principal components (of the data in a matrix of observations $B$):  The . . . eigenvectors of a sample covariance matrix $S$ for $B$, with the eigenvectors arranged so that the corresponding . . .
projection matrix (or orthogonal projection matrix):  A symmetric matrix $B$ such that . . .
A simple example is $B = . . .$
pseudoinverse (of $A$):  The matrix . . ., when $UDVT$ is a reduced singular value decomposition of $A$.

quadratic form:  A function $Q$ defined for $x$ in $\mathbb{R}^n$ by $Q(x) = . . .$, where $A$ is an $n \times n$ . . . matrix $A$ (called the matrix of the quadratic form).

reduced singular value decomposition:  A factorization $A = . . .$, for an $m \times n$ matrix $A$ of rank $r$, where $U$ is $\times$ with orthonormal columns, $D$ is $\times$ with . . ., and $V$ is $\times$ with orthonormal columns.

right singular vectors (of $A$):  The columns of . . . in the singular value decomposition $A = . . .$
row sum:  The sum of the entries . . .
sample mean:  The average $M$ of a set of vectors, $X_1, \ldots, X_N$, given by $M = . . .$
singular value decomposition (of an $m \times n$ matrix $A$):  $A = . . .$, where $U$ is an $\times$ . . . matrix, $V$ is an $\times$ . . . matrix, and $\Sigma$ is an $\times$ . . . matrix with . . .
singular values (of $A$):  The . . . of the eigenvalues of . . ., arranged . . .
spectral decomposition (of $A$):  A representation $A = . . .$, where . . .
symmetric matrix:  A matrix $A$ such that . . .
total variance:  The . . . of the covariance matrix $S$ of a matrix of . . .
uncorrelated variables:  Any two variables $x_i$ and $x_j$ (with $i \neq j$) that range over the $i$th and $j$th coordinates of the observation vectors in an observation matrix, such that . . .

variance of a variable $x_j$:  The diagonal entry . . . in the . . . matrix $S$ for a matrix of observations, where $x_j$ varies over the $j$th coordinates of the . . .
This chapter begins with geometric ideas that are already familiar to you. Affine combinations of vectors generalize the concepts of lines and planes to higher dimensions and describe the mathematical objects formed when subspaces are shifted away from the origin. Hyperplanes are a generalization of planes, and polytopes are a generalization of polygons.

The geometric objects studied in this chapter include line segments, solid triangles and polygons, special types of curves and surfaces, and generalizations to sets in more than three dimensions. These objects form the building blocks for applications to computer graphics and to linear programming.

8.1 AFFINE COMBINATIONS

This section introduces a special kind of linear combination used to describe useful geometric objects involving two or more variables.

KEY IDEAS
In this section, vectors are viewed as points in \( \mathbb{R}^n \). Important definitions: affine combinations, affine sets, and flats (or, translations of subspaces). The affine combination of two linearly independent vectors (points) is just the line through these two points; the affine combination of three linearly independent vectors (points) is the plane through these three points. Theorems 1 to 4 give important properties of affine combinations and affine sets.
STUDY NOTES

As you work through this section, try to link the new concepts here to similar concepts from Chapter 1. To write \( y \) as an affine combination of a set of vectors \( \{v_1, \ldots, v_p\} \), first write \( y - v_1 \) as a linear combination of \( v_2 - v_1, \ldots, v_p - v_1 \). To do this, row reduce the augmented matrix \( [v_2 - v_1 \ v_3 - v_1 \ \ldots \ v_p - v_1 \ y - v_1] \) and read your solution from the result. Once you have written \( y - v_1 \) as a linear combination of \( v_2 - v_1, \ldots, v_p - v_1 \), you can solve for \( y \) as an affine combination of \( v_1, \ldots, v_p \). Note that you can also translate theoretical questions about writing \( y \) as an affine combination of a set of vectors \( \{v_1, \ldots, v_p\} \) into questions about writing \( y - v_1 \) as a linear combination the of vectors \( \{v_2 - v_1, \ldots, v_p - v_1\} \).

SOLUTIONS TO EXERCISES

1. \( v_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \ v_2 = \begin{bmatrix} -2 \\ 2 \end{bmatrix}, \ v_3 = \begin{bmatrix} 0 \\ 4 \end{bmatrix}, \ v_4 = \begin{bmatrix} 3 \\ 7 \end{bmatrix}, \ y = \begin{bmatrix} 5 \\ 3 \end{bmatrix} \). First, translate the points by \( v_1 \):

\[
\begin{align*}
v_2 - v_1 &= \begin{bmatrix} -3 \\ 0 \end{bmatrix}, \\
v_3 - v_1 &= \begin{bmatrix} -1 \\ 2 \end{bmatrix}, \\
v_4 - v_1 &= \begin{bmatrix} 2 \\ 5 \end{bmatrix}, \\
y - v_1 &= \begin{bmatrix} 4 \\ 1 \end{bmatrix}.
\end{align*}
\]

To find scalars \( c_2, c_3, \) and \( c_4 \) such that

\[
c_2(v_2 - v_1) + c_3(v_3 - v_1) + c_4(v_4 - v_1) = y - v_1 \quad (*)
\]

use the strategy of Example 1, and row reduce the augmented matrix having these translated points as columns:

\[
\begin{bmatrix} -3 & -1 & 2 & 4 \\ 0 & 2 & 5 & 1 \end{bmatrix} \sim \cdots \sim \begin{bmatrix} 1 & 0 & -1.5 & -1.5 \\ 0 & 1 & 2.5 & .5 \end{bmatrix}.
\]

From this conclude that the general solution of (*) is \( c_2 = 1.5c_4 - 1.5 \) and \( c_3 = -2.5c_4 + .5 \), with \( c_4 \) free. Set \( c_4 = 0 \) to obtain a simple solution: \( c_2 = -1.5, c_3 = .5 \).

Thus,

\[
y - v_1 = -1.5(v_2 - v_1) + .5(v_3 - v_1), \text{ so that } y = 2v_1 - 1.5v_2 + .5v_3
\]

Another solution is found by setting \( c_4 = 1 \). In this case, \( c_2 = 0 \), and

\[
y - v_1 = -2(v_3 - v_1) + 1(v_4 - v_1), \text{ so that } y = 2v_1 - 2v_3 + v_4
\]

If \( c_4 = 3 \), then

\[
y - v_1 = 3(v_3 - v_1) - 7(v_4 - v_1) + 3(v_4 - v_1), \text{ and } y = 2v_1 + 3v_2 - 7v_3 + 3v_4
\]

Of course, many other answers are possible. Note that in all cases, the weights in the linear combination sum to 1.
7. The matrix \[ \begin{bmatrix} v_1 & v_2 & v_3 & p_1 & p_2 & p_3 \end{bmatrix} \]

reduces to

\[
\begin{bmatrix}
1 & 0 & 2 & 2 \\
0 & 1 & 0 & 1 & -4 & 2 \\
0 & 0 & 1 & 1 & 0 & -3 \\
0 & 0 & 0 & 0 & 0 & -5
\end{bmatrix}
\]

Parts (a), (b), and (c) use columns 4, 5, and 6, respectively, as the “augmented” column.

a. \( p_1 = 2v_1 + v_2 - v_3 \), so \( p_1 \) is in \( \text{Span} \ S \). The weights do not sum to 1, so \( p_1 \not\in \text{aff} \ S \)

b. \( p_2 = 2v_1 - 4v_2 + 3v_3 \), so \( p_2 \) is in \( \text{Span} \ S \). The weights sum to 1, so \( p_2 \in \text{aff} \ S \).

c. \( p_3 \not\in \text{Span} \ S \) because \( 0 \neq -5 \) (the fourth equation), so \( p_3 \) cannot possibly be in \( \text{aff} \ S \).

13. \( \text{Span} \ \{v_2 - v_1, v_3 - v_1\} \) is a plane if and only if \( \{v_2 - v_1, v_3 - v_1\} \) is linearly independent. Suppose \( c_2 \) and \( c_3 \) satisfy \( c_2(v_2 - v_1) + c_3(v_3 - v_1) = 0 \).

Then \( c_2v_2 + c_3v_3 - (c_2 + c_3)v_1 = 0 \). Then \( c_2 = c_3 = 0 \), because \( \{v_1, v_2, v_3\} \) is a linearly independent set. This shows that \( \{v_2 - v_1, v_3 - v_1\} \) is a linearly independent set.

Thus, \( \text{Span} \ \{v_2 - v_1, v_3 - v_1\} \) is a plane in \( \mathbb{R}^3 \).

19. If \( p, q \in f(S) \), then there exist \( r, s \in S \) such that \( f(r) = p \) and \( f(s) = q \). Given any \( t \in \mathbb{R} \), we must show that \( z = (1 - t)p + t q \) is in \( f(S) \). Since \( f \) is linear,

\[
z = (1 - t)p + t q = (1 - t)f(r) + t f(s) = f((1 - t)r + t s)
\]

Since \( S \) is affine, \((1 - t)r + t s \in S \). Thus, \( z \) is in \( S \) and \( f(S) \) is affine.

25. Since \( (A \cap B) \subseteq A \), it follows from Exercise 22 that \( \text{aff} \ (A \cap B) \subseteq \text{aff} \ A \).

Similarly, \( \text{aff} \ (A \cap B) \subseteq \text{aff} \ B \), so \( \text{aff} \ (A \cap B) \subseteq (\text{aff} A \cap \text{aff} B) \).
MATLAB Affine Combinations

Given points $v_1, v_2, v_3, v_4,$ and $y$, use $A = [v_2-v_1, v_3-v_1, v_4-v_1, y-v_1]$ to translate the points by $v_1$. To write $y$ as an affine combination of $v_1, v_2, v_3,$ and $v_4$, row reduce the matrix $A$ to write $y-v_1$ as a linear combination of $v_2-v_1, v_3-v_1,$ and $v_4-v_1$. Then solve for $y$.

If the chosen points $b_1, b_2, ..., b_n$ form a basis for $\mathbb{R}^n$, then the calculations are less involved. For any $y$ in $\mathbb{R}^n$, $y$ is a unique linear combination of $b_1, b_2, ..., b_n$. This combination is an affine combination of the $b$’s if and only if the weights sum to one. To determine if points $p_1$ or $p_2$ are affine combinations of a basis $b_1, b_2, b_3$ you can form the matrix $A = [b_1, b_2, b_3, p_1, p_2]$. If $C = \text{ref}(A)$, the MATLAB command $\text{sum}(C(:,4))$ will sum the weights in the fourth column of $C$ whereas the command $\text{sum}(C)$ will sum the entries in all of the columns of $C$.

The standard homogeneous form of a point $v$ is $[v;1]$. For calculation purposes it is easier to move the rows of 1’s to the top in the first step. To use the homogeneous form of the points given by $A = [v_1 \ v_2 \ v_3 \ p]$, the command $A = [\text{ones}(1,4); A]$ will place a row of 1’s in the top row. For an arbitrary number of columns, use $A = [\text{ones}(1, \text{size}(A,2)); A]$. The commands $C = \text{ref}(A)$ and $\text{sum}(C)$ will again be helpful here.

8.2 AFFINE INDEPENDENCE

Affine dependence of a set of vectors is a restricted type of linear dependence, which means that affine independence is a weaker condition than linear independence—and this is just what is needed for applications to computer graphics.

KEY IDEAS

The important definitions here are affine dependence, affine independence, and barycentric coordinates. Theorem 5 concerns affine dependence. Theorem 6 provides an important property of affinely independent sets, which leads to barycentric coordinates of a point.

STUDY NOTES

Again it is important to link the new concepts you are learning to concepts you already know. It may be helpful to review Sections 1.7 and 4.4 before continuing with this section. How are barycentric coordinates related to the coordinates of a vector relative to a basis?

Copyright © 2016 Pearson Education, Inc.
Notice that \( \{v_1, \ldots, v_p\} \) is an affinely dependent set if and only if \( \{v_2 - v_1, \ldots, v_p - v_1\} \) is a linearly dependent set. Use this idea both computationally and theoretically to help you solve problems in this section.

Finding barycentric coordinates is like finding the coordinates of a vector relative to a basis, except that you need an additional equation to ensure that the sum of the coefficients is 1. This adds a row of ones to the bottom of the augmented matrix you would use to find the coordinates of the vector relative to a basis.

**SOLUTIONS TO EXERCISES**

1. Let \( v_1 = \begin{bmatrix} 3 \\ -3 \end{bmatrix}, v_2 = \begin{bmatrix} 0 \\ 6 \end{bmatrix}, v_3 = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \). Then \( v_2 - v_1 = \begin{bmatrix} -3 \\ 9 \end{bmatrix}, v_3 - v_1 = \begin{bmatrix} -1 \\ 3 \end{bmatrix} \). Since \( v_3 - v_1 \) is a multiple of \( v_2 - v_1 \), these two points are linearly dependent. By Theorem 5, \( \{v_1, v_2, v_3\} \) is affinely dependent. Note that \( (v_2 - v_1) - 3(v_3 - v_1) = 0 \). A rearrangement produces the affine dependence relation \( 2v_1 + v_2 - 3v_3 = 0 \). (Note that the weights sum to zero.) Geometrically, \( v_1, v_2, \) and \( v_3 \) are collinear.

7. Denote the given points as \( v_1, v_2, v_3, \) and \( p \). Row reduce the augmented matrix for the equation \( x_1 \bar{v}_1 + x_2 \bar{v}_2 + x_3 \bar{v}_3 = \bar{p} \). Remember to move the bottom row of ones to the top as the first step to simplify the arithmetic by hand.

\[
\begin{bmatrix}
\bar{v}_1 & \bar{v}_2 & \bar{v}_3 & \bar{p} \\
1 & 1 & 1 & 1 \\
1 & 2 & 1 & 5 \\
-1 & 1 & 2 & 4 \\
2 & 0 & -2 & -2 \\
1 & 1 & 0 & 2
\end{bmatrix} \sim 
\begin{bmatrix}
1 & 0 & 0 & -2 \\
1 & 0 & 1 & 4 \\
0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

Thus, \( x_1 = -2, x_2 = 4, x_3 = -1 \), and \( \bar{p} = -2\bar{v}_1 + 4\bar{v}_2 - \bar{v}_3 \), so \( p = -2v_1 + 4v_2 - v_3 \), and the barycentric coordinates are \((-2, 4, -1)\).

**ALTERNATE SOLUTION:**

This problem can also be solved by “translating” it to the origin. That is, compute \( v_2 - v_1, v_3 - v_1, \) and \( p - v_1 \), find weights \( c_2 \) and \( c_3 \) such that

\[
c_2(v_2 - v_1) + c_3(v_3 - v_1) = p - v_1
\]

and then write \( p = (1 - c_2 - c_3)v_1 + c_2v_2 + c_3v_3 \). Here are the calculations for Exercise 7:
\[ v_2 - v_1 = \begin{bmatrix} 2 \\ 1 \\ -1 \\ 0 \\ 1 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad v_3 - v_1 = \begin{bmatrix} 1 \\ 2 \\ -1 \\ -2 \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ -2 \\ -4 \end{bmatrix}, \]

\[ p - v_1 = \begin{bmatrix} 5 \\ 4 \\ -2 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 5 \\ -4 \\ 1 \end{bmatrix} \]

\[
\left[ v_2 - v_1 \quad v_3 - v_1 \quad p - v_1 \right] \\
= \begin{bmatrix} 1 & 0 & 4 \\ 2 & 3 & 5 \\ -2 & -4 & -4 \\ 0 & -1 & 1 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & 4 \\ 2 & 3 & 5 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}
\]

Thus \( p - v_1 = 4(v_2 - v_1) - 1(v_3 - v_1), \) and \( p = -2v_1 + 4v_2 - v_3. \)

9.  
   a.  See Theorem 5 and the discussion just before Example 1.
   b.  See Theorem 5.
   c.  See the definition at the beginning of this section.
   d.  See the definition after Theorem 6, and see Figure 5.
   e.  See Example 6.

13. If \( \{v_1, v_2\} \) is affinely dependent, then there exist \( c_1 \) and \( c_2 \), not both zero, such that 
    \( c_1 + c_2 = 0, \) and \( c_1v_1 + c_2v_2 = 0. \) Then \( c_1 = -c_2 \neq 0 \) and \( c_1v_1 = -c_2v_2 = c_1v_2, \) which 
    implies that \( v_1 = v_2. \) Conversely, if \( v_1 = v_2, \) let \( c_1 = 1 \) and \( c_2 = -1. \) Then \( c_1v_1 + c_2v_2 = v_1 + (-1)v_1 = 0 \) and \( c_1 + c_2 = 0, \) which shows that \( \{v_1, v_2\} \) is affinely dependent.

19. If \( \{p_1, p_2, p_3\} \) is an affinely dependent set, then there exist scalars \( c_1, c_2, \) and \( c_3, \) not 
    all zero, such that \( c_1p_1 + c_2p_2 + c_3p_3 = 0 \) and \( c_1 + c_2 + c_3 = 0. \) Next, apply the linear 
    transformation \( f \) to both sides of the vector equation, and obtain 
    
    \[ c_1f(p_1) + c_2f(p_2) + c_3f(p_3) = f(c_1p_1 + c_2p_2 + c_3p_3) = f(0) = 0, \]
    
    since \( f \) is linear. This shows that the set \( \{f(p_1), f(p_2), f(p_3)\} \) is also affinely 
    dependent.
8.3 CONVEX COMBINATIONS

A convex combination of a set of vectors is a linear combination using nonnegative weights whose sum is 1. So a convex combination is a special type of affine combination.

KEY IDEAS

The important terms here are convex combination and convex hull. Theorem 7 concerns convex combinations, Theorems 8 and 9 study properties of intersections of convex sets, and Theorem 10 gives an upper bound on the number of point required when writing a given point as a linear combination of points in a set.

SOLUTIONS TO EXERCISES

1. The set \( V = \left\{ \begin{bmatrix} 0 \\ y \end{bmatrix} : 0 \leq y < 1 \right\} \) is the vertical line segment from (0,0) to (0,1) that includes (0,0) but not (0,1). The convex hull of \( S \) includes each line segment from a point in \( V \) to the point (2,0), as shown in the figure. The dashed line segment along the top of the shaded region indicates that this segment is not in \( \text{conv} \ S \), because (0,1) is not in \( S \).
7. Let \( \mathbf{v}_1 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \mathbf{v}_2 = \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \mathbf{v}_3 = \begin{bmatrix} 4 \\ 1 \end{bmatrix}, \mathbf{p}_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \mathbf{p}_2 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}, \mathbf{p}_3 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \mathbf{p}_4 = \begin{bmatrix} 0 \\ 2 \end{bmatrix}, \) and 
\( T = \{ \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \}. \)

a. Use homogeneous forms for all of the vectors, by adding a third row to each vector, with 1 in the third entry. Next, create an augmented matrix (with four augmented columns) to write the homogeneous forms of \( \mathbf{p}_1, \ldots, \mathbf{p}_4 \) in terms of the homogeneous forms of \( \mathbf{v}_1, \mathbf{v}_2, \) and \( \mathbf{v}_3. \) Then, to simplify calculations, interchange rows 1 and 3, to put a row of 1’s at the top. Finally, row reduction produces:

\[
\begin{bmatrix}
\mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 & \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \mathbf{p}_4 \\
\tilde{\mathbf{v}}_1 & \tilde{\mathbf{v}}_2 & \tilde{\mathbf{v}}_3 & \tilde{\mathbf{p}}_1 & \tilde{\mathbf{p}}_2 & \tilde{\mathbf{p}}_3 & \tilde{\mathbf{p}}_4 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & 2 & 4 & 2 & 3 & 2 & 0 \\
0 & 3 & 1 & 1 & 2 & 0 & 2 \\
0 & 0 & 1 & 1 & 2 & \frac{3}{4} & -\frac{1}{4}
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{2} & \frac{1}{2} \\
0 & 1 & 0 & \frac{1}{6} & \frac{1}{2} & -\frac{1}{4} & \frac{3}{4} \\
0 & 0 & 1 & \frac{1}{2} & \frac{1}{2} & \frac{3}{4} & -\frac{1}{4}
\end{bmatrix}
\]

The first four columns reveal that \( \frac{1}{3} \tilde{\mathbf{v}}_1 + \frac{1}{6} \tilde{\mathbf{v}}_2 + \frac{1}{2} \tilde{\mathbf{v}}_3 = \tilde{\mathbf{p}}_1 \) and 
\( \frac{1}{3} \mathbf{v}_1 + \frac{1}{6} \mathbf{v}_2 + \frac{1}{2} \mathbf{v}_3 = \mathbf{p}_1. \) Thus column 4 contains the barycentric coordinates of \( \mathbf{p}_1 \) relative to the triangle determined by \( T. \) Similarly, column 5 (as an augmented column) contains the barycentric coordinates of \( \mathbf{p}_2, \) column 6 contains the barycentric coordinates of \( \mathbf{p}_3, \) and column 7 contains the barycentric coordinates of \( \mathbf{p}_4. \)

b. \( \mathbf{p}_3 \) and \( \mathbf{p}_4 \) are outside \( \text{conv} \ T, \) because in each case at least one of the barycentric coordinates is negative. \( \mathbf{p}_1 \) is inside \( \text{conv} \ T, \) because all of its barycentric coordinates are positive. \( \mathbf{p}_2 \) is on the edge \( \mathbf{v}_1 \mathbf{v}_2 \) of \( \text{conv} \ T, \) because its barycentric coordinates are nonnegative and its first coordinate is 0.

11. a. See the definition at the beginning of this section.

b. See Theorem 7.

c. Try to construct a counterexample in \( \mathbb{R}^2. \)

13. If \( \mathbf{p}, \mathbf{q} \in f(S), \) then there exist \( \mathbf{r}, \mathbf{s} \in S \) such that \( f(\mathbf{r}) = \mathbf{p} \) and \( f(\mathbf{s}) = \mathbf{q}. \) The goal is to show that the line segment \( \mathbf{y} = (1 - t)\mathbf{p} + t\mathbf{q}, \) for \( 0 \leq t \leq 1, \) is in \( f(S). \) Since \( f \) is linear,

\[
\mathbf{y} = (1 - t)\mathbf{p} + t\mathbf{q} = (1 - t)f(\mathbf{r}) + tf(\mathbf{s}) = f((1 - t)\mathbf{r} + t\mathbf{s})
\]

Since \( S \) is convex, \( (1 - t)\mathbf{r} + t\mathbf{s} \in S \) for \( 0 \leq t \leq 1. \) Thus \( \mathbf{y} \in f(S) \) and \( f(S) \) is convex.
19. a. Since $A \subset (A \cup B)$, Exercise 18 shows that $\text{conv } A \subset \text{conv } (A \cup B)$. Similarly, $\text{conv } B \subset \text{conv } (A \cup B)$. Thus, $[(\text{conv } A) \cup (\text{conv } B)] \subset \text{conv } (A \cup B)$.

b. One possibility is to let $A$ be two adjacent corners of a square and $B$ be the other two corners. Then $(\text{conv } A) \cup (\text{conv } B)$ consists of two opposite sides of the square, but $\text{conv } (A \cup B)$ is the whole square.

8.4 HYPERPLANES

Lines play an important but subtle role in ordinary plane geometry because a line divides the plane into two parts. In three dimensions, a plane divides $\mathbb{R}^3$ into two regions. In higher dimensions, a hyperplane divides $\mathbb{R}^n$ into two regions.

KEY IDEAS

Section 8.1 defined a hyperplane in $\mathbb{R}^n$ as a translate of a subspace of dimension $n - 1$. Computations with hyperplanes in Section 8.4 require an implicit description, involving a linear functional, defined before Example 1. Several important terms from topology appear in a box just before Example 7. Finally, the text explains what it means for a hyperplane to separate two sets. Theorems 12 and 13 provide sufficient conditions for when two sets can be separated.

STUDY NOTES

It is convenient to represent a hyperplane in several different ways, and you want to be able to move comfortably between these different representations.

The implicit equation $a_1x_1 + a_2x_2 + \cdots + a_nx_n = d$ is one way to describe a hyperplane. Notice that $f(x) = a_1x_1 + a_2x_2 + \cdots + a_nx_n$ is a linear transformation from $\mathbb{R}^n$ to $\mathbb{R}$, and such a transformation is referred to as a linear functional.

The set of solutions to $a_1x_1 + a_2x_2 + \cdots + a_nx_n = 0$ corresponds to the null space of the matrix $A = [a_1 \ a_2 \ \ldots \ \ a_n]$, and hence can be written in parametric form (see Section 1.5). The solutions to $a_1x_1 + a_2x_2 + \cdots + a_nx_n = d$ can be described by adding a particular solution $p$ to the null space of $A$. 

Copyright © 2016 Pearson Education, Inc.
Notice also that
\[
\begin{bmatrix}
a_1 \\
a_2 \\
\vdots \\
\vdots \\
a_n \\
\end{bmatrix} \cdot 
\begin{bmatrix}
x_1 \\
x_2 \\
\vdots \\
\vdots \\
x_n \\
\end{bmatrix} = a_1x_1 + a_2x_2 + \cdots + a_nx_n,
\]
so the equation
\[
a_1x_1 + a_2x_2 + \cdots + a_nx_n = d
\]
can be expressed as \( \mathbf{n} \cdot \mathbf{x} = d \) where \( \mathbf{n} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \). Since the solutions to \( \mathbf{n} \cdot \mathbf{x} = 0 \) describe the set of vectors orthogonal to \( \mathbf{n} \), we refer to \( \mathbf{n} \) as the normal vector to the hyperplane \( \mathbf{n} \cdot \mathbf{x} = d \).

**SOLUTIONS TO EXERCISES**

1. \( \mathbf{v}_2 - \mathbf{v}_1 = \begin{bmatrix} 3 \\ 1 \\ -1 \\ 4 \end{bmatrix} - \begin{bmatrix} -1 \\ 4 \\ -3 \end{bmatrix} = \begin{bmatrix} 4 \\ -3 \end{bmatrix} \), so let \( \mathbf{n} = \begin{bmatrix} 3 \\ 4 \end{bmatrix} \). \( f(x_1, x_2) = 3x_1 + 4x_2 \) and
\[
d = f(\mathbf{v}_1) = 3(-1) + 4(4) = 13.
\]
This is easy to check by verifying that \( f(\mathbf{v}_2) \) is also 13.

7. a. Let \( \mathbf{v}_1 = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \), \( \mathbf{v}_2 = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \), \( \mathbf{v}_3 = \begin{bmatrix} -1 \\ 5 \end{bmatrix} \), \( \mathbf{n} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \) and compute the translated points
\[
\mathbf{v}_2 - \mathbf{v}_1 = \begin{bmatrix} 1 \\ -3 \end{bmatrix}, \quad \mathbf{v}_3 - \mathbf{v}_1 = \begin{bmatrix} -2 \\ -2 \\ 2 \end{bmatrix}.
\]

To solve the system of equations \( (\mathbf{v}_2 - \mathbf{v}_1) \cdot \mathbf{n} = 0 \) and \( (\mathbf{v}_3 - \mathbf{v}_1) \cdot \mathbf{n} = 0 \), reduce the augmented matrix for a system of two equations with three variables.
\[
\begin{bmatrix}
a & 3 & -2 \\
1 & b & c \\
\end{bmatrix} = 0, \quad \begin{bmatrix}
a & -2 & 3 \\
-2 & b & c \\
\end{bmatrix} = 0.
\]
Row operations show that
\[
\begin{bmatrix}
1 & 3 & -2 & 0 \\
-2 & -3 & 2 & 0 \\
\end{bmatrix} \sim \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 3 & -2 & 0 \\
\end{bmatrix}.
\]
A suitable normal vector is \( \mathbf{n} = \begin{bmatrix} 0 \\ 2 \\ 3 \end{bmatrix} \).
b. The linear functional is \( f(x_1, x_2, x_3) = 2x_2 + 3x_3 \), so \( d = f(1, 1, 3) = 2 + 9 = 11 \).

As a check, evaluate \( f \) at the other two points on the hyperplane:

\[
f(2, 4, 1) = 8 + 3 = 11 \quad \text{and} \quad f(-1, -2, 5) = -4 + 15 = 11.
\]

13. \( H_1 = \{ \mathbf{x} : \mathbf{n}_1 \cdot \mathbf{x} = d_1 \} \) and \( H_2 = \{ \mathbf{x} : \mathbf{n}_2 \cdot \mathbf{x} = d_2 \} \). Since \( \mathbf{p}_1 \in H_1, d_1 = \mathbf{n}_1 \cdot \mathbf{p}_1 = 4 \).

Similarly, \( d_2 = \mathbf{n}_2 \cdot \mathbf{p}_2 = 22 \). Solve the simultaneous system \([1 \ 2 \ 4 \ 2] \mathbf{x} = 4\) and \([2 \ 3 \ 1 \ 5] \mathbf{x} = 22\):

\[
\begin{bmatrix}
1 & 2 & 4 & 2 \\
2 & 3 & 1 & 5
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & -10 & 4 & 32 \\
0 & 1 & 7 & -1 & -14
\end{bmatrix}.
\]

The general solution provides one set of vectors, \( \mathbf{p}, \mathbf{v}_1, \) and \( \mathbf{v}_2 \). Other choices are possible.

\[
\mathbf{x} = \begin{bmatrix} 32 \\ -14 \\ 0 \\ 0 \end{bmatrix} + x_3 \begin{bmatrix} 10 \\ -7 \\ 1 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} -4 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \mathbf{p} + x_3 \mathbf{v}_1 + x_4 \mathbf{v}_2, \quad \text{where}
\]

\[
\mathbf{p} = \begin{bmatrix} 32 \\ -14 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{v}_1 = \begin{bmatrix} 10 \\ -7 \\ 1 \\ 0 \end{bmatrix}, \quad \text{and} \quad \mathbf{v}_2 = \begin{bmatrix} -4 \\ 1 \\ 0 \\ 1 \end{bmatrix}.
\]

Then \( H_1 \cap H_2 = \{ \mathbf{x} : \mathbf{x} = \mathbf{p} + x_3 \mathbf{v}_1 + x_4 \mathbf{v}_2 \} \).

19. Theorem 3 in Section 6.1 says that \((\text{Col } B)^\perp = \text{Nul } B^T\). Since the two columns of \( B \) are clearly linear independent, the rank of \( B \) is 2, as is the rank of \( B^T \). So \( \dim \text{Nul } B^T = 1 \), by the Rank Theorem, since there are three columns in \( B^T \). This means that \( \text{Nul } B^T \) is one-dimensional, and any nonzero vector \( \mathbf{n} \) in \( \text{Nul } B^T \) will be orthogonal to \( H \) and can be used as its normal vector. Solve the linear system \( B^T \mathbf{x} = \mathbf{0} \) by row reduction to find a basis for \( \text{Nul } B^T \):

\[
\begin{bmatrix}
1 & 4 & -7 & 0 \\
0 & 2 & -6 & 0
\end{bmatrix}
\sim
\begin{bmatrix}
1 & 0 & 5 & 0 \\
0 & 1 & -3 & 0
\end{bmatrix} \Rightarrow \mathbf{n} = \begin{bmatrix} -5 \\ 3 \\ 1 \end{bmatrix}.
\]

Now, let \( f(x_1, x_2, x_3) = -5x_1 + 3x_2 + x_3 \). Since the hyperplane is a subspace, it goes through the origin and \( d \) must be 0.

The solution is easy to check by evaluating \( f \) at each of the columns of \( B \).
8.5 POLYTOPES

This section introduces a class of compact convex sets that are used to study optimization problems in a variety of fields, including engineering design, linear programming, and business management.

KEY IDEAS

The important definitions here are polytope, supporting hyperplane, extreme point (of a convex set), profile (of a convex set), minimal representation of a polytope, simplex, and barycentric coordinates. Theorem 14 describes the minimal representation of a polytope. Theorem 16 is a fundamental result for linear programming. (Linear programming is the subject of Chapter 9, which is available online for colleges and universities that have a second course in applications of linear algebra). The second part of Section 8.5 discusses interesting geometric objects and shows how to visualize two special polytopes in four dimensions.

STUDY NOTES

In this section, it will help to think geometrically and to imagine or draw the shapes of the objects involved.

SOLUTIONS TO EXERCISES

1. Evaluate each linear functional at each of the three extreme points of $S$. Then select the extreme point(s) that give the maximum value of the functional.
   a. $f(p_1) = 1$, $f(p_2) = -1$, and $f(p_3) = -3$, so $m = 1$ at $p_1$.
   b. $f(p_1) = 1$, $f(p_2) = 5$, and $f(p_3) = 1$, so $m = 5$ at $p_2$.
   c. $f(p_1) = -3$, $f(p_2) = -3$, and $f(p_3) = 5$, so $m = 5$ at $p_3$.

7. The three inequalities are (a) $x_1 + 3x_2 \leq 18$, (b) $x_1 + x_2 \leq 10$, and (c) $4x_1 + x_2 \leq 28$.
   Line (a) goes from $(0,6)$ to $(18,0)$. Line (b) goes from $(0,10)$ to $(10,0)$. And line (c) goes from $(0,28)$ to $(7,0)$. One vertex is $(0,0)$. The $x_1$-intercepts (when $x_2 = 0$) are 18, 10, and 7, so $(7,0)$ is a vertex. The $x_2$-intercepts (when $x_1 = 0$) are 6, 10, and 28, so $(0,6)$ is a vertex. All three lines go through $(6,4)$, so $(6,4)$ is a vertex. The minimal representation is $\begin{bmatrix} 0 & \begin{bmatrix} 7 & 6 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 4 & 6 \end{bmatrix} \end{bmatrix}$.

13. a. To determine the number of $k$-faces of the 5-dimensional hypercube $C^5$, look at the pattern that is followed in building $C^4$ from $C^3$. For example, the 2-faces in
C^4 include the 2-faces of C^3 and the 2-faces in the translated image of C^3. In addition, there are the 1-faces of C^3 that are “stretched” into 2-faces. In general, the number of k-faces in C^n equals twice the number of k-faces in C^{n-1} plus the number of (k-1)-faces in C^{n-1}. The pattern is:

\[ f_k(C^n) = 2f_k(C^{n-1}) + f_{k-1}(C^{n-1}). \]

For k = 0, 1, …, 4, and n = 5, this gives \( f_0(C^5) = 32, \) \( f_1(C^5) = 80, \) \( f_2(C^5) = 80, \) \( f_3(C^5) = 40, \) and \( f_4(C^5) = 10. \) These numbers satisfy Euler’s formula since,

\[ 32 - 80 + 80 - 40 + 10 = 2. \]

b. The general formula is \( f_k(C^n) = 2^{n-k} \binom{n}{k}, \) where \( \binom{n}{k} = \frac{a!}{b!(a-b)!} \) is the binomial coefficient.

17. a. The facets of a cube are its faces. How many are there?
   b. See Theorem 14.
   c. See Theorem 16.
   d. See Euler’s formula with \( n = 2. \)

19. Let S be convex and choose x in \( cS + dS, \) where \( c > 0 \) and \( d > 0. \) Then there exist \( s_1 \) and \( s_2 \) in S such that \( x = cs_1 + ds_2. \) But then

\[ x = cs_1 + ds_2 = (c+d)\left( \frac{c}{c+d}s_1 + \frac{d}{c+d}s_2 \right). \]

Now \( \frac{c}{c+d} \) and \( \frac{d}{c+d} \) are both nonnegative and sum to one. Since S is convex,

\[ (c+d)\left( \frac{c}{c+d}s_1 + \frac{d}{c+d}s_2 \right) \in S. \] Thus \( x \in (c+d)S. \)

Conversely, let \( x \in (c+d)S. \) Then \( x = (c+d)s \) for some \( s \in S. \) But then

\[ x = cs + ds \in cS + dS, \] as desired.
This section moves beyond lines and planes to curves and surfaces used in engineering and CAD (computer-aided design).

**KEY IDEAS**

Control points for a curve are not necessarily points on the curve itself, but rather are points that appear in linear combinations whose weights are simple polynomials. Exercises 21 to 24 in Section 8.3 introduced quadratic and cubic Bézier curves. Section 8.6 studies matrix equations for both Bézier curves and Bézier surfaces. The practice problems in Section 8.6 introduce a related family of curves called B-splines.

**STUDY NOTES**

Think of a Bézier curve as a linear combination of the points \( p_0, p_1, \ldots, p_n \), with coefficients that contain the variable \( t \).

**SOLUTIONS TO EXERCISES**

1. The original curve is \( x(t) = (1 - t)^3 p_0 + 3t(1 - t)^2 p_1 + 3t^2(1 - t)p_2 + t^3 p_3 \) \((0 \leq t \leq 1)\). Since the curve is determined by its control points, it seems reasonable that to translate the curve, one should translate the control points. In this case, the new Bézier curve \( y(t) \) would have the equation

\[
y(t) = (1 - t)^3 (p_0 + b) + 3t(1 - t)^2 (p_1 + b) + 3t^2(1 - t)(p_2 + b) + t^3 (p_3 + b)
\]

A routine algebraic calculation verifies that

\[
(1 - t)^3 + 3t(1 - t)^2 + 3t^2(1 - t) + t^3 = 1
\]

for all \( t \). Thus \( y(t) = x(t) + b \) for all \( t \), and translation by \( b \) maps a Bézier curve into a Bézier curve.

7. From Exercise 3(b),

\[
x''(0) = 6(p_0 - p_1) + 6(p_2 - p_1) \quad \text{and} \quad x''(1) = 6(p_1 - p_2) + 6(p_3 - p_2)
\]

Use \( x''(0) \) with the control points for \( y(t) \), to get

\[
y''(0) = 6(p_3 - p_4) + 6(p_5 - p_4)
\]

Set \( x''(1) = y''(0) \) and divide by 6, to get

\[
(p_1 - p_2) + (p_3 - p_2) = (p_3 - p_4) + (p_5 - p_4) \quad (\star)
\]
Since the curve is $C^1$ continuous at $p_3$, the point $p_3$ is the midpoint of the segment from $p_2$ to $p_4$, by Exercise 5(a). Thus $p_3 = \frac{1}{2}(p_2 + p_4)$, which leads to $p_4 - p_3 = p_3 - p_2$. Substitution into (*) gives $(p_1 - p_2) + (p_3 - p_2) = -(p_3 - p_2) + p_5 - p_4$

$$p_1 + 2(p_3 - p_2) + p_4 = p_5$$

Finally, again from $C^1$ continuity, $p_4 = p_3 + p_3 - p_2$. Thus,

$$p_5 = p_3 + (p_1 - p_2) + 3(p_3 - p_2)$$

So $p_4$ and $p_5$ are uniquely determined by $p_1$, $p_2$, and $p_3$. Only $p_6$ can be chosen arbitrarily.

11. a. See equation (2). b. See Example 1. c. See Example 2.

13. a. From (12), $q_1 - q_0 = \frac{1}{2}(p_1 - p_0) = \frac{1}{2}p_1 - \frac{1}{2}p_0$. Since $q_0 = p_0$, $q_1 = \frac{1}{2}(p_1 + p_0)$.

b. From (13), $8(q_3 - q_2) = -p_0 - p_1 + p_2 + p_3$. So $8q_3 + p_0 + p_1 - p_2 - p_3 = 8q_2$.

c. Use (8) to substitute for $8q_3$, and obtain

$$8q_2 = (p_0 + 3p_1 + 3p_2 + p_3) + p_0 + p_1 - p_2 - p_3 = 2p_0 + 4p_1 + 2p_2$$

Then divide by 8, regroup the terms, and use part (a) to obtain

$$q_2 = \frac{1}{4}p_0 + \frac{1}{2}p_1 + \frac{1}{4}p_2 = \frac{1}{4}(p_0 + \frac{1}{4}p_1) + \frac{1}{4}(p_1 + \frac{1}{4}p_2) = \frac{1}{2}q_1 + \frac{1}{2}(p_1 + p_2)$$

$$= \frac{1}{2}(q_1 + \frac{1}{2}(p_1 + p_2))$$

17. a. $r_0 = p_0$, $r_1 = \frac{p_0 + 2p_1}{3}$, $r_2 = \frac{2p_1 + p_2}{3}$, $r_3 = p_2$

b. Hint: Write the standard formula (7) in this section, with $r_i$ in place of $p_i$ for $i = 1, \ldots, 4$, and then replace $r_0$ and by $p_0$ and $p_2$, respectively:

$$x(t) = (1 - 3t + 3t^2 - t^3)p_0 + (3t - 6t^2 + 3t^3)r_1 + (3t^2 - 3t^3)r_2 + t^3p_2 \quad (iii)$$

Use the formulas for $r_1$ and $r_2$ from part (a) to examine the second and third terms in this expression for $x(t)$
The MATLAB program **bezier.m** draws graphs of Bézier curves of degrees 2, 3, and 4 in the plane. Type `bezier` at the command prompt. Enter the degree of the desired curve and the geometry matrix representing the control points. For a cubic curve, the geometry matrix will need to be a 2x4 matrix corresponding to the four control points. The program includes the command `hold on`, so rerunning the program will allow MATLAB to display any new graphs in the same window. If you wish to start a new figure, close the existing figure before running the program again.

**Chapter 8  GLOSSARY CHECKLIST**

Check your knowledge by attempting to write definitions of the terms below. Then compare your work with the definitions given in the text’s Glossary. Ask your instructor which definitions, if any, might appear on a test.

- **affine combination**: A linear combination ….
- **affine dependence relation**: An equation of the form …where….
- **affine hull of a set S**: The set of all ….
- **affinely dependent set**: A set \{\(v_1, \ldots, v_p\)\} ….
- **affinely independent set**: A set \{\(v_1, \ldots, v_p\)\} ….
- **affine set**: A set \(S\) of points such that ….
- **affine transformation**: A mapping ….
- **barycentric coordinates**: The set of weights ….
- **boundary point of a set S**: A point \(p\) such that ….
- **bounded set**: A set that is ….
- **closed ball**: A set ….
- **convex combination**: A linear combination of vectors such that ….
- **convex hull**: The set of all ….
- **convex set**: A set with the property that ….
- **dimension of a flat**: The dimension of the corresponding ….
extreme point of a set $S$  A point $p$ in $S$ such that 

cancel

face  A nonempty subset 

cancel

facet  A ... dimensional face.  

cancel

flat  A translate of  

cancel

hypercube  Let $I_j$ be the line segment from the origin to the j-th standard basis vector. Then 

cancel

hyperplane  A flat of 

cancel

interior point  A point $p$ in $S$ such that 

cancel

linear functional  A linear transformation 

cancel

open ball  A set 

cancel

open set  A set 

cancel

polygon  A polytope in 

cancel

polyhedron  A polytope in 

cancel

polytope  The convex hull of 

cancel

positive combination  The set of linear combinations of a set of points with 

cancel

positive hull  The set of all 

cancel

profile  The set of 

cancel

quadratic Bezier curve  A curve whose description may be written in the form 

cancel

simplex  The convex hull of 

cancel

supporting hyperplane  A hyperplane 

cancel

tetrahedron  A three dimensional solid 

cancel
Technology Index of Procedures and Terms

The references here are to sections (not pages) that contain information about the use of technology with various procedures and objects considered in the text exercises. For MATLAB, consult the MATLAB boxes in the main Study Guide sections. For Maple, Mathematica, or the TI calculators, consult the corresponding sections in the appropriate technology appendix.

Access exercise data, 1.1, 1.3
Affine combinations, 8.1
Affine independence, 8.2
Augment a matrix, 1.3
Bezier curves, 8.6
Change-of-coordinates matrix, 4.7
Column of $A$, 2.1
Compare two vectors or matrices, 2.3
Compute a determinant, 3.2
Compute a product of numbers, 3.2
Condition number, 2.2
Construct a covariance matrix, 7.5
Construct an inverse, 2.2
Construct a matrix, 1.3, 2.4
Diagonalize a matrix, 5.3, 7.1
Display on screen,
  see format, 1.1, 1.6, 5.1, 5.8
Entry (in a matrix), 1.1, 2.1
Entrywise computation, 4.1, 5.6, 5.7, 6.6
Evaluate a polynomial at many points, 5.2
Exercise data, 1.1, 1.3
Extract diagonal entries, 3.2
Find a basis for a column space, 4.3
Find a basis for a null space, 4.3
Find a basis for an eigenspace, 5.1, 7.1
Find a characteristic polynomial, 5.2
Find eigenvalues and eigenvectors, 5.1
Find the zeros of a polynomial, 4.8
Function of a vector, 4.1, 6.6
Generate a sequence, 1.10, 5.6
Gram-Schmidt process, 6.4
Graph a polynomial, 5.2
Graph several functions, 4.1, 5.2, 6.8
Graph a trajectory, 5.6
Identity matrix, 2.1, 2.2
Inner product, 2.1, 6.1
Inverse power method, 5.8
LU factorization, 2.5
Matrix notation and operations, 2.1
Matrix power, 2.1
Matrix-vector multiplication, 1.4
Numbers in MATLAB,
  displayed, 1.1
  rational, 1.6, 5.1
Orthogonal projection, 6.2, 6.3, 6.4
Outer product, 2.1
Partitioned matrix, 2.4
Permutated LU factorization, 2.5
Power of a vector function, 4.1
Power method, 5.8
QR factorization, 6.4
Random matrix, 2.1, 2.3
Recall previous commands, 1.3, 1.10
Reduced echelon form, 2.8, 2.9, 4.3
Row of $A$, 2.1
Row operations, 1.1
Row reduction, 1.4
Scientific notation, 1.10
Select part of a matrix, 2.1
Sequence, generate, 1.10, 5.6
Singular value decomposition, 7.4
Solve $Ax = 0$, 1.5
Solve $Ax = b$, 1.4, 2.5
Special matrices, 2.1
Hilbert matrix, 2.3
Trajectory, 5.6
Test for orthogonality, 6.2
Transpose, 2.1
Introduction to MATLAB

MATLAB stands for MATrix LABoratory. Originally written by Cleve Moler for college linear algebra courses, MATLAB has evolved into a high-level language and interactive environment for linear algebra computations in science and industry all over the world. Using MATLAB in this course will save you time on homework, help you learn linear algebra, and give you a glimpse of how linear algebra is applied in practical work.

At the end of many Study Guide sections, MATLAB “boxes” will show you how to avoid routine arithmetic calculations, so you can focus on new concepts. Later in the course, you will run several programs for computations virtually impossible to do by hand. For instance, given a $32 \times 32$ matrix $A$ (which is small by today’s standards), MATLAB’s eigenvalue program provides important and accurate information about $A$ in a fraction of a second, after performing about one million arithmetic operations.

GETTING STARTED WITH MATLAB

You need access to the MATLAB program, either on a personal computer (PC or Mac) or on a shared computer network system at your school. If the school network is prepared for your course, someone may have already installed special MATLAB programs written for this course, along with data for about 850 exercises in the text. Accessing a matrix or some vectors for a problem requires only a few keystrokes. (This avoids data entry error and saves you time.) And then MATLAB performs the computations for you. Once you see how much time you can save, you’ll probably start using MATLAB for most of the numerical exercises!

If you plan to work at a computer away from school, you can purchase a Student Version of MATLAB at modest cost. You need to download the special programs and exercise data from the Web, at www.pearsonhighered.com/lay. These programs and data are in a folder called the Laydata5 Toolbox. Download the folder into the main MATLAB folder and then add the Laydata5 Toolbox to the MATLAB PATH. A ReadMe file in the Toolbox has instructions. When that is done, MATLAB will have immediate access to everything in the Toolbox.

The following paragraphs provide basic information about the MATLAB environment, including how to save and print a copy of your work. After that, the MATLAB boxes in this Guide will form your “lab manual” for the use of MATLAB during the course. (Begin with Section 1.1.) For your reference, an index of commands is included at the end of this appendix.
How to Start, Stop, And Run MATLAB

Once you start MATLAB, you will see the MATLAB logo as the program loads, followed by a prompt symbol `>>`, or `EDU>>`, in what is called the command window. Recent versions of MATLAB have at least four other windows including the Current Folder, the Workspace, and the Command History. The Study Guide focuses mainly on the Command Window. Type a MATLAB command at the prompt and press <Enter>. MATLAB’s response will follow. Here are some simple commands:

\[ x = 2 \times 3 \]
Sets up a variable \( x \) in your workspace and stores 6 in it.
The symbols `+`, `—`, `*`, `/`, and `^` (for exponents) are used for arithmetic.

\[ X = 5 \times x \]
Sets up a new variable with the value 30. MATLAB is case-sensitive.
Extra spaces are ignored. They were inserted here for readability.

\[ A = \begin{bmatrix} 1 & 2 & 3; & 4 & -5 & 6 \end{bmatrix} \]
Creates a matrix with 1,2,3 in the first row and 4, -5,6 in the second row. One or more spaces separate entries in each row.

\[ \text{cls1} \]
Requests data for exercises in Chapter 1 Section 1, for example.
See the MATLAB box at the end of Study Guide notes for Section 1.1.

\[ \text{who} \]
Lists the variables in your workspace (a portion of computer memory that stores the variables created since you turned on the program).

\[ \text{clear} \]
Removes all variables (and their contents) from the workspace.

\[ \text{help clear} \]
Tells you about the clear command.

\[ \text{help} \]
Displays a long list of topics for which help is available.

\[ \text{quit} \]
Immediately clears the workspace and terminates MATLAB.
\[ \text{exit} \] does the same.

You can place several commands on one line, if you wish, separated by commas or semicolons. A semicolon after a command instructs MATLAB not to display the result of the command.

To interrupt a computation which may be taking too long, press `Ctrl-C` in the Command Window.

How to Format MATLAB Output

Ordinarily, MATLAB displays up to 5 digits for each number and places a line feed between each object it displays. The following commands allow you to modify this standard display style.
format compact  Suppresses extra line-feeds, so you can see more output on one screen. Many students turn this on at the beginning of a work session.

format long  Displays all 15 digits that were used in the calculation.

format short  Displays only 5 digits.

format rat  Displays numbers as fractions, as accurately as possible.

format  Restores the standard display style.

Use help format to see other formatting options.

How to Save and Print a Copy of Your Work

You can save a record of every keystroke you make, along with MATLAB’s responses, in a file you choose by turning on the “diary” feature. For instance, if you select the filename project1.txt, use the commands:

```
diary project1.txt  Writes output (following this command) to the file named.
                   If the file already exists, the output is appended to the file.

diary off  Stops the recording

-diary on  Starts writing output again to project1.txt, named earlier.
            Places new output at the end of the file.
```

The file project1.txt will be located in the “current directory,” which initially is the main directory that holds MATLAB and all of its subdirectories. You can change this directory, but the easiest way to control where your file goes is to include a path at the front of the name. For example, if you have a USB flash drive in drive E, the command `diary E:\project1.txt` will store your work there. Alternatively, you can use the toolbar at the top of the screen to change the current directory.

Once your output is in a text file, you can use any convenient text editor to add comments to your work, delete extraneous output or errors in typing, and then print the file. If you want to print while working on a network, you may need to find out what special network commands are required.

This concludes the main part of the MATLAB appendix. You should be ready to use MATLAB with your homework (as soon as the Laydata5 Toolbox is installed). Enjoy yourself!
SCRIPT M-FILES

When you have some experience using MATLAB, you may wish to experiment with script M-files to prepare homework assignments and projects. The diary process works reasonably well for short homework assignments. However, for longer projects the diary file can be quite long and require extensive editing particularly if you make multiple attempts to get the project working.

A script M-file is simply a sequence of normal MATLAB commands stored in a text file that has “.m” as its extension. Instead of running these commands at the command prompt, you type the name of the script M-file at the prompt and press <Enter>. For example, if you name your script file project1.m and place it in the MATLAB working directory, then project1 becomes a new MATLAB “command.” Entering project1 at the prompt causes the commands in the file to be executed as if you had entered them one at a time.

The main advantage of using a script M-file instead of the diary to create a project report is that errors in the script can be corrected easily and the entire script run again with no effort. If you discover a mistake in the middle of a project that is being recorded by the diary, you have to repeat all the commands. (Later, you’ll also have to edit out the first set of commands that did not work properly.) A second reason for using a script is that you can reuse it later if you need to solve a similar problem. Just make a few appropriate edits, and you are ready to go.

How to Create a Script M-file

Click the New Script button on the Home tab to open the M-file Editor/Debugger. The first line of a script M-file should be

    echo on

Displays the commands along with results. Without this command, only the results will be displayed.

Place the MATLAB commands next. The last line of the file should be

    echo off

Turns off the previous echo command.

When you are finished, save the M-file in the current working directory.

You can modify an M-file using the M-file Editor/Debugger accessible from the toolbar or the File menu. Since an M-file is a plaintext file, you are free to use any text editor. If you do use another editor or you download an M-file, save the file as a text file with the .m extension.

Another way to create a script M-file is to use the Command History window. Highlight the commands which you have already typed in the Command History, right-click and select “Create Script.”
To run a script file you can either type the script name in the Command Window or click the Run button on the Editor tab at the top the Editor.

**How to Prepare a Final Report**

When the script M-file is running satisfactorily, you need to document your work with comments, which are statements in the file that begin with a percent sign. Comments are used to outline the steps in your calculations and to interpret the results of various computations. For example:

```
%The system is inconsistent, because 0 = 5/2 is not true.
```

MATLAB ignores all such comments, but your instructor will appreciate them.

To create a printed copy of your work, use the diary command. If your file is problem1.m, enter the following commands at the command prompt:

```
diary problem1.txt
problem1
```

Records all commands and output that follow. Executes the commands in project1 and displays all comments in the file on the screen. The diary captures these comments as well.

```
diary off
```

Now you have a text file that shows all your work and which you can edit and format in another program. Depending on the requirements of your course, you can either print the file for submission or attach the file in an email.

**Other Features**

MATLAB has an extensive help system built into it. Here are some procedures you may want to try with its help.

When creating graphs, use the menus and buttons in the figure window which allow you to add axes, labels, titles and other annotations to the graph. Save your file as a `.fig` file to keep your annotations since you may want to edit the figure later.

If you want to include the figure in a report, you may want to export the figure since what you see on the screen is not necessarily what you get in a file. Select Export setup from the figure’s File menu. There is a multitude of options, and the documentation center is there to help you.

MATLAB is capable of loading other common format files such as spreadsheet data. With your file in the “Current Folder” window, you can double-click on the file and follow the resulting prompts.
Index of MATLAB commands

\ (backslash), 2-27, 4-15, 6-16, 6-19
: (colon), 2-4
./ (divide entry-wise), 5-28
^ (exponent), 2-4, 4-3, 6-20
' (transpose), 2-4
↑ (up-arrow), 1-14, 1-42, 2-14, 5-25, 5-32

abs, 5-32
ans, 1-6, 1-19
bgauss, 1-19, 1-24, 2-9
bezier, 8-16
cond, 2-14
cos, 4-3, 6-20
det, 3-8
diag, 2-4, 3-8, 7-18
eig, 5-5, 5-15, 5-21, 5-28, 7-5, 7-15
exp, 6-20
eye, 2-4, 2-9, 5-4, 5-5, 5-28
for...end, 5-25
format, 1-6
    compact, 1-6, 5-32
    long, 1-42, 2-14, 5-32
    rat (rational), 1-27, 2-14, 5-5, 8-7
    short, 1-27, 1-42, 2-14, 5-5
gauss, 1-19, 1-24, 2-9, 2-27, 3-8, 5-9
grid, 4-3, 5-9, 5-25
gs, 6-12
hilb, 2-14
help, 2-4, 4-4, 5-9, 6-12
hold, 4-4, 5-9, 5-25, 8-16
imag, 5-21, 5-28
inv, 2-14, 2-28, 5-15, 6-16
linspace, 4-3, 5-9, 6-26
log, 6-20
lu, 2-27
max, 5-32
mean, 7-18
norm, 6-2, 6-9
nullbasis, 5-4, 5-5, 5-15, 5-28, 7-5
null, 7-15
ones, 2-4, 7-18, 8-4, 8-7
plot, 4-3, 5-9, 5-25, 6-26
plottools, 4-4
poly, 5-9
polyval, 5-9
prod, 3-8
proj, 6-12, 7-5
qr, 6-12
rand, 2-4, 2-14
randomint, 2-4, 4-23, 5-9
rank, 2-41, 4-23
real, 5-21, 5-28
ref, 2-37, 2-38, 2-41, 4-10, 4-23, 4-27, 5-4,
6-16, 6-19, 8-4, 8-7
replace, 1-6, 1-14
roots, 4-31
rref, 2-37, 2-38, 4-10
scale, 1-6, 1-14
size, 2-4, 7-15, 7-18, 8-4, 8-5
sqrt, 7-15, 7-18
sum, 8-4, 8-7
svd, 7-16, 7-18
swap, 1-6, 1-14
zeros, 1-24, 2-4, 7-15
GETTING STARTED WITH MAPLE

Using Maple with *Linear Algebra and Its Applications*

The Maple commands for most linear algebra operations, including the manipulation of vectors and matrices, can be found in the *LinearAlgebra* package that comes with Maple. The *laylinalg* package contains Maple implementations of the special MATLAB commands presented in the text, data for many problems in the text, and data for the Maple Projects. The current version of the *laylinalg* package has been prepared in Maple 18, and is backward compatible to at least Maple 10. The *laylinalg* package can be downloaded from [http://www.pearsonhighered.com/lay](http://www.pearsonhighered.com/lay).

First, download the *laylinalg.zip* file from the book’s website and extract the contents to your local computer. If you have write permission to the local installation of Maple, the easiest solution is to copy the two files to the *lib* folder within Maple’s file structure. (Often this will be C:\Program Files/Maple 18/lib.) If you cannot write to Maple’s *lib* folder, create a new folder, say C:\Documents and Settings/MyMapleLibs, copy the two *laylinalg* files to the new folder, and add this new folder to *libname*, with

> libname := libname, "C:\Documents and Settings/MyMapleLibs";

To ensure these steps are performed every time you open a Maple worksheet, this command should be placed in a Maple initialization file. The *readme.html* is designed to be viewed in your browser; it contains a more detailed version of the information in this paragraph, including information about Maple initialization files. For even more information, see the Maple help page produced by the command

> ?mapleinit

If you experience any difficulties with this installation process, please check with your instructor for specific instructions for your site.

To load a Maple “package” into a Maple session, use the *with* command. The command

> with( laylinalg );

loads the *laylinalg* package and displays the name of each of the commands in the package. This particular command also loads Maple’s own *LinearAlgebra* package.

Maple provides two options for preparing and presenting your work: worksheets and documents. Documents eliminate the traditional Maple input prompt, allow for more natural use of 2-D input, and command-free manipulations. While these features allow for more professional-looking presentations, the fact that the actual steps to enter an object or to obtain a result are often invisible greatly reduces their appeal for instructional purposes. For these reasons, all displays in these notes and all supplemental materials have been prepared as worksheets.

Copyright © 2016 Pearson Education, Inc.
For additional information about the features of worksheets and documents, please see the information in Maple’s help system on the topic "documentsvsworksheets". The direct URL to this information is [http://www.maplesoft.com/support/help/Maple/view.aspx?path=worksheet/help/documentsvsworksheets](http://www.maplesoft.com/support/help/Maple/view.aspx?path=worksheet/help/documentsvsworksheets).

Each Maple command must end with a semicolon or colon. The semicolon tells Maple to display the result of the command; a colon suppresses the output. Maple is case-sensitive but ignores all whitespace (blanks) around numbers, names, keywords, and operators in commands. For example, Maple will not recognize `with( LayLinAlg );` but will understand `with(laylinalg);`. The commands in these notes include extra spaces to increase readability.

### Entering Matrices in Maple

Maple’s `Matrix` command needs three pieces of information to create a matrix: the number of rows, the number of columns, and the value of each entry in the matrix. When `Matrix` is used with only two arguments, the elements of the matrix are assumed to be zero (see matrix `A`, below). The optional third argument specifies the entries of the matrix. If the third argument is a list or vector, these quantities are used to fill the matrix row-by-row from left-to-right (see matrices `B` and `C`). The third argument can also be a Maple function of two variables. In this case each element of the matrix is assigned a value by evaluating the function with the corresponding row and column indices (see matrix `E`). Diagonal matrices are easily defined using the `DiagonalMatrix` command from the `LinearAlgebra` package (see matrix `F`).

```maple
> A := Matrix( 2, 3 );

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

> B := Matrix( 2, 3, [1,2,3,4,5,6] );

$$B := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

> C := Matrix( 2, 3, [1,2,3,4] );

$$C := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 0 & 0 \end{bmatrix}$$

> E := Matrix( 2, 3, (i,j) -> 1/(i+j) );

$$E := \begin{bmatrix} \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{bmatrix}$$
```

Copyright © 2016 Pearson Education, Inc.
> with( LinearAlgebra ): 

> F := DiagonalMatrix( [1, 2, 3] );

\[
F := \begin{bmatrix} 
1 & 0 & 0 \\
0 & 2 & 0 \\
0 & 0 & 3 
\end{bmatrix}
\]

Similar rules apply when creating column vectors with the Vector command (see vectors \( v_1 \) and \( v_2 \)).

> v1 := Vector( 3 );

\[
v1 := \begin{bmatrix} 
0 \\
0 \\
0 
\end{bmatrix}
\]

> v2 := Vector( [ a, b ] );

\[
v2 := \begin{bmatrix} 
a \\
b 
\end{bmatrix}
\]

A shorthand syntax for entering matrices and vectors uses angle brackets (\( < > \)) instead of the Matrix and Vector commands. In this form commas (\( , \)) separate rows and vertical bars (\( | \)) separate columns. Matrices can be built from either rows or columns (see row vectors \( r_1 \) and \( r_2 \), column vectors \( c_1 \), \( c_2 \), and matrices \( M_1 \), \( M_2 \), and \( M_3 \)).

> r1 := < 1 | 2 | 3 >;

\[
r1 := \begin{bmatrix} 
1 \\
2 \\
3 
\end{bmatrix}
\]

> r2 := < 4 | 5 | 6 >;

\[
r2 := \begin{bmatrix} 
4 \\
5 \\
6 
\end{bmatrix}
\]

> M1 := < r1, r2 >;

\[
M1 := \begin{bmatrix} 
1 & 2 & 3 \\
4 & 5 & 6 
\end{bmatrix}
\]

> c1 := < 1, 4 > ;

\[
c1 := \begin{bmatrix} 
1 \\
4 
\end{bmatrix}
\]

> c2 := < 2, 8 > ;

\[
c2 := \begin{bmatrix} 
2 \\
8 
\end{bmatrix}
\]

Copyright © 2016 Pearson Education, Inc.
The Matrix palette is a graphical but seldom-used method for defining a matrix or vector. This palette, which is normally docked along the left edge of the Maple window, is expanded by clicking anywhere in its header. Enter the number of rows and columns, or use the “Choose...” button to specify the size of the matrix you wish to create. The requested object is inserted in the worksheet when the “Insert” button at the bottom of the Matrix panel is clicked. For example, if you type

\[ H := \text{Matrix}([[ m[1,1], m[1,2], m[1,3], m[1,4] ],
                         [ m[2,1], m[2,2], m[2,3], m[2,4] ]]); \]

Each occurrence of the string \( m[i,j] \) must be replaced with an appropriate value (numeric or symbolic). The <Tab> key can be used to move to the next \( m[i,j] \) entry in the matrix. Remember to insert a semicolon or colon, as appropriate depending on whether you want the results displayed or not. When the <Enter> key is pressed, Maple executes the command (or commands) in the input region where the cursor is currently located. Note that the cursor can be anywhere within the region, not necessarily at the end of the command.

Getting Help about Maple within Maple

These notes are intended to be self-contained. If, however, you are interested in additional information, you should first consult Maple’s online help. The command

\[ \text{help( laylinalg );} \]

or, equivalently, \(?\text{laylinalg} \) accesses the help worksheet with an overview the \text{laylinalg} package. Help worksheets exist for all Maple commands, including all commands in the \text{LinearAlgebra} and \text{laylinalg} packages. For example, the command \(?\text{gauss} \) accesses the help worksheet for the \text{gauss} command in the \text{laylinalg} package.

The Help menu on the Maple user interface can also be used to search for help on a specific keyword. A Full Text Search lists all help worksheets containing keywords that you specify. A Topic Search interactively shows all help worksheets whose topic matches the search string. For example, there is a long list of help topics that begin with the string \text{la}. When the search string is expanded to \text{lay} the list of matching help topics reduces to all of the help worksheets for the \text{laylinalg} package.

The entire Maple Help system is available online. The web address is \text{http://www.maplesoft.com/support/help/}. Note that the help for the \text{laylinalg} commands is available only within a Maple session, after the package has been loaded.
STUDY GUIDE NOTES

At the beginning of each Maple session for this linear algebra course, enter the commands:

```maple
restart;
with( laylinalg );
```

Notice the use of a colon to suppress the output.

SECTION 1.1 Row Operations

The data for the exercises in this section are contained in the `laylinalg` package. The list of exercises within chapter 1 section 1 for which Maple data is available is displayed by executing:

```maple
> c1s1();
```

Data is available for exercises 11-16, 19-22, 25, 29-32 and 34

If Maple simply repeats your input, it is likely that the `laylinalg` package has not been loaded in this Maple session. (Use `with( laylinalg );` to load the package.)

The data for a specific problem, such as Exercise 11 in Section 1.1, can be made available for use within your current Maple worksheet by executing the command:

```maple
> c1s1( 11 );
```

The output displays each assignment that Maple has made. In this section the data for each exercise are stored in a matrix \( M \). To see the current contents of the matrix \( M \), execute:

```maple
> M;
```

The output displays each assignment that Maple has made. In this section the data for each exercise are stored in a matrix \( M \). To see the current contents of the matrix \( M \), execute:

```maple
> M;
```

Row operations on \( M \) can be performed using commands from either the `laylinalg` or the `LinearAlgebra` package. Table 1 summarizes the commands for the three elementary row operations.

Notes:

- The syntax and functionality of the commands in the `LinearAlgebra` and `laylinalg` packages are almost identical. The names in the `LinearAlgebra` package are used throughout Maple; those used in `laylinalg` more closely match those used elsewhere in the textbook. The `laylinalg` commands will be used throughout this appendix.

- The name of any matrix in your current Maple session can be inserted in place of \( M \) in the commands above.

- The letters \( r \) and \( s \) are positive integers corresponding to row numbers. The names \( m \) and \( c \) are scalar expressions (often numbers, but sometimes symbolic expressions) that you choose.
Table 1: Comparison of elementary row operations in the \texttt{laylinalg} and \texttt{LinearAlgebra} packages.

If you enter one of the row operation commands, say,
\begin{verbatim}
> swap( M, 1, 3 );
\end{verbatim}
then the new matrix produced from $M$ is not given a name. You can refer to this result using \texttt{\%}, the history operator or using Maple's \textit{equation labels} (see \texttt{help( label );}). If, instead, you type
\begin{verbatim}
> M1 := swap( M, 1, 3 );
\end{verbatim}
then the answer is stored in the matrix $M1$. If the next operation is
\begin{verbatim}
> M2 := replace( M1, 2, 5, 1 );
\end{verbatim}
then the result after performing this row operation on $M1$ is placed in $M2$, and so on. Note the use of the two-character assignment operator, $:=$, in \textit{assignments}; the one-character symbol $=$ is used to create an \textit{equation}.

One advantage of giving a new name to each new matrix is that you can easily go back one or more steps if you do not like what you just did to a matrix. If you type
\begin{verbatim}
> M := replace( M, 2, 5, 1 );
\end{verbatim}
then the result is placed back in $M$ and the "old" $M$ is lost. Of course, the "reverse" operation
\begin{verbatim}
> M := replace( M, 2, -5, 1 );
\end{verbatim}
will recreate the original matrix $M$.

For the problems in this section and the next, the multiple $m$ needed in the \texttt{replace} or \texttt{RowOperation} command will usually be a small integer or fraction that you can compute in your head. The following notes describe how to use Maple in situations where $m$ is not easily computed mentally.

Notes:

- The entry in row $r$ and column $c$ of a Maple matrix $M$ is denoted by $M[r,c]$. If the number stored in $M[r,c]$ is a floating-point number, that is, it is displayed with a decimal point, then the number may be accurate to only about eight (8) digits. It is generally more
accurate — and simpler — to use \( M[r,c] \) instead of re-typing the displayed values in computations.

- For instance, to use the entry \( M[s,c] \) to change \( M[r,c] \) to 0, enter the commands:

\[
\text{> } m := -M[r,c] / M[s,c]; \quad \# \text{ mult of row } s \text{ to add to row } r \\
\text{> } M1 := \text{replace}( M, r, m, s ); \quad \# \text{ adds } m \times \text{ row } s \text{ to row } r
\]

Or, you could use the single command:

\[
\text{> } M1 := \text{replace}( M, r, -M[r,c]/M[s,c], s );
\]

- The environment variable \texttt{Digits} controls the number of significant digits Maple uses in floating-point computations. The default is 10. While this should be more than sufficient for the problems in this text, you can change this by assigning a new value to \texttt{Digits}. In general, you should expect a floating-point computation to be accurate to approximately \texttt{Digits-2} digits.

\section*{SECTION 1.2 Symbolic Row Operations}
A powerful feature of Maple is its ability to work with matrices and vectors that involve unspecified parameters. For example, Exercise 25 in Section 1.1 can be solved using Maple and the \texttt{laylinalg} package in the following steps

\[
\text{> } \text{c1s1( 25 );}
\]

\[
\text{> } M; \quad \# \text{ display the augmented matrix}
\]

\[
\text{> } M1 := \text{replace}( M, 3, 2, 1 ); \quad \# \text{ add } 2 \times \text{ row } 1 \text{ to row } 3
\]

\[
\text{> } M2 := \text{replace}( M1, 3, 1, 2 ); \quad \# \text{ add row } 2 \text{ to row } 3
\]

\[
\text{> } M2[3,4] = 0; \quad \# \text{ eqn to make system consistent}
\]

Symbolic multipliers can be used in the \texttt{replace} and \texttt{scale} commands, for example: \texttt{replace( A, 3, a, 1 );}, but all row numbers must be explicit integers.

\section*{SECTION 1.3 Constructing a Matrix}
To see the exercises with data for Section 1.3, execute the command:

\[
\text{> } \text{c1s3( );}
\]

The data for Exercise 25, for example, consists of the matrix \( A \), the right-hand side vector \( b \), and the columns of the matrix, \( a_1, a_2, a_3 \). The corresponding Maple names, \( A, b, a1, a2, \) and \( a3 \), can be assigned by executing the command:

\[
\text{> } \text{c1s3( 25 );}
\]

The command

\[
\text{> } M := < a1 | a2 | a3 | b >;
\]
creates a matrix with the given vectors as its columns. The same matrix could also be created by

\[
\text{> } M := < A | b >;
\]
Exercises 11–18 and 25–29 can be solved using the \texttt{laylinalg} commands \texttt{replace}, \texttt{swap}, and (occasionally) \texttt{scale} or by using the \texttt{RowOperation} command found in the \texttt{LinearAlgebra} package. The syntax for each of these is summarized in Table 1 on page MP-6.

**SECTION 1.4** \texttt{gauss} and \texttt{bgauss}

To solve $Ax = b$, row reduce the augmented matrix
\[ M := \begin{bmatrix} A & b \end{bmatrix}; \]

The command
\[ x := \begin{bmatrix} 5 \\ 3 \\ -7 \end{bmatrix}; \]
creates a column vector, $x$, with entries 5, 3, and -7. The non-commutative multiplication used to implement matrix-vector multiplication is denoted as a period:
\[ A \cdot x; \]

The \texttt{laylinalg} package contains two commands, \texttt{gauss} and \texttt{bgauss}, that can be used to speed up row reduction of a matrix. Remember to load the \texttt{laylinalg} package, by executing `with( laylinalg );`, before you try to use anything in the \texttt{laylinalg} package.

To make row reduction of $M = \begin{bmatrix} A & b \end{bmatrix}$ more efficient, the command \texttt{gauss( M, r );} chooses the leading entry in row $r$ of $M$ as a pivot and uses row replacements to create zeros in the pivot column below this pivot entry. The result can be assigned to a name, otherwise the only ways to access the result is with the history operator (%) or using Maple's equation labels (see \texttt{help( label );}). In either case, the result is displayed to the screen.

For the backward phase of row reduction, \texttt{bgauss( M, r );} selects the leading entry in row $r$ of $M$ as the pivot, and creates zeros in the column above the pivot. Use \texttt{scale} to create leading 1's in the pivot positions.

**Note:**
- Exercises 15 and 16 of this section can be solved using symbolic row operations as discussed in the Maple Note for Section 1.2.

**SECTION 1.5** Zero Matrices

The command \texttt{Vector( m, 0 );} creates the zero vector in $\mathbb{R}^m$ and \texttt{Matrix( m, n, 0 );} creates the $m \times n$ matrix of zeros. To solve an equation $Ax = 0$, use the command
\[ M := \begin{bmatrix} A & \text{Vector}( m, 0 ) \end{bmatrix}; \]
to augment $A$ with the column vector containing $m$ zeros, then use \texttt{gauss}, \texttt{swap}, \texttt{bgauss}, and \texttt{scale} to complete the row-reduction of $M$.

**SECTION 1.6** Rational and Floating-Point Format

Chemical equation-balance problems are best studied using exact or symbolic arithmetic. This is because the balance variables must be whole numbers (with no round-off allowed). When
the chemical equations have integer coefficients — as they do here — Maple will perform exact computations. That is, when \texttt{linearalgebra} row operations are applied to an integer- or rational-valued matrix, the resulting matrix will have integer or rational entries. No floating-point approximations will be used. Once you find a rational solution of a chemical equation-balance problem, you can multiply the entries in the solution vector by a suitable integer to produce a solution that involves only whole numbers.

Notes:

- To convert the entries of a matrix, \( M \), to floating-point numbers, use either \texttt{evalf( M )};
or \texttt{convert( M, float ));}.
- The command \texttt{convert( M, rational )}; returns the matrix in which each entry of the matrix is replaced with an approximately equal rational number.

\textbf{SECTION 1.10 Generating a Sequence}

The data for Exercises 9–13 in Section 1.10 consists of the migration matrix, \( M \), and the initial population vector \( x_0 \). The following set of commands generates the first ten (10) terms in the sequence defined by \( x_{k+1} = Mx_k \):

\begin{verbatim}
> x||0 := x0;  # display initial population
> for k from 0 to 9 do
>    x||(k+1) := M . x||k;  # compute \( x_{k+1} \) from \( x_k \)
> end do;
\end{verbatim}

Note that the first command reassigns \( x0 \) to itself. This command can be omitted if you do not wish to include the initial population in the output.

Numbers are entered in Maple without commas. Maple uses scientific notation for numbers larger than 1,000,000 (\( 10^6 \)) or smaller than \( 10^{-6} \). In Maple, numbers in scientific notation can be entered as \( 4.0 \times 10^5 \) or \( 2.46 \times 10^{-3} \). Note that the parentheses around a negative exponent are required. When the mantissa is entered as a floating point number, the number is displayed (and computed) with \texttt{Digits} digits. The default setting of \texttt{Digits}, 10, should be more than sufficient for all computations in this text.

If you wish to demonstrate the numerical stability of an algorithm, including the effects of floating-point arithmetic, you might find it useful to reduce the value of \texttt{Digits}.

\textbf{SECTION 2.1 Matrix Notation and Operations}

Several methods for defining a matrix in Maple have already been described in this appendix. Refer to these discussions for information about creating your own matrices.

In Maple, the \( (i, j) \)-entry of matrix \( A \) is \( A[i,j] \). There is a similar syntax to access entire rows or columns of a matrix. For example, \( A[\ldots,3] \) is all rows of column 3 of \( A \) and \( A[2\ldots] \) is all columns of row 2 of \( A \). To extract the first two columns of \( A \), use \( A[\ldots,1..2] \). (The expression \( a..b \) represents the integers from \( a \) to \( b \).)
The command \( A[\ a..b,\ c..d\ ]; \) returns a matrix whose entries come from rows \( a \) through \( b \) and columns \( c \) through \( d \) of \( A \). The number of rows in \( A \) is given by \( \text{RowDimension}(A) \) and the number of columns by \( \text{ColumnDimension}(A) \). The command \( \text{Dimension}(A) \) returns the number of columns and the number of rows in the matrix \( A \), separated by a comma. For a vector \( v \), \( \text{Dimension}(v) \) returns the number of rows in the vector.

Maple uses + and − to denote matrix addition and subtraction, respectively. Multiplication of a matrix or vector by a scalar can be obtained with the commutative multiplication operator \(*\); the non-commutative operator . must be used for matrix multiplication, including matrix–vector products. If \( A \) is square and \( k \) is an integer, \( A^k \) denotes the \( k \)th power of \( A \).

The transpose of \( A \) is \( A^{\%T} \). The outer product of \( u \) and \( v \) is computed in response to the command \( u . v^{\%T}; \). The command \( \text{DotProduct}(u,v); \) gives the inner product of \( u \) and \( v \). (If the entries of \( u \) and/or \( v \) are complex, then you probably want to use \( \text{BilinearForm} \) instead of \( \text{DotProduct} \); see also the Maple Note for Section 6.1 on page MP-18.)

Maple commands for the construction of many special matrices are given below:

\[
> M := \text{Matrix}(5, 6, 0); \quad \text{# a 5 \times 6 matrix of zeros}
\]

\[
> M := \text{Matrix}(3, 5, 1); \quad \text{# a 3 \times 5 matrix of ones}
\]

\[
> M := \text{DiagonalMatrix}( [3, 5, 7, 2, 4]); \quad \text{# a 5 \times 5 diagonal matrix}
\]

\[
> M := \text{IdentityMatrix}(6); \quad \text{# the 6 \times 6 identity matrix}
\]

\[
> M := \text{randomint}(6, 4); \quad \text{# a 6 \times 4 matrix (-9 \leq \text{entries} \leq 9)}
\]

**SECTION 2.2**  The Identity Matrix and \( A^{-1} \)

When \( A \) is \( 5 \times 5 \), the command \( M := < A | \text{IdentityMatrix}(5)>; \) creates the augmented matrix \([ A \quad I]\). Use \text{gauss}, \text{swap}, \text{bgauss}, \text{and scale} to reduce the augmented matrix to \([ I \quad A^{-1}]\), if possible. See the Maple Note for Section 1.4 on page MP-8.

There are other Maple commands that row reduce matrices, invert matrices, and solve systems \( Ax = b \). They will be introduced later, after you have studied the concepts and algorithms of this section.

**SECTION 2.3**  \textbf{MatrixInverse}, \textbf{ConditionNumber}, and \textbf{HilbertMatrix}

Determining whether a matrix is invertible is not always a simple matter. A fast and fairly reliable method is to use the command \( \text{MatrixInverse}(A); \), which computes the inverse of \( A \). An error message is displayed if the matrix is singular.

\[
> \text{MatrixInverse}(\text{Matrix}(3, 3, 1));
\]

\[
\text{Error, (in LinearAlgebra:-MatrixInverse) singular matrix}
\]

In Exercises 41–44 of this section, \( \text{ConditionNumber}(A); \) computes the condition number of any matrix \( A \) using quantities called the singular values of \( A \) (discussed in Section 7.4). To perform the experiment described in Exercise 42 you can use the following Maple commands:
> c2s3( 42 );  
> x := RandomVector( 4 );  
> b := A . x;  
> x1 := MatrixInverse(A) . b;  
> x - x1 ;  

Because Maple uses exact arithmetic for integers, x and x1 should be the same. To force Maple to use floating-point computations, replace the definition of b with b := evalf( A . x );.

For Exercise 45, the command
> HilbertMatrix( n );
produces the n \times n Hilbert matrix. The n \times n Hilbert matrix with floating-point entries is obtained with
> evalf( HilbertMatrix( n ) );

SECTION 2.4  Partitioned Matrices

Partitioned matrices can be created in Maple with the < > matrix constructor. For example, if A, B, C, E, F, and G are matrices of appropriate sizes, then the command
> M := < < A | B | C >, < E | F | G > >;
creates a larger matrix of the form

\[
M = \begin{bmatrix} A & B & C \\ E & F & G \end{bmatrix}.
\]

Once M is formed, there is no record of the partition that was used to create M. For instance, although B is the (1,2)-block used to create M, the value of M[1,2] is the same as the (1,2)-entry of A (assuming A has at least two columns). You should be aware, however, that if A[1,2] is an unevaluated name, say a, at the time matrix M is created and a is subsequently assigned a value, this affects all instances of a in the current Maple session. In this case, assigning a value to a changes both A[1,2] and M[1,2].

SECTION 2.5  LU Factorization and the \texttt{linsolve} Command

Row reduction of A using the \texttt{gauss} command will produce the intermediate matrices needed for an LU factorization of A. You can try this on the matrix in Example 2, stored as Exercise 33 of this section in the \texttt{laylinalg} package. The matrices in (5) for Example 2 (page 128 of the text) are produced by the commands:

> c2s5( 33 );  
> U := gauss(A,1);  
> U := gauss(U,2);  
> U := gauss(U,3);  

You can copy the information from the screen onto your paper, and divide by the pivot entries to produce L as in the text.
To construct a permuted LU factorization, use the Maple command

```maple
> U := gauss( U, r, v );
```

where \( r \) is the row index of the pivot and \( v \) is a row vector that lists the rows to be changed by replacement operations. For example, if \( A \) has 5 rows and the first pivot is in row 4, use

```maple
> U := gauss( A, 4, [1,2,3,5] );
```

If the next pivot is in row 2, use

```maple
> U := gauss( U, 2, [1,3,5] );
```

To build the permuted matrix \( L \), use full columns from \( A \) or the partially reduced \( U \), divided by the corresponding pivot. Then change entries to zero if they are in a row already selected as a “pivot row”.

The Maple command \texttt{LUDecomposition(A, output=\{'P','L','U'\})}; returns the three matrices \( P, L, \) and \( U \) for the permuted LU factorization of the matrix \( A \), that is, \( A = PLU \).

When \( A \) is invertible, the simplest way to solve \( Ax = b \) with Maple is to use the \texttt{LinearSolve} command: \( x := \text{LinearSolve}(A,b) \);. The command \( x := \text{MatrixInverse}(A) \cdot b; \) is less efficient and can be less accurate, particularly when the entries of \( A \) are floating-point numbers.

### SECTION 2.8 ref

The Maple command \texttt{ref} in the \texttt{laylinalg} package produces the reduced echelon form of the matrix \( A \). From this matrix you can write a basis for \( \text{Col} A \) or write the homogeneous equations that describe \( \text{Nul} A \). (Do not forget that \( A \) is a coefficient matrix, not an augmented matrix.)

Note that \texttt{ref} is exactly the same as the built-in command \texttt{ReducedRowEchelonForm} in the \texttt{LinearAlgebra} package.

### SECTION 2.9 Rank

You can use \texttt{ref} to check the rank of a matrix \( A \). You should be alert to the fact that, for floating-point matrices, roundoff error or an extremely small pivot can produce an incorrect echelon form. A more reliable command for computing the rank of a matrix \( A \) is \texttt{Rank(A)};

### SECTION 3.2 Computing Determinants

To compute \( \det A \), define \( U := A; \) and then repeatedly use the commands \( U := \text{gauss}(U,r); \) and \( U := \text{swap}(U,r,s); \) for appropriate positive integers \( r \) and \( s \) to reduce \( A \) to an echelon form \( U \) (see the Maple Note for Section 1.4 on page \text{MP-8}). Then, the determinant of an \( n \times n \) matrix \( A \) can be found using

```maple
> detA := (-1)^k * mul( U[i,i], i=1..n );
```

where \( U \) is in echelon form and \( k \) is the number of row swaps made while reducing \( A \) to \( U \).

The command \texttt{Determinant(A)}; can be used to check your work, but the longer sequence of commands is preferred — at this time — because it emphasizes the process of computing a determinant.
SECTION 4.1  Graphing Functions

The following commands plot \( y = f(t) \) for the function \( f \) in Exercise 37 (on page 200 of the text):

\[
> f := 1 - 8\cos(t)^2 + 8\cos(t)^4;
> \text{plot}( f, t=0..2\pi );
\]

In Maple’s \texttt{plot} command, the first argument is the function (or list of functions) to be plotted and the second argument is an equation with left-hand side is the name of the independent variable (here, \( t \)) and the right-hand side is the interval on which the plot is to be created. Note that \( \pi \) is Maple’s name for the constant \( \pi \). Consult Maple’s online help for the \texttt{plot} command and \texttt{plots} package for optional arguments and additional examples.

SECTION 4.3  \texttt{ref} and \texttt{GenerateMatrix}

The Maple command \texttt{ref}(A) in the \texttt{laylinalg} package produces the reduced echelon form of the matrix \( A \). From this matrix you can write a basis for \( \text{Col} \ A \) or write the homogeneous equations that describe \( \text{Nul} \ A \). (Do not forget that \( A \) is a coefficient matrix, not an augmented matrix.)

For Exercises such as 37 and 38 (page 217 of the text) the \texttt{GenerateMatrix} command can be used to generate the coefficient matrix \( A \) from a list or set of linear equations involving a set of unknowns, say \( c[1], c[2], \ldots \). For example, the coefficient matrix \( A \) and right-hand side \( b \) for Exercise 38 can be obtained by the following commands:

\[
> \text{EQN} := \text{add}( c[j] \cdot \cos(t)^j, j=0..6 ) = 0; \quad # \text{genl form of eqn}
> \text{eq}[0] := \text{eval}( \text{EQN}, t=0 ); \quad # \text{eqn w/ } t = 0
\]

\[
> \vdots
> \text{eq}[6] := \text{eval}( \text{EQN}, t=\pi ); \quad # \text{eqn w/ } t = \pi
> A := \text{GenerateMatrix}( [ \text{eq}[k] \mid k=0..6 ], \quad # \text{coef matrix}
> [ c[k] \mid k=0..6 ] );
\]

SECTION 4.4  \texttt{LinearSolve}

The Maple command \texttt{LinearSolve}( A, b ); produces the solution of \( A x = b \) when \( A \) is invertible. The \texttt{LinearSolve} command has additional capabilities that will be introduced later, after you have the appropriate background.

SECTION 4.6  \texttt{ref}, \texttt{Rank}, and \texttt{randomint}

In this course, you can use either \texttt{ref}( A ); or \texttt{Rank}( A ); to check the rank of a matrix \( A \). In practical work, particularly with floating-point matrices, \texttt{Rank} is generally more reliable than \texttt{ref}.

The \texttt{laylinalg} command \texttt{randomint}( m, n ); creates an \( m \times n \) matrix whose entries are random integers between -9 and 9.
Note:
• In earlier editions of the textbook the \texttt{randomint} command was called \texttt{randint}. The name change was necessary because MATLAB now has a \texttt{randint} command in its Communications Toolbox.

SECTION 4.7  Change-of-Coordinates Matrix

The \texttt{laylinalg} package includes data for Exercises 7–10 and 17–19 (on pages 245 and 246 of the text). To reduce the matrix \[
\left[ \begin{array}{cc|c} c_1 & c_2 & b_1 \\ c_1 & c_2 & b_2 \end{array} \right]
\] to reduced row echelon form use \texttt{ref( < c1 | c2 | b1 | b2 > );}. See Section 1.3 (page MP-7) for a description of the use of angle brackets ( \texttt{< >} ) to build matrices from a collection of column vectors.

SECTION 4.8  solve and polyroots

In Exercises 7–16 (page 253) and 25–28 (page 254), the coefficients of the polynomial in the auxiliary equation are stored in a vector \texttt{p}. The coefficients are stored in descending order. For example, the polynomial $x^2 + 6x + 9$ would be represented as \texttt{p := < 1, 6, 9 >;}.

The \texttt{polyroots} command in the \texttt{laylinalg} package computes floating-point approximations to all (real and complex) roots of the polynomial whose coefficients are stored in the vector \texttt{p}. The syntax is \texttt{polyroots( p );}.

Given the coefficient vector \texttt{p} the corresponding polynomial can be created with the following commands:
\begin{verbatim}
> n := Dimension(p);  # degree of polynomial
> P := add( p[k]*x^(n-k), k=1..n );  # polynomial w/ coefs p
\end{verbatim}

The exact roots of the polynomial can now be found using
\begin{verbatim}
> solve( P=0, x );
\end{verbatim}
Note, however, that it might not be possible to find all roots for a polynomial of degree 5 or higher.

SECTION 4.9  randomstoc

The Maple box for Section 1.10 contains information that is useful for the Exercises in this section. A random $n \times n$ stochastic matrix (with entries that are floating-point numbers between $-1$ and 1) can be obtained through the command
\begin{verbatim}
> randomstoc( n );
\end{verbatim}

SECTION 5.1  Finding Eigenvectors

The \texttt{laylinalg} package contains a command that will simplify your homework by automatically producing a basis for an eigenspace when you know the eigenvalue. For example, if $A$ is a $3 \times 3$ matrix with eigenvalue $\lambda = 7$, then the command
\begin{verbatim}
> randomstoc( n );
\end{verbatim}
The Maple command `CharacteristicPolynomial` can be used to check your answers in Exercises 9–14 (page 282 of the text). Note that if $A$ is an $n \times n$ matrix, $\text{CharacteristicPolynomial}(A, \lambda)$ is $\det(\lambda I - A)$. Observe that when $A$ is an $n \times n$ matrix, $\det(\lambda I - A) = (-1)^n \det(A - \lambda I)$. This means Maple’s definition of the characteristic polynomial agrees with the book’s definition only when $n$ is even; otherwise, they are negatives of one another.

For Exercises 28 and 29 on page 283 of the text, create a $4 \times 4$ matrix with random integer entries with the command `randomint(4)`. For Exercise 29, use `gauss` and perhaps `swap` to create the echelon form without row scaling. See also the Maple Note for Section 1.4 on page MP-8.

The following commands create the graph of the characteristic polynomial of the matrix $A$ in Exercise 30 (on page 283 of the text) for all values of the parameter $a$ in a list `a_vals`:

```maple
> c5s2(30);  # load data for Exercise 30 in Section 5.2
> p := CharacteristicPolynomial( A, lambda );  # char poly in terms of parameter \(a\)
> P := [seq( p, a=a_vals )];  # char poly for each value of \(a\)
> plot( P, lambda=0..3 );  # graph all char polys on same axes
```

**SECTION 5.3 Diagonalization and Eigenvalues**

To practice the diagonalization procedure in this section, you should use the `nulbasis` command to produce eigenvectors. For Exercises 33–36 (page 289 of the text), the command

```maple
> ev := Eigenvalues( A );
```

can be used to provide the eigenvalues. See also the Maple Note for Section 5.1 on page MP-14.

In later work, you may automatize the diagonalization process. The command

```maple
> DD,P := Eigenvectors(A);
```

returns the eigenvalues of $A$ (with repetitions) in the vector `DD` and the corresponding eigen-
vectors of $A$ in the columns of the matrix $P$. If $P$ is invertible, then $A$ is diagonalizable. Note that the matrix $P$ may not be the same as the one you find by hand. A quick check that your answer is correct is to confirm that $A \cdot P - P \cdot \text{DiagonalMatrix}(\text{DD})$ returns the zero matrix. When the matrices involve floating-point entries, one or more entries might be very small (but not exactly zero). In any case, a visual inspection suffices to check if the difference $AP - PD$ is the zero matrix.

SECTION 5.3 Complex Eigenvalues, $\text{Re}$, and $\text{Im}$

The command $\text{DD}, P := \text{Eigenvectors}(A)$; (mentioned in the previous Maple Note) also can be used for matrices with complex eigenvalues. In this case the vector $P$ and the vector $\text{DD}$ have some complex entries. (In Maple, the imaginary unit, $i$, is represented by $I$. As a result, the complex number $a + ib$ is entered as $a + I \ast b$. The real and imaginary parts of a complex number Maple quantity $z$ are $\text{Re}(z)$ and $\text{Im}(z)$, respectively.

If $A$ is a $2 \times 2$ real matrix with a pair of complex conjugate eigenvalues, let $\lambda$ be the eigenvalue with negative imaginary part. You can use the $\text{nulbasis}$ command to find a complex eigenvector $v$ corresponding to $\lambda$. Then, build $2 \times 2$ real matrices $P$, from the real and imaginary parts of $v$, and $C$, from the real and imaginary parts of $\lambda$, as follows:

\[
V := \text{nulbasis}(A - \lambda[1]\ast\text{IdentityMatrix}(2));
\]
\[
P := \langle \text{Re}(V) \mid \text{Im}(V) \rangle;
\]
\[
C := \langle \langle \text{Re}(\lambda) \mid \text{Im}(\lambda) \rangle, \langle -\text{Im}(\lambda) \mid \text{Re}(\lambda) \rangle \rangle;
\]

These matrices fit the framework of Theorem 9 on page 301 of the text.

In general, the real and imaginary parts of a complex-valued matrix, vector, or scalar are obtained with the $\text{Re}$ and $\text{Im}$ commands, respectively.

SECTION 5.6 Plotting Discrete Trajectories

The following commands create the sequence of vectors $x$, $Ax$, $A^2x$, …, $A^{15}x$ for a given initial vector $x$:

\[
> x := \langle 1, 0 \rangle; \quad \text{# initial vector (change as needed)}
\]
\[
> T := \langle x \rangle; \quad \text{# initialize list of points}
\]
\[
> \text{for } j \text{ from 1 to 15 do}
\]
\[
> \quad x := A \cdot x; \quad \text{# change 15 to number of iterations}
\]
\[
> \quad T := \langle T \mid x \rangle; \quad \text{# add new point to list}
\]
\[
> \text{end do;}
\]

These points can be plotted with the command

\[
> \text{plot}(T^{\%T}, \text{style=POINT});
\]

Omitting the second argument of the $\text{plot}$ command will “connect the dots” between the points. Other optional arguments can be used to further customize the plot. The $\text{display}$ command — in the $\text{plots}$ package — can be used to combine multiple trajectories into a single plot. (Do not forget to execute the command $\text{with(plots)}$; to load the $\text{plots}$ package.)
For Exercise 17 (on page 312 of the text), the following commands create a graph of the first component in each data point in the list $T$ constructed above.

```maple
> T1 := < < ($0..ColumnDimension(T)) >^%T, T[2,..] >;
> plot( T1^%T);
```

To graph the sum of the first two components of each data point in $T$ replace $T1$ with $T2$ defined by

```maple
> T2 := T[1,..] + T[2,..];
```

Similarly, to plot the quotient of the first two components in each data point in $T$ replace $T1$ with $T3$ defined by

```maple
> T3 := T[2,..] /~ T[1,..];
```

Note that $/$ is the element-wise division operator.

**SECTION 5.7  Plotting Trajectories for Differential Equations**

The following template of Maple commands can be used to create a plot of the direction field and a solution curve for the $2 \times 2$ system of differential equations in Example 1 from Section 5.7 (see pages 314 – 315):

```maple
> with( DEtools );
> A := < < -1.5, 1 > | < 0.5, -1 > >;
> SYS := [ diff(y1(t),t) = A[1,1]*y1(t) + A[1,2]*y2(t),
> diff(y2(t),t) = A[2,1]*y1(t) + A[2,2]*y2(t) ];
> IC := < y1(0)=5, y2(0)=4 >;
> RANGE := t = 0..10;
> WINDOW := y1 = -5..5, y2 = -4..4;
> DEplot( SYS, [y1(t),y2(t)],
> RANGE, [IC], WINDOW );
```

Note:

- To create plots for other systems, simply modify the coefficient matrix and the initial condition then modify the time interval and window to create the desired plot.

- The `with( DEtools );` command needs to be entered only once — anytime before the first use of `DEplot`.

**SECTION 5.8  Power Method and Inverse Power Method**

A Maple implementation of the Power Method (page 319 of the text) assumes that $A$ is an $n \times n$ matrix with a strictly dominant eigenvalue and an initial vector $x_0$:  

Copyright © 2016 Pearson Education, Inc.
> x := x0 ;  # initial vector
> for k from 1 to 15 do  # change 15 to number of iterations
>   mu := Norm( x );  # largest element in x in absolute value
>   x := A . x/mu;  # next iterate in power method
> end do;

The value of $\mu$ should approach the dominant eigenvalue and the vector $x$ should approach a corresponding eigenvector.

One possible implementation of the Inverse Power Method (described on page 325 of the text) is:

> I3 := IdentityMatrix( 3 );  # 3 x 3 identity matrix
> alpha := 3.3;  # initial guess at e-value
> x := < 1, 1, 1 >;  # initial guess at e-vector
> for k from 1 to 15 do  # change bounds as needed
>   y := LinearSolve( A-alpha*I3, x );  # solve $(A-\alpha I)y_k = x_k$
>   mu := Norm( y );  # largest element in abs value
>   nu := alpha + 1/mu;  # updated approx to e-value
>   x := y/mu;  # updated approx to e-vector
> end do;

Notes:

• The initial value of $\alpha$ should be reasonably close to the desired eigenvalue and $x$ is a vector whose largest element (in absolute value) is 1.

• Recall that $\text{Norm}(x)$ returns the maximum of the absolute values of the entries in $x$.

SECTION 6.1 DotProduct and Norm

The Maple command $\text{DotProduct}(u, v)$; computes the inner product of two real-valued column vectors $u$ and $v$. The associated Euclidean length of a vector $v$ is $\text{Norm}(v, 2)$. (See also the Maple Note for Section 2.1 on page MP-9.)

SECTION 6.2 Orthogonality

In Exercises 1–10 and 17–22 (pages 346 amd 347 of the text), a fast way to test the orthogonality of a set such as \{u_1, u_2, u_3\} is to create a matrix

> U := < u1 | u2 | u3 >;

whose columns are the vectors from the set, and test whether

> U^%T . U;

yields a diagonal matrix.

Notes:

• To understand how this works, see Theorem 6 on page 345 of the text.
• The \texttt{IsOrthogonal} command could also be used, but this does not emphasize the essential properties of orthogonality.

\section*{SECTION 6.3 \hspace{1cm} Orthogonal Projections}

The orthogonal projection of a vector onto a single vector was described in the previous Maple Note (for Section 6.2). The orthogonal projection onto the set spanned by an orthogonal set of vectors is the sum of the one-dimensional projections. Another way to construct this projection is to normalize the orthogonal vectors, place these orthonormal vectors in the columns of a matrix $U$, and apply Theorem 10 (see page 353 of the text). For instance, if $\{y_1, y_2, y_3\}$ is an orthogonal set of nonzero vectors, then the matrix $U$ defined by the Maple command:

\begin{verbatim}
> U := < y1/Norm(y1,2) | y2/Norm(y2,2) | y3/Norm(y3,2) >=;
\end{verbatim}

has orthonormal columns. The orthogonal projection of $y$ onto $\text{Span}\{y_1, y_2, y_3\}$ is obtained with the command:

\begin{verbatim}
> U . U^%T . y;
\end{verbatim}

\section*{SECTION 6.4 \hspace{1cm} The Gram-Schmidt Process, proj, and gs}

If $A$ has only two linearly independent columns, then the Gram-Schmidt process is:

\begin{verbatim}
> v1 := A[..,1];
> v2 := A[..,2] - DotProduct(A[..,2],v1)/DotProduct(v1,v1) * v1;
\end{verbatim}

If $A$ has three columns, add the command:

\begin{verbatim}
> v3 := A[..,3] - DotProduct(A[..,3],v1)/DotProduct(v1,v1) * v1 - DotProduct(A[..,3],v2)/DotProduct(v2,v2) * v2;
\end{verbatim}

The generalization to larger matrices should now be apparent.

Notice that Maple commands in the form

\begin{verbatim}
> DotProduct(y,u)/DotProduct(u,u) * u;
\end{verbatim}

provide the orthogonal projection of a vector $y$ onto a vector $u$. A simpler, but less illustrative, alternative is to use the \texttt{proj} command from the \texttt{laylinalg} package. The syntax is:

\begin{verbatim}
> proj( y, <u> );
\end{verbatim}

The angle brackets make the column vector $u$ into a matrix (with one column).

After you are familiar with the Gram-Schmidt process, use the command \texttt{proj( x, V );} to compute the projection of a vector $x$ onto the subspace spanned by the columns of a matrix $V$. For example:

\begin{verbatim}
> v2 := A[..,2] - proj( A[..,2], <v1> );
> v3 := A[..,3] - proj( A[..,3], <v1|v2> );
\end{verbatim}

Copyright © 2016 Pearson Education, Inc.
You should construct the $QR$ decomposition of a matrix $A$ using the approach in the text. The $gs$ command from the laylinalg package provides direct access to a complete implementation of the Gram-Schmidt process. The syntax is simply $gs(A)$; After you are very familiar with the Gram-Schmidt process using $gs$ saves time.

**SECTION 6.5  LinearSolve and LeastSquares**

When $A$ has linearly dependent columns, you can write the general description of all least-squares solutions on paper after you row reduce the augmented matrix for the normal equations:

```maple
> ref( `< A^%T.A | A^%T.b >`);
```

When $A$ has linearly independent columns, the solution to the normal equations is found with:

```maple
> LinearSolve( A^%T.A, A^%T.b );
```

Other approaches to solving the normal equations include the use of $\text{ref}$ as described above and the explicit use of $(A^T A)^{-1}$. For Exercises 15 and 16 on page 368 of the text, see the Numerical Note on page 367 of the text and use $\text{LinearSolve}( R, Q^T b );$ to solve $R \mathbf{x} = Q^T \mathbf{b}$. You should use the normal equations or the QR factorization. This will give you a solid conceptual background for applying least-squares techniques later in your career.

For Exercise 26 (page 369 of the text), the command

```maple
> A := `< A1, A2 >;
```

creates the (partitioned) matrix whose top block is $A_1$ and bottom block is $A_2$. Of course, each block must have the same number of columns. (See also the Maple Note for Section 2.4 on page MP-11 for more information on partitioned matrices.)

**SECTION 6.6  Computing Regression Coefficients**

Maple can be used to construct the design and observation vectors from a list of data points. For example, for Exercise 1 on page 376 of the text:

```maple
> c6s6( 1 );   # load data from laylinalg
> y := row( DATA, 2 );   # observation vector
> n := Dimension( y );   # number of data points
> X := `< Vector(n,1) | x >;   # design matrix
```

The least squares solution to $X \beta = y$ is found as in the previous Maple Note (for Section 6.5).

**SECTION 6.6  LeastSquares, Functions of Vectors, and map**

Once you create the design matrix $X$ and the observation vector $y$, your computation for least-squares solutions here are the same as those described in the Maple Note for Section 6.5. Here, $A$ and $b$ are replaced by $X$ and $y$, respectively. The Maple command

```maple
> ref( `< X^%T.X, X^%T.y >` );
```

leads to the general description of all least-squares solutions. When $X$ has linearly independent columns, the command

Copyright © 2016 Pearson Education, Inc.
LinearSolve( X^%T.X, X^%T.y ); yields the least-squares solution. In subsequent courses, you might encounter examples where the columns of X are not linearly independent. Then LeastSquares( X, y ); returns all least-squares solutions.

To construct the design matrix for an exercise in this section, you might find it helpful to use Maple’s map command to apply a function to each element of a vector. If x is a vector and f is a Maple function, then map( f, x ); is a vector the same size as x whose entries are the result of applying f to the entries in x. For example, in Exercise 7 on page 376 the vector x can be created as above and the vector of the square of each element is x2 := map( u->u^2, x );. The design matrix is now easily constructed with the use of the angle bracket matrix constructor. When f is a built-in Maple function, say the exponential function exp, the syntax is somewhat simpler: map( exp, x );. (You could also use exp˜( x ); to map the exp function to each element of x.)

SECTION 6.7  int and Inner Products on C[0,2π]

The Maple command to evaluate the definite integral \( \int_a^b f(t) \, dt \) is int( f, t=a..b );. To solve Exercise 28 (page 385 of the text), we need a Maple procedure that computes the inner product of two functions in C[0,2π]:

\[
\text{ip} := (f,g) \mapsto \int_{0}^{2\pi} f(t)g(t) \, dt;
\]

The easiest way to define this inner product procedure, ip, and the four functions \( p_0, p_1, p_2, \) and \( p_3 \) is to load them from laylinalg package by executing the first command displayed below. The Gram–Schmidt process described in the Maple Note for Section 6.4 on page MP-19 is applicable here. For example, the first two steps are:

\[
\begin{align*}
& c6s7( 28 ); \quad q0 := p0; \\
& q1 := p1 - \text{ip}(p1,q0)/\text{ip}(q0,q0) * q0;
\end{align*}
\]

Observe that the only difference between these commands and the ones introduced in Section 6.4 is that DotProduct is replaced with ip, the newly-defined inner product on C[0,2π].

SECTION 6.8  Graphing Functions

The plot command can be used to plot one or more functions. For example, the two plots in Figure 3 for Example 4 on page 390 of the text can be created as follows:

\[
\begin{align*}
& f := t; \quad \text{# define function } f \\
& F3 := \pi - 2*\sin(t) - \sin(2*t) \quad \text{# 3rd-order Fourier approx} \\
& \quad - 2/3*\sin(3*t); \\
& \text{plot( [f,F3], t=0..2*Pi );} \quad \text{# Figure 3 (a)} \\
& F4 := F3 - 2/4*\sin(4*t); \quad \text{# 4th-order Fourier approx} \\
& \text{plot( [f,F4], t=0..2*Pi );} \quad \text{# Figure 3 (b)}
\end{align*}
\]

The Maple Note for Section 4.1 on page MP-12 contains additional information about Maple’s plot command.

Copyright © 2016 Pearson Education, Inc.
SECTION 7.1 Orthogonal Diagonalization

Use the Eigenvalues and nulbasis commands to find the eigenvalues and corresponding eigenvectors as in Section 5.3 on page MP-15. If you encounter a two-dimensional eigenspace, with a basis \( \{v_1, v_2\} \), use

\[
> v2 := v2 - \text{DotProduct}(v2,v1)/\text{DotProduct}(v1,v1) \ast v1;
\]

or

\[
> v2 := v2 - \text{proj}(v2, <v1>);
\]

to ensure the two eigenvectors are orthogonal. See the Maple Note for Section 6.4 on page MP-19. After you normalize the vectors and create \( P \), check whether \( P^T P = I \) to verify that \( P \) is indeed an orthogonal matrix.

SECTION 7.4 The Singular Value Decomposition

The singular value decomposition of a matrix \( A \) is constructed from the orthogonal diagonalization of \( A^T A \). The orthogonal diagonalization of a matrix is discussed in the Maple Note for Section 7.1 on page MP-21. Note that the eigenvalues of \( A^T A \) may not be sorted in decreasing order. You may need to reorder some of the information from the orthogonal diagonalization when creating the diagonal matrix \( D \) and matrix of right singular vectors \( V \). For example, if \( A^T A \) is a 3 \times 3 matrix and \( \lambda \) is a list of the eigenvalues of \( A^T A \) with, say, \( \lambda[3] > \lambda[1] > \lambda[2] \), then the command

\[
> \Sigma := \text{DiagonalMatrix}(\text{seq}(\sqrt{\lambda[i]}, i=[3,1,2]));
\]

creates the diagonal matrix with the same dimensions as \( A \) with the sorted singular values of \( A \) along its main diagonal. The matrix of left singular vectors, \( U \), can be formed by normalizing the nonzero columns of the product \( A V \). If \( U \) is not square, the missing columns must be chosen to form an orthonormal basis for \( \text{Nul} A^T \). (See Figure 4 on page 423 in the text.) Since

\[
> \text{GramSchmidt}(\text{NullSpace}(A^%T), \text{normalized});
\]

produces this orthonormal basis, the square matrix \( U \) is obtained with

\[
> U := < U, \text{GramSchmidt}(\text{NullSpace}(A^%T), \text{normalized})>;
\]

This construction encourages you to think about the fundamental properties of the singular value decomposition. As you become more familiar with the steps in this process, you might find the SingularValues command more useful as it provides direct access to the singular values and vectors of a matrix. The command

\[
> S,U,Vt := \text{SingularValues}(A, \text{output}=[’S’,’U’,’Vt’]);
\]

returns a vector \( S \) of singular values, the matrix \( U \) of left singular vectors, and the transpose of the matrix \( V \) of right singular vectors. To check these calculations, let \( \Sigma \) be the diagonal matrix the same size as \( A \) with the singular values of \( A \) along its main diagonal. Then \( U \cdot \Sigma \cdot Vt \) should produce the matrix \( A \).

SECTION 7.5 Computing Principal Components

The \( p \)-dimensional vector containing the mean of each row of a \( p \times N \) matrix \( X \) can be created with the command:
The $p \times N$ matrix containing $N$ copies of the row averages is obtained with:

$$< Rmean \ N >$$

Thus, to convert the data in $X$ into mean-deviation form, use

$$B := X - < Rmean \ N >;$$

The sample covariance matrix is produced by:

$$S := B . B^\%T/ (N-1);$$

The principal component analysis is completed by performing a singular value decomposition on the matrix $A = \frac{1}{\sqrt{N-1}} B^T$. (See the Numerical Note on page 431 of the text.)

**SECTION 8.1  Affine Combinations**

Given points $v_1, v_2, v_3, v_4$, and $y$, use

$$A := < v2-v1 \ v3-v1 \ v4-v1 \ y-v1>$$

to translate the points by $v_1$. Having the row reduced form of $A$ enables us to write $y - v_1$ as a linear combination of $v_2 - v_1, v_3 - v_1, v_4 - v_1$. Then add $v_1$ to obtain the expression for $y$.

If the chosen points $b_1, b_2, \ldots, b_n$ form a basis for $\mathbb{R}^n$, then the calculations are less involved. For any $y$ in $\mathbb{R}^n$, $y$ is a unique linear combination of $b_1, b_2, \ldots, b_n$. This combination is an affine combination of the $b$'s if and only if the weights sum to one. To determine if points $p_1$ or $p_2$ are affine combinations of a basis $b_1, b_2, b_3$ begin by forming the matrix

$$A := < b1 \ b2 \ b3 \ p1 \ p2 >;$$

If $C := \text{ref}(A)$, the Maple command

$$\text{add}( C[i,4], i=1..3 \ )$$

sums the weights in the fourth column of $C$ whereas the command

$$< \text{seq}( \text{add}( C[i,j], i=1..3 \ ), j=1..5 \ ) >^\%T;$$

will sum the entries in all of the columns of the $3 \times 5$ matrix $C$.

The standard homogeneous form of a point $v$ in $\mathbb{R}^n$ is $< v, 1 >$. For calculation purposes it is easier to move the rows of 1's to the top in the first step. To use the homogeneous form of the points given by $A :=< v1 \ v2 \ v3 \ p >$, the command

$$< \text{Vector}(3,1)^\%T, A >;$$

places a row of 1's in the top row. For an arbitrary number of columns, use

$$< \text{Vector}(\text{ColumnDimension}(A),1)^\%T, A >;$$

The commands $C = \text{ref}(A)$; and $< \text{seq}( \text{add}( C[i,j], i=1..3 \ ), j=1..5 \ ) >^\%T$; will again be helpful here.

**SECTION 8.2  Affine Independence**

To determine if the points $v_1, v_2, v_3, v_4$ are affinely dependent, construct and row reduce the matrix $A := < v2-v1 \ v3-v1 \ v4-v1>$; Row reducing the matrix $A$ makes it possible to write $v_4 - v_1$ as a linear combination of $v_2 - v_1$ and $v_3 - v_1$. More algebra is needed to write $v_4$ as a linear combination of $v_1, v_2$, and $v_3$. See the Maple Note for Section 8.1.
Given an affinely independent set \{a, b, c\}, the following steps will find the barycentric coordinates of a point \( p \) determined by the set \{a, b, c\} by row reducing the augmented matrix of points in homogenous form

\[
> A := < a | b | c | p >;
> A := < Vector(ColumnDimension(A),1)^%T, A >; \quad \# \text{place row of ones in top row}
> C := ref(A); \quad \# \text{row echelon form of } A
> r, c := Dimension(C); \quad \# \text{obtain dimension of } C
> < seq( add( C[i,j], i=1..r ), j=1..c ) >^%T; \quad \# \text{sum entries in columns of } C
\]

**SECTION 8.6  bezier**

The `bezier` command in the `laylinalg` package can be used to draw graphs of Bézier curves of degrees 2, 3, and 4 in the plane. The basic syntax is `bezier( d, G )` where \( d \) is the degree and \( G \) is the geometry matrix corresponding to the \( d + 1 \) control points. Note that \( G \) must be a \( 2 \times (d + 1) \) matrix.

To create an image with more than one Bézier curve, create each curve separately and use the `display` command to combine them into the final image. For example, the following commands produce a curve similar to Figure 4 on page 486 of the text:

\[
> G := < <0,3> | <4,0> | <5,3> | <7,4> | <8,4.5> | <9,3> | <7,-1> >;
> P1 := bezier( 3, G[..,1..4] ):
> P2 := bezier( 3, G[..,4..7] ):
> plots:-display( P1, P2 );
\]
Index of Maple Commands

\(^{\sim}\) (element-wise division), MP-16
\(^{\sim}\) (element-wise function mapping), MP-20
* (multiplication, scalar), MP-9
+ (plus), MP-9
, (comma), MP-3
- (minus), MP-9
\(\rightarrow\) (arrow operator), MP-20
. (multiplication, matrix), MP-7 MP-9
.. (range), MP-9
:= (assignment), MP-6
< > (matrix constructor), MP-3 MP-10 MP-13 MP-19 MP-20
= (equality), MP-6
?, see help
\% (history operator), MP-6 MP-8
^ (power), MP-9
^%T (transpose), MP-9 MP-16 MP-18 MP-20
ad., MP-13 MP-14 MP-22 MP-23
convert
  float, MP-8
  rational, MP-8
diff, MP-16
Digits, MP-6 MP-9
evalf, MP-8
for..do..end do (repetition statement), MP-9 MP-16 MP-17
help, MP-4
Im (imaginary part), MP-15
int, MP-20
map, MP-20
Matrix, MP-2 MP-3 MP-8 MP-10
mul, MP-12
Pi, MP-12
plot, MP-12 MP-15 MP-16 MP-20 MP-21
Re (real part), MP-15
seq, MP-15 MP-21 MP-23
solve, MP-14
sqrt, MP-21
Vector, MP-2 MP-8
with, MP-1

help, MP-9
labels, MP-8

Matrix palette, MP-3
DEtools package, MP-17
DEplot, MP-16 MP-17
laylinalg package, MP-1 MP-4 MP-5
bezier, MP-23
bgauss, MP-8 MP-10
gauss, MP-3 MP-8 MP-10 MP-12 MP-13
gs, MP-19
nulbasis, MP-14 MP-15 MP-21
polyroots, MP-13
proj, MP-19 MP-21
randomint, MP-10 MP-13 MP-14
randomstoc, MP-14
ref., MP-11 MP-14 MP-19 MP-20 MP-22 MP-23
replace, MP-5 MP-6
scale, MP-5 MP-8 MP-10
swap, MP-5 MP-6 MP-8 MP-10 MP-12 MP-14

LinearAlgebra package, MP-1 MP-4 MP-5
BilinearForm, MP-9
CharacteristicPolynomial, MP-14 MP-2 MP-15
ColumnDimension, MP-9 MP-23
ConditionNumber, MP-10
Determinant, MP-12
DiagonalMatrix, MP-2
DiagonalMatrix, MP-10 MP-21
Dimension, MP-9 MP-14 MP-19 MP-22 MP-23
DotProduct, MP-9 MP-17 MP-18 MP-20 MP-21
Eigenvectors, MP-14 MP-15 MP-21
Eigenvalues, MP-14 MP-15 MP-21
GenerateMatrix, MP-15
GramSchmidt, MP-21
HilbertMatrix, MP-10
IdentityMatrix, MP-10 MP-14
IsOrthogonal, MP-18
LeastSquares, MP-20
LinearSolve, MP-11 MP-13 MP-17 MP-19 MP-20
LUDecomposition, MP-11
MatrixInverse, MP-10[MP-11]
Norm, MP-17[MP-18]
NullSpace, MP-21
RandomVector, MP-10
Rank, MP-12[MP-13]
ReducedRowEchelonForm, MP-12
RowDimension, MP-9
RowOperation, MP-5[MP-6]
SingularValues, MP-21

plots package

display, MP-16[MP-23]
Notes for the *Mathematica* Computer Algebra System

**GETTING STARTED WITH MATHEMATICA**

**Using Mathematica with Linear Algebra and Its Applications**

To use *Mathematica* for your homework in this course, you will need additional *Mathematica* files which were created specifically for your text. These are available online through Pearson Publications. Ask your instructor.

**Mathematica Notebooks**

The electronic data downloaded from the web correspond to selected sections and exercises in *Linear Algebra and Its Applications*, 5th edition. These files are interactive documents called notebooks. The class exercise notebooks have a name of the form C(chapter#)S(section#).nb or C(chapter#)Ssuppl.nb. For example, the notebook C2S3.nb contains electronic data for selected exercises in Section 3 of Chapter 2. There are also applied projects, case studies, and computer assignment notebooks, discussed in the *Mathematica* Manual.

Each notebook is divided into a sequence of individual units called *cells*. In many cases, cells are grouped together into sections and subsections. Each cell has a corresponding *cell bracket* appearing in the right margin of the notebook. The first cell contains the title of the notebook. For the notebooks with exercises at the end of each section, the second set of cells in the notebook consists of general instructions and the third set contains a summary of the commands to be used in the notebook with instructions for any new commands. The remaining cells contain selected data for the homework exercises. Notice the double cell brackets for the sections; the bracket with the little triangle at the bottom indicates that there are more cells contained in those sections. To open all grouped cells in the notebook, highlight the outermost bracket on the right, then select the *Cell* menu item at the top, select *Grouping* and pick Open All Subgroups (Close All Subgroups if you wanted to close them). You could also open (or close) the sections of cells one at a time by double-clicking on the bracket that has the small triangle at the bottom.

**Mathematica Representation of a Matrix**

Most of the data in the electronic notebooks for *Linear Algebra and Its Applications* consists of matrices.

The *Mathematica* representation of the matrix
\[
\begin{pmatrix}
a_{11} & a_{12} & \cdots & a_{1n} \\
a_{21} & a_{22} & \cdots & a_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{m1} & a_{m2} & \cdots & a_{mn}
\end{pmatrix}
\]
is
\[
\{ \{a_{11}, a_{12}, \ldots, a_{1n}\}, \{a_{21}, a_{22}, \ldots, a_{2n}\}, \ldots, \{a_{m1}, a_{m2}, \ldots, a_{mn}\} \}.
\]

Each set of objects contained in curly brackets (braces) \{ \} is called a *list* and therefore a matrix is just a list of lists.
Using Notebook Data

To illustrate how to work with the electronic data found in the notebooks, start the *Mathematica* program and then open the notebook C2S3.nb. After reviewing the Summary of Commands, open the section containing Exercise Data. Each exercise is in its own subsection. When you open the subsection for Exercise 1, you will see an *input cell* containing the matrix for Exercise 1. It contains two statements, where the Return key or Enter key on the QWERTY pad has been used at the end of the first line to enter a new line.

\[
A = \begin{bmatrix} 5 & 7 \\ -3 & -6 \end{bmatrix}
\]

\[
A//\text{MatrixForm}
\]

The first statement assigns to \( A \) the matrix listed and the second will show you \( A \) in matrix form. Click anywhere inside the cell containing the assignment statement and then execute as follows:

- If you are using a keyboard with a numerical keypad, you may simply press the *Enter key on the numerical keypad*.
- The Enter or Return key on the QWERTY keyboard does not perform in the same manner. To execute a command with this *Enter or Return key*, you must press it *together with the Ctrl key*.

When you do this, the labels In[1]= will then appear at the start of the assignment statement. Also, output cells beginning with Out[1]= and Out[2]//MatrixForm= will appear displaying the matrix used in the assignment statement.

\[
\begin{align*}
\text{In}[1] &= A = \begin{bmatrix} 5 & 7 \\ -3 & -6 \end{bmatrix} \\
A//\text{MatrixForm} \\
\text{Out}[1] &= \begin{bmatrix} 5 & 7 \\ -3 & -6 \end{bmatrix} \\
\text{Out}[2]//\text{MatrixForm} &= \begin{bmatrix} 5 & 7 \\ -3 & -6 \end{bmatrix}
\end{align*}
\]

The In[ ] and Out[ ] appear automatically after you execute a command. Notice that the command, \text{MatrixForm} has no spaces and the “M” and the “F” must be capitalized. The // is a postfix command that will be used with other *Mathematica* commands.

You will notice that the instructions from your text telling you what you are to do for each exercise are not repeated in the *Mathematica* notebooks. Only the data sets are in these Exercise cells. Typically you will only be working with a small number of exercises in a given notebook. You can work problems within a notebook in any order, skipping any homework data cells you want. If you wish to print the exercises you completed, hold down the Ctrl key and select all of the cells you want printed; then choose *Print Selection* from the *File* pull-down menu. Another way to print your results is to save it as a pdf and print it or copy and paste your work into a Word document. Using the *Copy As* option under *Edit* pull-down menu allows you to copy cell contents in many forms, including plain text and pictures.

Creating Matrices with *Mathematica*

When working with one of the notebooks, you may on occasion want to enter your own data into the notebook. For instance, to create your own matrix, you first need to form a new input cell. To do this, use your mouse and click between the two cells where you want to put your matrix and a new cell will appear as soon as you begin entering information. Suppose you want to define \( A \) to be the matrix

\[
\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix}
\]
One way to do this is to enter the following command in a new input cell:

\[ A = \{ \{ 1, 2, 3, 4 \}, \{ 5, 6, 7, 8 \}, \{ 9, 10, 11, 12 \} \} \]

You must use the curly braces here to enclose the entire matrix and to enclose each row. Also, insert commas between entries and between rows.

Another way to enter the matrix is to select Insert from the Menu at the top and choose Table/Matrix and New. Then indicate the number of rows and columns you want, in this case 3 and 4, respectively. Then select OK and a 3 x 4 matrix with blank entries will appear:

\[ A = \begin{pmatrix} \_ & \_ & \_ & \_ \\ \_ & \_ & \_ & \_ \\ \_ & \_ & \_ & \_ \end{pmatrix} \]

You may enter your numbers in the appropriate cells. However, even though you enter the matrix in this form, the default output will be

\[ \{ \{ 1, 2, 3, 4 \}, \{ 5, 6, 7, 8 \}, \{ 9, 10, 11, 12 \} \} \]

If you want to see the output of this matrix in matrix form, you can use the \texttt{/MatrixForm} as illustrated earlier or you can define your matrix within a MatrixForm command:

\[ \text{MatrixForm}[A = \{ \{ 1, 2, 3, 4 \}, \{ 5, 6, 7, 8 \}, \{ 9, 10, 11, 12 \} \}] \]

A word of caution here – if you ASSIGN A to be a matrix in MatrixForm, do NOT define A this way:

\[ A = \{ \{ 1, 2, 3, 4 \}, \{ 5, 6, 7, 8 \}, \{ 9, 10, 11, 12 \} \} /\text{MatrixForm} \]

This will cause an error in later computations with the matrix.

The entry in the \(i^{th}\) row and \(j^{th}\) column of matrix A is accessed by entering \(A[[i, j]]\). Note the use of the double square brackets. For example, the input:

\[ A[[2, 3]] \]

outputs the entry in the 2\textsuperscript{nd} row, 3\textsuperscript{rd} column: 7

\textit{Mathematica} commands are reserved words that cannot be used in assignment statements. For example, if you execute the following input statement, it results in an error message stating that \(N\) is a protected symbol. If you enter

\[ N = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \]

You will get the error message:

\texttt{Set::wrsym: Symbol N is Protected. >>}

\{\{1, 2\}, \{3, 4\}\}

One way to avoid this problem is to use a lower case letter \(n\) instead of a capital \(N\) in the assignment statement. The above Symbol error is due to the fact that all \textit{Mathematica} commands and functions begin with a capital letter and therefore a word beginning with a lower case letter will never result in a protected symbol error. Another way around this problem is to use an entire word (starting with a lower case letter) for the name of a matrix in an assignment statement. For example, we could let mat\(N\) represent the matrix in the following assignment statement. Any such word created by the user may end with a digit (0-9), but cannot begin with a digit and no spaces can occur.

\[ \text{matN} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \]
Built-in Palettes in Mathematica

On some homework exercises, you will find it useful to use one or more of the palettes that are built into Mathematica. One particularly useful one is the Basic Math Input palette which contains additional shortcuts for computing powers, square roots, entering 2 x 2 matrices and so forth. To open this palette, choose Palettes from the menu, then select Other, followed by Basic Math Input. The Basic Math Input palette will appear on the right side of the screen. If it overlaps your notebook window, you can move your notebook window or the palette around and change the size of your window if necessary so that the palette and the notebook appear separately in two nonoverlapping windows. If you were to start with a 2 x 2 matrix, you can add rows by entering CTRL+ and add columns by entering CTRL, . In fact, if you were to open a line with a left parentheses, you can create a matrix by adding rows and columns.

Local and Online Help

One of the new features in Mathematica 9, the Wolfram Predictive Interface, has been enhanced in Mathematica 10. Similar to how a search engine assists you in completing your anticipated query, this Predictive Interface anticipates your Mathematica command and helps you complete it. This is true for major commands, as well as for options within commands. It also helps suggest what you might want to do next. Mathematica 10 continues to support the Free-Form Linguistic Input. This allows users to enter plain English and get immediate results and the Mathematica input for further exploration—without the need for syntax. This feature can eliminate many of the frustrations individuals encounter in using Mathematica and it provides immediate access to the powerful Wolfram|Alpha website (http://www.wolframalpha.com/) containing a vast collection of data that can be integrated into your assignments if desired. Using Wolfram|Alpha requires internet connectivity.

Another way to seek help is to select Help from the menu. Under the Help menu, you can choose many ways to receive help, including tutorials. One category under Help that you might want to explore is the Function Navigator. If you select this, the Function Navigator window with many menu items will pop up. If you want to learn about commands that are useful in linear algebra, select Mathematics and Algorithms and then select the subitem Matrices and Linear Algebra. We will be calling upon many of the built-in commands described in this section.
STUDY GUIDE NOTES

SECTION 1.1 Row Operations

Since matrices in Mathematica are represented as lists of lists, one way that row operations can be done is to assign each row of a matrix a name. For example, consider a matrix with three rows. The use of the semicolon separates commands and also suppresses output.

\[ v_1 = \{ 1, 2, 3, 4 \}; \quad v_2 = \{ 0, 0, 8, 5 \}; \quad v_3 = \{ 2, 6, 0, 7 \}; \]
\[ A = \{ v_1, v_2, v_3 \} \]

We can redefine \( A \) to be the matrix with rows 2 and 3 reversed. To do this we could simply reassign the value of \( A \) to be this new matrix, but it is a good idea to un-assign it first using the Clear command. You will notice that every argument in Mathematica is enclosed in square brackets.

\[ \text{Clear}[A] \]
\[ \text{MatrixForm}[A = \{ v_1, v_3, v_2 \}] \]

Then the output in matrix form will be

\[
\begin{bmatrix}
1 & 2 & 3 & 4 \\
2 & 6 & 0 & 7 \\
0 & 0 & 8 & 5
\end{bmatrix}
\]

If you want to now replace the new row 2 (named \( v_3 \)) with “row2 – 2 row1”, you can do the following:

\[ \text{Clear}[A] \]
\[ \text{MatrixForm}[A = \{ v_1, v_3-2 v_1, v_2 \}] \]

Then the output in matrix form will be

\[
\begin{bmatrix}
1 & 2 & 3 & 4 \\
0 & 2 & -6 & -1 \\
0 & 0 & 8 & 5
\end{bmatrix}
\]

Suppose we now divide row 2 by 2 and divide row 3 by 8.

\[ \text{Clear}[A] \]
\[ \text{MatrixForm}[A = \{ v_1, (v_3-2 v_1)/2, v_2/8 \}] \]

Then the output in matrix form will be

\[
\begin{bmatrix}
1 & 2 & 3 & 4 \\
0 & 1 & -3 & -1/2 \\
0 & 0 & 1 & 5/8
\end{bmatrix}
\]

All of the above and more can also be accomplished by a single command, called RowReduce, which will be discussed later.

Symbolic and Elementary Row Operations

The operations performed above could also be performed on a matrix with variable or unspecified parameters such as

\[
\begin{pmatrix}
5 & p \\
q & 2
\end{pmatrix}
\quad \text{or} \quad
\begin{pmatrix}
a & b \\
c & d
\end{pmatrix}
\]
Exact and Approximate Calculations

Mathematica will perform exact calculations as long as all the numbers used in your work are integers or rational numbers not containing any decimal points. For example, 4/9 is an exact number, but 4.0/9 or 4/9. are approximate numbers which Mathematica assumes are only accurate to a fixed number of decimal places. If you perform a calculation involving both an approximate number and an exact number, then the result will be an approximate number.

At first it may seem that you would always want to perform exact calculations in order to obtain exact answers. But it takes Mathematica more computer time to perform exact calculations and when performing computations with large matrices, results may be slow in coming if you use exact numbers. In other instances, Mathematica will not be able to perform exact calculations. Therefore at times it will be necessary to enter approximate numbers to obtain approximate results.

SECTION 1.3 Transpose

The transpose of a given matrix \( A \) is a matrix whose \( i^{th} \) column is the \( i^{th} \) row of \( A \). The command \textbf{Transpose} creates such a matrix from a given matrix. Notice the use of square brackets. For example, suppose \( A = \begin{pmatrix} 1 & 3 & 5 \\ -7 & 0 & 8 \end{pmatrix} \). Then executing \textbf{Transpose}[A] results in an output of \( \begin{pmatrix} 1 & -7 \\ 3 & 0 \\ 5 & 8 \end{pmatrix} \).

Representation of a Vector

A vector \( v \) which is represented as either \((v_1, v_2, \ldots, v_n)\) or as \( \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \) in the textbook, can either be represented by a list \( \{v_1, v_2, \ldots, v_n\} \) or as the matrix \( \{\{v_1\}, \{v_2\}, \ldots, \{v_n\}\} \) in Mathematica. We also refer to \( \{v_1, v_2, \ldots, v_n\} \) as being in row form. Executing the command \textbf{Transpose}[\{ v \}] turns \( v = \{v_1, v_2, \ldots, v_n\} \) into \( \{\{v_1\}, \{v_2\}, \ldots, \{v_n\}\} \).

Constructing a Matrix

For matrix \( A \) and vector \( b \), each with the same number of rows, The \textbf{Append} command in Mathematica will add the vector \( b \) onto the matrix \( A \) as an extra row. However, we want the vector \( b \) to be appended to the columns of \( A \) and we accomplish this by using the \textbf{Transpose} command twice as follow. We will \textbf{Append} the vector \( b \) to the Transpose of \( A \). The final matrix we want will be the Transpose of that result.

\[
A = \{\{a_{11}, a_{12}, a_{13}\}, \{a_{21}, a_{22}, a_{23}\}\};
\]

\[
b = \{b_1, b_2\};
\]

\[
\text{MatrixForm[augAb=Transpose[Append[Transpose[A], b]]]}
\]

Then the output in matrix form will be

\[
\begin{pmatrix}
a_{11} & a_{12} & a_{13} & b_1 \\

a_{21} & a_{22} & a_{23} & b_2
\end{pmatrix}.
\]

The semicolons suppress the first two outputs.

In general, if we wish to extract the columns of a matrix for analyses as vectors, we apply the double
bracket command to the transpose of the matrix. So, for example in the above matrix A, the command
\[ \texttt{v1 = Transpose[A][[1]]} \]
would output the vector \( \texttt{v1} \) as \( \{a_{11}, a_{21}\} \), the first column of the matrix A. If you
ask for this in matrix form, it would be \( \begin{pmatrix} a_{11} \\ a_{21} \end{pmatrix} \).

**SECTION 1.4 Solving the Matrix Equation \( Ax = b \)**

To solve \( Ax = b \), row reduce the matrix \( M = [A \ b] \). This can be done using the elementary row operations
discussed in Section 1.1. However, *Mathematica* has a command that will do these steps correctly and in
the most efficient manner. The command is \texttt{RowReduce}. Consider Exercise 12 in Section 1.4:

\[
A = \begin{pmatrix} 1 & 2 & 1 \\ -3 & -1 & 2 \\ 0 & 5 & 3 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix}
\]

\[
\texttt{MatrixForm[augAb=Transpose[Append[Transpose[A],b]]]} \quad \texttt{MatrixForm[rrAb=RowReduce[augAb]]}
\]

Then the output in matrix form will be

\[
\begin{pmatrix}
1 & 2 & 1 & 0 \\
-3 & -1 & 2 & 1 \\
0 & 5 & 3 & -1
\end{pmatrix}
\]

We have used semicolons on all commands except those that display the augmented matrix and the row
reduced form of the augmented matrix in matrix form. Notice that you can read the solution to the
system of equations from the last matrix: \( x_1 = \frac{3}{5}, \ x_2 = -\frac{4}{5}, \ \text{and} \ x_3 = 1. \)

Suppose that the vector \( b \) had consisted of variable expressions as in Example 3 in Section 1.4:

\[
A = \begin{pmatrix} 1 & 2 & -1 \\ -3 & -4 & 2 \\ 5 & 2 & 3 \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}
\]

\[
\texttt{MatrixForm[augAb=Transpose[Append[Transpose[A],b]]]} \quad \texttt{MatrixForm[rrAb=RowReduce[augAb]]}
\]

Then the output in matrix form will be

\[
\begin{pmatrix}
1 & 2 & 1 & b_1 \\
-3 & -1 & 2 & b_2 \\
0 & 5 & 3 & b_3
\end{pmatrix}
\]

\[
\begin{pmatrix}
1 & 0 & 0 & \frac{1}{10} (13b_1 + b_2 - 5b_3) \\
0 & 1 & 0 & \frac{1}{10} (-9b_1 - 3b_2 + 5b_3) \\
0 & 0 & 1 & \frac{1}{2} (3b_1 + b_2 - b_3)
\end{pmatrix}
\]

**Multiplying a Matrix and a Vector**

A period (\( . \)) is used to multiply a matrix and a vector. Consider the following example.

\[
M = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 5 & 5 & 8 & 7 \end{pmatrix}
\]
\[ v = \{-1, 6, 0, 10\}; \]
\[
\text{MatrixForm}[M.v]
\]
Then the output in matrix form will be
\[
\begin{pmatrix} 36 \\
95 \end{pmatrix}.
\]

**SECTION 1.5 The Zero Matrix**

The command `ConstantArray[c, \{m, n\}]` creates an \( m \times n \) matrix of \( c \)'s. Therefore, for the zero vector of length 3, use the command `ConstantArray[0, 3]`. The following example demonstrates how to set up and `RowReduce` an augmented matrix \( M \) corresponding to \( Ax = 0 \).

\[
A = \{(1, 2), (3, 4)\};
\]
\[
b = \text{ConstantArray}[0, 2];
\]
\[
\text{MatrixForm}[\text{augAb} = \text{Transpose}[[\text{Append}[\text{Transpose}[A], b]]]]
\]
\[
\text{MatrixForm}[\text{rrAb} = \text{RowReduce}[\text{augAb}]]
\]
Then the output in matrix form will be
\[
\begin{pmatrix} 1 & 2 & 0 \\
3 & 4 & 0 \\
1 & 0 & 0 \\
0 & 1 & 0 \end{pmatrix}
\]

**SECTION 1.6 Rational Format**

Chemical equation-balance problems are studied best using exact or symbolic arithmetic, because the balance variables must be whole numbers (with no round-off allowed). *Mathematica* will display the exact (rational) value of every entry during row reduction as long as each number you enter is an integer or rational number not containing a decimal point. Once you find a rational solution of a chemical equation-balance problem, you can multiply the entries in the solution by a suitable integer to produce a solution that involves only whole numbers.

**SECTION 1.10 Generating a Sequence**

Recall that we can identify a geometric point \((a, b)\) with the column vector \( \begin{pmatrix} a \\ b \end{pmatrix} \). With *Mathematica*, the ordered pair \((a, b)\) is represented by \{a, b\}. The data for Exercise 13, Section 1.10, is used here to demonstrate how to generate a sequence. Open CISIO.nb and then open the cell corresponding to Exercise 13. Execute the input cell.

\[
M = \{(0.95, 0.03), (0.05, 0.97)\};
\]
\[
xO = \{600000, 400000\};
\]
Suppose you want to create the first 20 terms of the sequence \(\{X_0, X_2, X_3, \ldots\}\) using the linear difference equation \(X_{k+1} = M X_k\) for \(k = 0, 1, 2, \ldots\), where \(X_0 = xO\). You can do this by first creating a function \(x\) satisfying \(x[k] = X_k\) for \(k = 0, 1, 2, \ldots\). When defining a new function with *Mathematica*, it is good practice to first clear out previously defined values of the function using the `Clear` command as demonstrated below. Next, the assignment statement \(x[0] := xO\) defines the value of \(x[0]\) to be \(xO\) and then \(x[1], x[2], x[3], \ldots\) are recursively defined by letting \(x[k_] := M.x[k-1]\). The underscore immediately following \(k\) on
the left hand side of the assignment statement is necessary in order to define k to be a variable and the colon-equals sign(:=), which is called a delayed equals sign, must be used instead of an equals sign (=), when defining a function recursively. The use of the colon before the = also suppresses output.

```
Clear[ x]
x[0] := x0
x[k_] := M.x[k-1]
```

Now that the function x is defined, the value of X_n is retrieved by executing x[n] for a numerical value n.

```
x[100 ]
```

Then the output will be the 100th element of the sequence generated:

```
{ 375054., 624946.}
```

The value of X_{100} is displayed in row form. Notice that Mathematica displays 6 significant digits for each entry in this output. To see 20 digits of precision, replace x[100] with N[x[100], 20]. Executing the Table command generates and displays the values of x[1], x[2], ..., x[n] in a table where x[k] is displayed as the kth ordered pair of the table. For the first 5 pairs of outputs, we do the following, expressing the result in TableForm.

```
TableForm[Table[ x[k],{ k, 1, 5 }]]
```

The output will be

```
582000.  418000
565440.  434560.
550205.  449795.
536188.  463812.
523293.  476707.
```

Note that Mathematica notation for evaluating the function x at k is x[k] where square brackets [...] are used instead of the more natural notation of x(k) that is typically seen in mathematics. Be sure to use square brackets because Mathematica uses round brackets to group terms, but not in defining or using functions.

**SECTION 2.1 Matrix Operations**

Matrix addition and subtraction are performed using the plus (+) and minus (-) signs, respectively. Matrix multiplication is performed using a period (.). To compute A^k for an n x n matrix A, as defined at the end of Section 2.1 in the textbook, execute the command `MatrixPower[A, k]`. Entering the command A^k returns a matrix whose entries are the kth powers of the corresponding entries in A. Remember that matrix addition and subtraction are defined only for matrices of the same size, and matrix multiplication for two matrices, A and B, is defined only if the number of columns of A is the same as the number of rows of B. You will get an error message in Mathematica if you try to do operations that are not valid.

Multiplying a scalar c and matrix A is performed by entering c A, leaving a space between the c and the A. This product can also be produced using an asterisk (*) or parenthesis by executing c * A, c(A) or (c)A.

To compute A^T F, enter `Transpose[A].F` . For vector multiplication, if u and v are vectors in row form and have the same size, then u.v represents their inner product, and the outer product u.v^T is computed with `Transpose[{ u }].{ v }` . This last input looks reversed, but it will yield the correct outer product.

**Special Matrices**

Mathematica has commands that construct many special matrices. For example,
**SECTION 2.2  IdentityMatrix**

As mentioned above, the $n \times n$ identity matrix is denoted by IdentityMatrix[$n$]. If we consider a $3 \times 3$ matrix $A$ augmented by the identity matrix of the same size, we use the `Transpose` command twice as we did before, coupled this time with the `Join` command instead of the `Append` command.

\[
A = \begin{pmatrix}
1 & 2 & -1 \\
-3 & -4 & 2 \\
5 & 2 & 3 \\
\end{pmatrix}
\]

\[
M = \text{Transpose}[\text{Join} [\text{Transpose}[A], \text{IdentityMatrix}[3]]]
\]

MatrixForm[M]

\[
\text{RowReduce}[M]
\]

The output will be

\[
\begin{pmatrix}
1 & 2 & -1 & 1 & 0 & 0 \\
-3 & -4 & 2 & 0 & 1 & 0 \\
5 & 2 & 3 & 0 & 0 & 1 \\
\end{pmatrix}
\]

The $3 \times 3$ array on the right of the last matrix represents the inverse of the matrix $A$ and this will be discussed more in the next section on the inverse of a matrix.

**SECTION 2.3  Inverse**

Determining whether a matrix is invertible is not always a simple matter. A fast and fairly reliable method is to use the `Inverse[A]` command which computes the inverse of $A$, if it exists. If all the entries in $A$ are in exact form (i.e., none of the entries in $A$ contain decimal points) and if $A$ is singular (noninvertible), then a message is displayed stating that $A$ is singular. If at least one of the entries in $A$ contains a decimal point, a warning message is given if $A$ is singular or nearly singular. In the latter case, Mathematica might not be able to find the inverse. But if you first replace each entry in $A$ with its exact, rational equivalent, then the inverse is computed and displayed, even if it is nearly singular.

For large matrices, computing the inverse of $A$ with Mathematica can be slow if all the entries in $A$ are exact, rational numbers. If you have already created a matrix $A$ whose entries are exact integers or exact rational numbers, then the assignment statement $A = \text{N}[A]$ or $A = A // \text{N}$ will convert all entries in $A$ to approximate real numbers. Mathematica will then be able to compute the approximate value the inverse of $A$ more quickly.

For Exercises 41-44, the command `SingularValueList` computes the singular values (discussed in Section 7.4) of a matrix and the condition number of a square matrix $A$, is the ratio of the largest singular value to the smallest singular value. All the entries in $A$ must be approximate real numbers containing a decimal point. Notice the use of the command `//N` here to generate approximate numerical output. Since the singular values are listed in order from largest to smallest, the double bracket command in the third input row will compute the condition number which equals the largest singular value divided by (/)
the smallest singular value. The “-1” designation calls on the last element in the sv list.

A = {{4, 0, -3, -7}, {-6, 9, 9, 9}, {7, -5, 10, 19}, {-1, 2, 4, -1}};
sv = SingularValueList[A] // N
sv[1]/sv[-1]

The output lists the singular values and the ratio of the largest (first) to the smallest (last):

\{25.9609, 14.4566, 4.66997, 2.2868\}
11.3525

To perform the experiment described in Exercise 42, be sure the A contains approximate real numbers and then execute the following Mathematica instructions. RandomInteger[{-10, 10}, 4] generates a 4-dimensional vector with integer entries between -10 and 10.

A = {{4, 0, -3, -7}, {-6, 9, 9, 9}, {7, -5, 10, 19}, {-1, 2, 4, -1}} // N;
x = RandomInteger[{-10, 10}, 4];
b = A.x;
xl = Inverse[A] . b;
x - xl

The only output is from the last input

\{0., 0., -3.55271*10^{-15}, 0.\}

Since A consists of approximate real numbers, the remaining computations will also result in approximate number. Displaying the value of x - xl is the best way to compare x and xl. You can execute the assignment statement again to repeat the experiment and your answers will vary.

Exercises 44 and 45 refer to the Hilbert matrix, which is built into Mathematica. The command HilbertMatrix[5, 5] yields the matrix in Exercise 44.

SECTION 2.4  Partitioned Matrices

The command A[[\{i1, i2, ... , ir\}, {j1, j2, ... , js\}]] creates an r x s submatrix of A whose entries consist of the numbers contained in rows i1, i2, ... , ir and in columns j1, j2, ..., js. Applied to the matrix

A = {{4, 0, -3, -7}, {-6, 9, 9, 9}, {7, -5, 10, 19}, {-1, 2, 4, -1}},

the command A[[\{2,3\},\{1,2\}]] would output the matrix \{-6, 9\}, \{7, -5\}, the entries in the 2nd and 3rd rows and 1st and 2nd columns.

The Append command creates partitioned matrices. For instance, if B, D, F, G, H, and J are matrices having appropriate sizes, the command Append[\{\{B, D, F\}, \{G, H, J\}\}] creates the matrix

\[
\begin{pmatrix}
B & D & F \\
G & H & J
\end{pmatrix}
\]

SECTION 2.5  LU Factorization

Mathematica contains the command LUDecomposition that produces a permuted LU factorization of any square matrix A. LUDecomposition returns a list of three elements. The first element is a combination of upper and lower triangular matrices, the second element is a vector specifying rows used for pivoting (a permutation vector which is equivalent to the permutation matrix), and the third element is an estimate of the condition number. Since the output for this command is somewhat complicated, Example 1 from Section 2.5 will serve as a demonstration of how to interpret and work with the output of this function. We will be interested in the matrix in the first part of the output and will want to look at it in MatrixForm. The command “lu = %[[1]]” assigns lu to the first part of the most
recent output, called on using the percent symbol.

```
mat={{3,-7,-2,2},{-3,5,1,0},{6,-4,0,-5},{-9,5,-5,12}};
LUDecomposition[mat]
MatrixForm[lu = %[[1]]]
```

The output of the LUDecomposition is shown in list form and the lu component is shown in matrix form:

```
{{{3,-7,-2,2},{-1,-2,-1,2},{2,-5,-1,1},{-3,8,3,-1}}, {1,2,3,4}, 1}
```

Looking at the lower and upper triangular matrices shown in Example 1, you can see both matrices built into this output matrix, 

```
{{3,-7,-2,2},{-1,-2,-1,2},{2,-5,-1,1},{-3,8,3,-1}}. The nonzero elements of the upper triangular matrix are displayed in bold; the nonzero and non-diagonal elements of the lower triangular matrix are displayed in italics. The diagonal elements of the lower triangular matrix will be 1’s, so the lower and upper can be defined readily. There are different ways within Mathematica to reconstruct the lower and upper triangular matrices from lu, but we will do it by copying and pasting the lu matrix into our input commands and making appropriate alterations. After defining lower and upper, the last command validates that the original matrix is the product of lower and upper.

If lower=
```
{{1,0,0,0},
{-1,1,0,0},
{2,-5,1,0},
{-3,8,3,1}}
```
and upper=
```
{{3,-7,-2,2},
{0,-2,-1,2},
{0,0,-1,1},
{0,0,0,1}}
```
then lower.upper=
```
{{3722,0212,0011,0001},
{-1121,0011,0001,0001},
{-2511,0011,0001,0001},
{-3831,0011,0001,0001}}
```

If you called on the LinearSolve[A,b] command in Mathematica to solve \(Ax = b\), this command uses the LU decomposition techniques internally and ALS = LinearSolve[A] outputs the LinearSolveFunction which becomes useful when you need to solve \(Ax = b\) repeatedly with a different \(b\) each time. Having named the function ALS, inputting ALS[b] for any vector \(b\) of the proper dimension, outputs the solution to the system. For details, consult the Mathematica documentation. The alternative command \(x = Inverse[A].b\) works, but is less efficient and can be less accurate.

**SECTION 2.6 Solving the Matrix Equation \(x = Ax + d\)**

To form the matrix \((1 – A)^{-1}\) needed to solve \(x = Ax + d\), after defining an \(n \times n\) matrix \(A\), we execute the command coeff=Inverse[IdentityMatrix[n] – A] and multiply this by the vector \(d\).

**SECTION 2.8 MatrixRank, and NullSpace**

As we have seen, the command RowReduce[A] produces the reduced row echelon form of \(A\). From this matrix, a basis for the column space of \(A\) is obtained. If the entries of \(A\) are all exact, rational numbers,
RowReduce[A] reveals the rank of A, since Mathematica performs exact arithmetic in finding the reduced row echelon form of A. If at least one of the entries of A contains a decimal point, then roundoff error or an extremely small pivot entry can produce an incorrect echelon form.

Mathematica also contains the command MatrixRank for determining the rank of a matrix, and the command NullSpace for determining a list of vectors that form a basis for the null space of a matrix. Consider the following example using the matrix in Exercise 11 from Section 2.8.

\[ A = \begin{pmatrix} 3 & 2 & 1 & -5 \\ -9 & -4 & 1 & 7 \\ 9 & 2 & -5 & 1 \end{pmatrix} \]

\[ \text{MatrixForm[RowReduce[A]]} \]
\[ \text{MatrixRank[A]} \]
\[ \text{NullSpace[A]} \]

The output for this is the matrix A, its rank and basis for the null space:

\[
\begin{pmatrix}
1 & 0 & -1 & 1 \\
0 & 1 & 2 & -4 \\
0 & 0 & 0 & 0
\end{pmatrix}
\]

2
\{\{-1, 4, 0, 1\}, \{1, -2, 1, 0\}\}

SECTION 3.1 Computing Determinants

For evaluation of the determinant of a matrix A, you can use the Mathematica command Det[A], but you can also explore the process of expansion by minors by first defining an n x n matrix A and then using the Mathematica commands Minors[A, n-1, Identity] and Minors[A]. Consider the matrix similar to those in Section 3.1. To use a blank matrix for input, select the menu item Insert and choose the Table/Matrix-New option.

\[ A = \begin{pmatrix} 1 & 3 & 5 \\ 2 & 1 & 1 \\ 3 & 4 & 2 \end{pmatrix} ; \]

\[ \text{MatrixForm[Minors[A, 2, Identity]]} \]
\[ \text{MatrixForm[mA = Minors[A]]} \]

The output will be

\[
\begin{pmatrix}
1 & 3 & 1 & 5 & 3 & 5 \\
2 & 1 & 2 & 1 & 1 & 1 \\
3 & 4 & 1 & 5 & 3 & 5 \\
1 & 3 & 1 & 5 & 3 & 5 \\
3 & 4 & 1 & 5 & 3 & 5 \\
2 & 1 & 2 & 1 & 1 & 1 \\
3 & 4 & 2 & 4 & 2 \\
-5 & -9 & -2 \\
-5 & -13 & -14 \\
5 & 1 & -2
\end{pmatrix}
\]
If you look back at the matrix A, you will see that the 2 \times 2 submatrix in the first output whose determinant is used to compute the cofactor for the element in the first row, first column of A is in the bottom right-hand corner of the matrix in the first output. In fact, the three submatrices in the bottom row of the first output correspond, in reverse order, to the matrices whose determinants are used to compute the cofactors for the terms in the first row of A. The values of the determinants of the submatrices are given in the final matrix. We will use these ideas to compute the determinant of A using cofactors and verify our answer with the \texttt{Det[A]} command. To extract the proper elements from A and mA, we will use the double brackets and use spaces for products of terms from A and their corresponding cofactors. Also note the negative in our middle term, since the cofactor for the second term in the first row is negative.

\[
A[[1,1]] \ mA[[3,3]] - A[[1,2]] \ mA[[3,2]] + A[[1,3]] \ mA[[3,1]]
\]

Whose outputs are

\[
\begin{align*}
20 \\
20
\end{align*}
\]

\textit{Mathematica} also has a \texttt{Cofactor} command, but this command is within a package and not part of the standard kernel. To use the \texttt{Cofactor} command, you need to first load the \textit{Combinatorica Package}. We do this with the next command; you may ignore the error message you get when this package is loaded.

\texttt{Needs["Combinatorica"]}

Now you can use the \texttt{Cofactor} command. The input below assumes that A is an 3 \times 3 matrix. If we want to compute the determinant of A by the cofactor expansion around the third column of A, we can use the \texttt{Sum} command and use the double brackets to call upon the \([i, j]\) entries of the original matrix. Here, \(j = 3\), since we want the expansion around the third column, and we let \(i\) go from 1 to 3.

\[
\text{Sum}[A[[i,3]] \ \text{Cofactor}[A,\{i,3\}],\{i,1,3\}]
\]

\textbf{SECTION 3.3 The Adjugate Matrix}

Referring to the Cofactor command discussed under Section 3.1, we can use this command to create a matrix of cofactors, which is referred to as the adjugate matrix associated with a matrix A. We would first load the \textit{Combinatorica Package}, \texttt{Needs["Combinatorica"]}, and then proceed as in the following example similar to those from Section 3.3

\[
A = \{(3,6,7),(0,2,1),(2,3,4)\}
\]

\texttt{MatrixForm[adjA=Table[Cofactor[A,\{i,j\}],\{i,1,3\},\{j,1,3\}]]}

Using the adjugate matrix formed, we can determine the inverse of A

\texttt{MatrixForm[1/Det[A] \ Transpose[adjA]]}

\textbf{SECTION 4.1 Graphing Functions}

You can use the following commands to define and plot the function \(f\) given in Exercise 37. Be sure to use the underscore after your variable on the left side of the equation when defining a function. The cosine function, denoted \texttt{Cos[ ]} is a built-in function in \textit{Mathematica} as is the sine function, denoted by \texttt{Sin[ ]}. Notice that the squared sign comes after the functions argument. To insert a superscript, either select the button from the Basic Math Input Palette or use the ^ key before the superscript. When plotting a function, you must always include the independent variable and domain over which you want the function plotted in curly braces after the function and comma. For the symbol \(\pi\), you can either select...
the \( \pi \) symbol from the Basic Math Input Palette or type Pi. We can then Plot the function over a domain.

\[
f[t_] := 1 - 8 \cos[t]^2 + 8 \cos[t]^4
\]
\[
\text{Plot}[f[t], \{t, 0, 2\pi\}]
\]

SECTION 4.2 Null Space and Column Space

Recall from Section 2.8 the Mathematica command \texttt{NullSpace[A]} that outputs a list of vectors forming a basis for the null space of \( A \). If \( A \) is in exact form, the product of \( A \).\texttt{NullSpace[A]} will be a zero matrix. But if \( A \) contains decimals, the product will not yield a matrix with entries that are exactly zero.

You can determine a set of vectors for a basis for the column space of \( A \) by \texttt{RowReduce[Transpose[A]]}. The nonzero rows of the result will form a basis for the column space of \( A \).

SECTION 4.3 Using the Solve Command on Equations

For Exercise 37 in Section 4.3, an assignment statement is used to define a vector function. Then we will use a \texttt{Table} command to generate a matrix of coefficients for the constants \( c_1, c_2, c_3, \) and \( c_4 \). In the first case, we will let \( t \) take on values from 0 to 2 (in jumps of 1, by default). We will then define our matrix equation to satisfy equation (5) and use the Solve command to compute values of the constants.

\[
fns[t_] := \{t, \sin[t], \cos[2t], \sin[t] \cos[t]\}
\]
\[
\text{Table}[fns[t], \{t, 0, 2\}];
\]
\[
eq q2 = fnt2.\{c1, c2, c3, c4\} == \text{ConstantArray[0,3]} // \text{N};
\]
\[
\text{Solve[q2, \{c1, c2, c3, c4\}]}
\]

Approximate solutions for the constants are given in the output.

\[
\{\{c1 \to 0. + 0.945944 c4, c2 \to 0. - 1.66446 c4, c3 \to 0.}\}
\]

The above output shows that not all the constants have to be zero in order for equation (5) to be true. Note that two consecutive equal signs, \( == \), represent an equal sign appearing in an equation. The first input of the Solve command indicates the equations to be solved and the second input specifies the variables for which to solve. We will now create a table with the function evaluated at four points \( (t = 0, 1, 2, 3) \) and see what happens. We also had to update our zero-vector to dimension 4.

\[
fnt3 = \text{Table}[fns[t], \{t, 0, 3\}];
\]
\[
eq q3 = fnt3.\{c1, c2, c3, c4\} == \text{ConstantArray[0,4]} // \text{N};
\]
\[
\text{Solve[q3, \{c1, c2, c3, c4\}]}
\]

This results show that all the constants have to be zero.

\[
\{\{c1 \to 0., c2 \to 0., c3 \to 0., c4 \to 0.\}\}
\]

SECTION 4.4 LinearSolve

The command \texttt{LinearSolve[A, b]}, introduced in the discussion for Section 2.5, solves \( Ax = b \) where \( A \) is an \( m \times n \) matrix. If \( Ax = b \) is inconsistent, then a message is returned stating that a solution does not exist. If more than one solution exists, one possible solution is returned. LinearSolve has additional capabilities that will be introduced later, after you have the appropriate background.

SECTION 4.6 Rank and Random Matrices

As was mentioned in the discussion for Section 2.8, Mathematica contains the \texttt{MatrixRank[A]} command. If the entries of \( A \) are in exact, rational form, \texttt{RowReduce[A]} also reveals the rank of \( A \) and the rank of its null space, since Mathematica performs exact arithmetic in finding the reduced row
echelon form of $A$. If at least one of the entries of $A$ contains a decimal point, then roundoff error or an extremely small pivot entry can produce an incorrect echelon form. The command `NullSpace[A]` outputs a list of vectors that form a basis for the null space of a matrix, thus demonstrating the dimension of the null space of $A$.

The command `RandomInteger[ {a, b}, {m, n}]` can be used to create an $m \times n$ matrix containing random integers between $a$ and $b$. So, for example, `RandomInteger[ {-9, 9}, {3, 2}]` creates a 3 x 2 matrix with random integers between -9 and 9.

**SECTION 4.7 Change-of-Coordinates Matrix**

The notebook C4S7.nb has data for Exercises 7-10 and 17-19. Given vectors $c_1$, $c_2$, $b_1$, and $b_2$ in $\mathbb{R}^2$, the command `Mcb=Transpose[{c1,c2,b1,b2}]` produces a 2 x 2 matrix whose columns are $c_1$, $c_2$, $b_1$, and $b_2$, respectively. The command `RowReduce[Mcb]` row reduces $Mcb$ showing the 2 x 2 identity matrix on the left and the transition matrix $P_{C \leftarrow B}$ on the right. Similarly, if `Mbc=Transpose[{b1, b2, c1, c2}]`, then `RowReduce[Mbc]` displays the identity matrix on the left and the transition matrix $P_{B \leftarrow C}$ on the right.

**SECTION 4.8 Solve**

In Exercises 7-16 and 25-28, the coefficients of the polynomial in the auxiliary equation are stored in row vector form, with coefficients in ascending order. For instance, the set of coefficients of the polynomial from Exercise 16 is represented by $p = (-3, 8, 16)$. We can create a vector containing powers of $x$ of the same length, $xv = \{1, x, x^2\}$ and then use the `Mathematica` command `Solve`, which outputs the roots of the polynomial described by $p$. Note the use of the double equal sign in the equation and the variable after the comma specifying the variable for which we wish to solve.

\[
p = \{-3, 8, 16\};
\]
\[
xv = \{1, x, x^2\};
\]
\[
Solve[p.xv == 0, x]
\]

The output is
\[
\{(x \rightarrow -4/3), (x \rightarrow 4)\}
\]

Another perspective would be to recreate the difference equation by forming a table with the solutions given and asking for a solution to the system of difference equations formed. Consider Exercise 8.

\[
\text{fntab} = \text{Transpose}[\text{Table}[\{2^k, 4^k, (-5)^k\}, \{k, 0, 3\}]];
\]
\[
\text{fntab} // \text{MatrixForm}
\]
\[
\text{Solve}[\text{fntab}\cdot\{c1, c2, c3, c4\} == \{0, 0, 0\}, \{\{c1, c2, c3, c4\}\}]
\]

The output is

\[
\begin{pmatrix}
1 & 2 & 4 & 8 \\
1 & 4 & 16 & 64 \\
1 & -5 & 25 & -125
\end{pmatrix}
\]

\[
\{(c2 \rightarrow -((11 c1)/20), c3 \rightarrow -(c1/40), c4 \rightarrow c1/40)\}
\]

Letting $c1=40$, the coefficient list is $\{40, -22, -1, 1\}$, which are the coefficients in the difference equation.

**SECTION 4.9 MatrixPower and Matrices with Randomly Generated Entries**

See the discussion presented in Section 1.10 for information on how to use `Mathematica` for these exercises. For a matrix such as those in Exercise 21, you can use the `MatrixPower` command to arrive at an approximation to the steady-state vector. We look at the 100th power of $Q$ below. Note that column entries of $Q^{100}$ are very close in value

\[
Q = \{\{.97, .05, .10\}, \{0, .90, .05\}, \{.03, .05, .85\}\};
\]
5.2 • The Characteristic Polynomial and Plot

MatrixPower[Q, 100]

The list form of \( Q^{100} \) is

\[
\begin{align*}
\{0.735312, 0.735215, 0.735259, \\
0.0882205, 0.0883003, 0.0882644, \\
0.176467, 0.176485, 0.176477\}
\end{align*}
\]

In Exercise 22, you can use the `Random[]` command to generate random numbers between 0 and 1 and call on this inside a `Do` loop. We begin with an empty matrix and we set \( n = 2 \) to generate 2 \( \times \) 2 matrices, each of whose row entries sum to one (accomplished by dividing each row generated by the sum of its entries). The randomly generated Markov matrix will be the Transpose of the `mm` matrix. You may change the value of \( n \) to create a larger matrix. The `Do` loop has no output.

```mathematica
mm = {}; n = 2;
Do[r = Table[Random[], {n}], rn = c/Sum[c[[i]], {i, 1, n}], mm = AppendTo[mm, rn], {n}]
MatrixForm[mk = Transpose[mm]]
```

The output will be the matrix:

\[
\begin{pmatrix}
0.517986 & 0.835665 \\
0.482014 & 0.164335
\end{pmatrix}
\]

Each time you execute this command, even with the same value of \( n \), the results will differ.

SECTION 5.1 Eigenvectors and Eigenvalues

To compute eigenvalues, eigenvectors, or both together, `Mathematica` has three built-in commands, `Eigenvalues`, `Eigenvectors`, and `Eigensystem`. Consider the matrix from Exercise 37.

```mathematica
mat = {{8, -10, -5}, {2, 17, 2}, {-9, -18, 4}};
vals = Eigenvalues[mat]
vecs = Eigenvectors[mat]
```

The output is the list of the eigenvalues and the list of eigenvectors:

\[
\begin{align*}
\{13, 13, 3\} \\
\{-1, 0, 1\}, \{-2, 1, 0\}, \{5, -2, 9\}
\end{align*}
\]

Notice that the eigenvalues are listed in order from largest to smallest, although, in this example, there is a repeated eigenvalue. In the second output, the first vector in the list is the eigenvector corresponding to the largest eigenvalue, the second corresponds to the second largest, etc. You can verify this by multiplying the matrix by each vector.

A single command yields both the eigenvalues and eigenvectors:

```mathematica
MatrixForm[Eigensystem[mat]]
```

The `MatrixForm` of this result enables us to see which eigenvalue is associated with which eigenvector.

\[
\begin{pmatrix}
13 & 13 & 3 \\
\{-1, 0, 1\} & \{-2, 1, 0\} & \{5, -2, 9\}
\end{pmatrix}
\]

SECTION 5.2 The Characteristic Polynomial and Plot

To obtain the characteristic polynomial of an \( n \times n \) matrix \( A \), execute `Det[A - \lambda IdentityMatrix[n]]`, leaving a space between \( \lambda \) and `IdentityMatrix[n]` for scalar multiplication. You can use this to check your answers in Exercises 9-14. For example, if \( A \) is the 3 \( \times \) 3 matrix in Exercise 13, we can do the following.
\[ A = \begin{bmatrix} 6 & -2 & 0 \\ -2 & 9 & 0 \\ 5 & 8 & 3 \end{bmatrix}; \]
\[ \text{cpoly} = \text{Det}[A - \lambda \text{IdentityMatrix}[3]] \]

The output is the characteristic polynomial:
\[ 150 - 95 \lambda + 18 \lambda^2 - \lambda^3 \]
We can plot this characteristic polynomial and also find its roots.
\[
\text{Plot[cpoly, \{\lambda, 0, 12\}]}
\]
\[
\text{Solve[cpoly==0, \lambda]} \]

You can observe the graph and its zeros:
\[
\{(\lambda \rightarrow 3), (\lambda \rightarrow 5), (\lambda \rightarrow 10)\}
\]

There is also a command in \textit{Mathematica} called \texttt{CharacteristicPolynomial} and we can specify the independent variable (\(\lambda\) or \(x\) or...). For Exercise 30, the steps would be similar, except that there is a variable \(a\) in the matrix. When we ask for the characteristic polynomial, we will get a function of \(x\) and \(a\).

\[ A = \begin{bmatrix} -6 & 28 & 21 \\ 4 & -15 & -12 \\ -8 & a & 25 \end{bmatrix}; \]
\[ \text{cpoly} = \text{CharacteristicPolynomial}[A, x] \]
gives the output
\[ -382 + 12 a + 379 \lambda - 12 a \lambda + 4 \lambda^2 - \lambda^3 \]
To plot this function, we will let \(a\) take on different values and create a set of functions we will graph. The use of the command \texttt{cpoly/.a\rightarrow\text{takes the polynomial in Out}[2] and evaluates it at the values of \(a\) specified. We could have used all five values specified in Exercise 30; then we would specify five colors or styles. Many color names and styles are built into \textit{Mathematica} and there are other ways in which you may call on them with the \texttt{PlotStyle} option.

\[
\text{clist=cpoly/.a\rightarrow\{31.8, 32, 32.2\};}
\]
\[
\text{Plot[clist, \{\lambda, 0, 3\}, PlotStyle \rightarrow \{\text{Red, Green, Blue}\}]}
\]

The graphs are shown below. You could also have created each plot separately, giving each a name such as \(p1, p2, p3\) and then use the \texttt{Show[p1, p2, p3]} command to show them together.
SECTION 5.3  Diagonalization and Eigensystem

To practice the diagonalization procedure in this section, we will use the Eigensystem command to enable us to generate the matrices to diagonalize a matrix. See the Section 5.1 discussion. Notice how the top row of the eigensystem output is defined as evals (the set of eigenvalues), while the bottom row of the matrix is defined to be evecs (the set of eigenvectors). The matrix A corresponds to Exercise 34 in Section 5.3. The matrix P will be the transpose of the set of eigenvectors; the eigenvalues are the elements in the diagonal matrix (mD).

\[
A = \begin{pmatrix}
0 & 13 & 8 & 4 \\
4 & 9 & 8 & 4 \\
8 & 6 & 12 & 8 \\
0 & 5 & 0 & -4
\end{pmatrix};
\]

\{\text{evals, evecs}\} = \text{Eigensystem}[A];
\text{MatrixForm}[P = \text{Transpose}[\text{evecs}]]
\text{mD} = \text{DiagonalMatrix}[\text{evals}]

The matrix P will be output along with the diagonal matrix mD in list form.

\[
\begin{pmatrix}
28 & -1 & -2 & 1 \\
28 & 0 & 0 & 1 \\
36 & 0 & 1 & -2 \\
5 & 1 & 0 & 1
\end{pmatrix}
\]

\{(24,0,0,0),(0,-4,0,0),(0,0,-4,0),(0,0,0,1)\}

You can verify your results by computing P.mD.Inverse[P]//Simplify. This product will equal the matrix A. The //Simplify command is sometimes needed to simplify the products. A warning message will occur if matrix P in not invertible. For large matrices, you may want to use decimals.

SECTION 5.5  Complex Eigenvalues, Re and Im

The letter I is a protected Mathematica symbol representing the imaginary number i. The matrix, A, in Exercise 1 is a 2 x 2 real matrix that contains complex eigenvalues, \(a \pm i b\) and complex eigenvectors. To build a 2 x 2 real matrix P from the real and imaginary parts of these eigenvalues (evals) and eigenvectors (evecs), we use the Mathematica expressions \(\text{Re}[\text{evals}]\) and \(\text{Re}[\text{evecs}]\) and \(\text{Im}[\text{evals}]\) and \(\text{Im}[\text{evecs}]\), respectively. The center matrix, termed mD here, will not be a diagonal matrix. Its entries on the main diagonal will be the real part (a) of the eigenvalues, while the entries on the other diagonal contain the imaginary parts (-b and +b) of the eigenvalues. In general, you can compute mD as \(P^T A P\).

\[
A = \{(1,-2),(1,3)\};
\{\text{evals, evecs}\} = \text{Eigensystem}[\text{mat}]
\text{Re}[\text{evals}]
\text{Im}[\text{evals}]
P = \text{Transpose}[[\text{Re}[\text{evecs}][[1]],\text{Im}[\text{evecs}][[2]]]]
\text{Inverse}[P]
mD = \text{Inverse}[P].A.P
\]

No output is shown here. You can verify your results by computing P.mD.Inverse[P]//Simplify.

SECTION 5.6  Plotting Discrete Trajectories

For a given 2 x 2 matrix \(A\), if \(x_{k+1} = A \cdot x_k\) then the following commands are used to create \(x_k\) for \(k = 1, 2, 3, \ldots\)
As an example, the matrix $A$ and initial vector $x_0$ defined are found in Example 6 in your textbook. Because of the way that $x$ will be used, it is critical that we **Clear** any previous assignments for $x$. The purpose of the next set of commands is to define a function $x$ where $x[k]$ represents $x_k$. Use of the **underscore** command here is critical. We then use the **Table** command to define a list called $pts$ representing $x[0]$ through $x[50]$. Then the command **ListPlot[pts]** is used to plot the points in the xy-plane and the trajectory is assigned the name $traj1$. The optional **PlotStyle** command increases the size of the displayed points and the **PlotRange→All** command is used to ensure that the entire trajectory is displayed.

```
Clear[x, A]
A = {{.8, .5},{-.1, 1}};
xO={0, 2.5};
x[0] = xO;
x[i_] := A.x[i -1]
pts = Table[ x[i], { i, 0, 50 } ];
traj1 = ListPlot[ pts, PlotStyle → PointSize[.02], PlotRange → All]
```

The graph of the first 50 iterations starts at $(0, 2.5)$ and ends close to $(0,0)$

To create another trajectory, repeat the steps above with a new value of $x_0$, replacing $traj1$ with $traj2$ to represent the new trajectory. Repeating this process to obtain $traj3$, $traj4$, etc. you can use **Show[traj1, traj2, traj3, traj4]** to display all the trajectories on a single plot. You can alter the command **PlotStyle → PointSize[.02]** to get red points. Other named colors can be used or the option **RGBColor[a,b,c]** can also be used, with a, b, and c numbers for red, green, and blue tones.

For Exercise 17, create $x[0]$ through $x[8]$ representing $x_0$, ..., $x_8$. Then $x[0][[1]]$ through $x[8][[1]]$ represent the number of juveniles in years 0 through 8, respectively, and $x[0][[2]]$ through $x[8][[2]]$ represent the number of adults in years 0 through 8, respectively. The following commands will produce a graph of the juvenile population.

```
Clear[x, A]
A = {{0, 1.6},{.3, 1.1}};
xO={15, 10};
x[0] = xO;
x[i_] := A.x[i -1]
juv = Table[ {i, x[i][[1]]}, { i, 0, 8 } ];
ListPlot[ juv, PlotStyle → PointSize[.02]]
```

To graph the sum of the two entries in each of $x[0]$ through $x[8]$, execute the following two commands.

```
sum = Table[ {i, x[i][[1]]+x[i][[2]]}, { i, 0, 8 } ];
ListPlot[ sum, PlotStyle → PointSize[.02]]
```

To graph the ratio of juveniles to adults, execute the following, where / is the division key.

```
ratio = Table[ {i, x[i][[1]]/ x[i][[2]] }, { i, 0, 8 } ];
ListPlot[ratio, PlotStyle → PointSize[.02]]
```
SECTION 5.7 Solutions to Differential Equations

To find the eigenvalues and eigenvectors of \( A \), use \{evals, evecs\} = Eigensystem\[A\]. If an eigenvalue is complex, the corresponding eigenvector will be complex. The real and imaginary parts of evecs are \text{Re}[evecs] \text{ and } \text{Im}[evecs]\]. respectively. Your eigenvectors will be in row form and they should be multiples of those in the text's answers (when the eigenspaces are one-dimensional). To test whether a vector \( v \) is a multiple of a vector \( w \), compute \( v/w \). This divides each entry in \( v \) by the corresponding entry in \( w \). If \( v \) is a multiple of \( w \), the result of \( v/w \) should be a vector whose entries are all equal.

SECTION 5.8 The Power Method

The commands \text{Eigenvalues}[A] \text{ or } Eigensystem[A] \text{ in } Mathematica \text{ output the numerical eigenvalues sorted in order of decreasing absolute value, so the first eigenvalue in the list will be the one with largest absolute value. Also, the command } \text{Max}\{\text{Abs}\[\text{list}\]\} \text{ would output the element in list with the largest absolute value.}

Below is an application of the power method algorithm to Example 2 from Section 5.8. The goal of this string of commands is to create a list \( \text{lis} \) (avoid \text{List}, which is a built-in Mathematica command) that contains the sequence of numbers and vectors converging to the dominant eigenvalue and its corresponding eigenvector. We can use the \text{Do} command as we did before. The value of \( y \) stores the current value of \( Ax_k \), \( \mu \) stores the current value of \( \mu x \) and then Append[lis, \{\mu, x\}] appends the values of \( \mu \) and \( x \) to lis. Finally \( x \) is overwritten with \((1/\mu) y\), representing \( x_{k+1} \) in the power method algorithm. This is repeated six times. After the \text{Do} loop, the \text{lis} statement causes the list to appear in an output statement.

\[
\begin{align*}
\text{Clear}[A, x, y]; \\
A &= \{\{6, 5\}, \{1, 2\}\}; \\
\text{lis} &= \{\}; \\
x &= \{0., 1.\}; \\
\text{Do[} \{y=A.x, \mu=\text{Max}\{\text{Abs}\[y]\}, \text{lis=Append[lis, \{\mu, x\}]}, \\
x = (1/\mu) y, \{6\}\} \\
\text{lis}
\end{align*}
\]

The output gives a list of the sequence of numbers and vectors corresponding to the dominant eigenvalue and its corresponding eigenvector:

\[
\{5., \{0., 1.\}\}, \{8., \{1., 0.4\}\}, \{7.125, \{1., 0.225\}\}, \{7.01754, \\
\{1., 0.203509\}\}, \{7.0025, \{1., 0.2005\}\}
\]

Notice that the last element of \( \text{lis} \) is very close to the largest eigenvalue of \( A \), 7, with eigenvector \( \{1., .2\} \).

The Inverse Power Method

The inverse power method is a modification of the power method. The following code corresponds to Example 3 from Section 5.8. It will seek out the smallest eigenvalue and corresponding eigenvector.

\[
\begin{align*}
\text{Clear}[A, x, y]; \\
A &= \{\{10, -8, -4\}, \{-8, 13, 4\}, \{-4, 5, 4\}\}; \\
\text{lis} &= \{\}; \\
x &= \{1., 1., 1.\}; \\
\alpha &= 1.9; \\
\text{Do[} \{y=\text{LinearSolve}[A-\alpha \text{IdentityMatrix}[3], x], \\
\mu=\text{Max}\{\text{Abs}\[y]\}, v=\alpha+1/\mu, \\
\text{lis=Append[lis, \{v, x\}]}, x=(1/\mu) y, \{5\}\} \\
\text{lis}
\end{align*}
\]

Copyright © 2016 Pearson Education, Inc.
The output gives a list of the sequence of numbers and vectors corresponding to the dominant
eigenvalue and its corresponding eigenvector:

\[
\{\{2.02889,\{1.,1.,1.\}\},\{2.00081,\{0.57359,0.0646492,1.\}\}, \\
\{2.00005,\{0.50536,0.0044530,1.\}\},\{2.,\{0.50038,0.00031281,1.\}\}, \\
\{2.,\{0.500027,0.0000220072,1.\}\}\}
\]

The last element of lis is very close to the smallest eigenvalue of A, 2, with eigenvector \{.5, 0, 1\}.

### SECTION 6.1 Inner Product and Norm

If \( \mathbf{u} \) and \( \mathbf{v} \) are vectors in row form, then \( \mathbf{u} \cdot \mathbf{v} \) is their inner product. The length or norm of \( \mathbf{v} \) is \( \sqrt{\mathbf{v} \cdot \mathbf{v}} \).

You can also determine the length or norm of a vector with the built-in command \texttt{Norm[v]}.  

### SECTION 6.2 Orthogonality

In Exercises 1-9 and 17-22, a quick way in Mathematica to test a set \{u1, u2, u3\} for orthogonality is to
create a matrix \( U = \text{Transpose}\{u1, u2, u3\} \), where \( u1, u2, \) and \( u3 \) represent the vectors from the set in row form. Then test whether \( \text{Transpose}[U] \cdot U \) is a diagonal matrix. \textit{Note:} \( U \) is constructed using the Transpose command in order to produce a matrix whose \textit{columns} are \( u1, u2, ... \), \( \text{un} \).

In general, for row vectors \( \mathbf{y} \) and \( \mathbf{u} \), the orthogonal vector projection of \( \mathbf{y} \) onto \( \mathbf{u} \) is \( (\mathbf{y} \cdot \mathbf{u})/(\mathbf{u} \cdot \mathbf{u}) \mathbf{u} \) or, you can use the \textit{Mathematica} command \texttt{Projection[y, u]}.

### SECTION 6.3 Orthogonal Projections

The orthogonal projection of a single vector onto a single vector is described in the \textit{Mathematica} note for Section 6.2. The orthogonal projection onto the set spanned by an orthogonal set of vectors is the sum of the one-dimensional projections. Another way to construct this projection is to normalize the orthogonal vectors, place them in the columns of a matrix \( U \), and use Theorem 10. Consider the vectors given in Exercise 4.

\[
\mathbf{y} = \{6,3,-2\}; \\
\mathbf{u1} = \{3,4,0\}; \\
\mathbf{u2} = \{-4,3,0\}; \\
U = \text{Transpose}\{[\mathbf{u1}/\text{Norm[\mathbf{u1}]], \mathbf{u2}/\text{Norm[\mathbf{u2}]]}\} \\
\text{proj} = (\mathbf{u} \cdot \text{Transpose}[\mathbf{u}]) \mathbf{y}
\]

The last command produces the orthogonal projection of \( \mathbf{y} \) onto \( \text{Span}\{\mathbf{u1}, \mathbf{u2}\} \). (The parentheses used in the command speed up the computation by avoiding a matrix-matrix product.)

### SECTION 6.4 The Gram-Schmidt Process

Let \( A \) be a matrix with \( n \) linearly independent columns. \textit{Mathematica} has a command \texttt{Orthogonalize[A]} and its default method is the Gram-Schmidt basis. For practice in learning the general procedure, we will consider Exercise 10 in Section 6.4. We begin by selecting the columns of \( A \) and, for simplicity, we will normalize each column. Then we use the Gram-Schmidt technique and normalize the output vectors \{\( v1, v2, v3 \)\}.

\[
A = \{\{-1,6,6\},\{3,-8,3\},\{1,-2,6\},\{1,-4,-3\}\} \\
Ac = \text{Table}[\text{Transpose}[A][[k]], \{k,1,3\}] \\
\{x1,x2,x3\} = \text{Table}[Ac[[i]]/\text{Norm[Ac[[i]]]], \{i,1,3\}] \\
v1 = x1; \\
v22 = x2-(x2.v1)v1;
\]
v2=v22/Norm[v22];
v33=x3-(x3.v1) v1-(x3.v2)v2;
v3=v33/Norm[v33];
MatrixForm[{v1,v2,v3}]/N

The decimal output in matrix form is:

\[
\begin{pmatrix}
-0.288675 & 0.866025 & 0.288675 & 0.288675 \\
0.866025 & 0.288675 & 0.288675 & -0.288675 \\
-0.288675 & -0.288675 & 0.866025 & -0.288675
\end{pmatrix}
\]

If you execute the Mathematica command Orthogonalize[Transpose[A]], you will get the same output.

Another built-in Mathematica command QRDecomposition[A] outputs two matrices, the first of which is the same as the Gram-Schmidt matrix just generated. When the Transpose of this matrix is multiplied by the second matrix generated by QRDecomposition[A], the product yields the original matrix A.

SECTION 6.5 Finding a Least Squares Solution

To find the projection of a vector onto the columns of a matrix, refer to the discussion for Section 6.3. To find the least squares solution, you should first determine whether or not A has linearly dependent columns. If the columns are linearly independent you can use the LinearSolve command to find a solution to the normal equations \( A^T Ax = A^T b \). When defining A, assign AT=Transpose[A], the LinearSolve[AT.A, AT.b].

Regardless of whether or not the columns of A are linearly independent, you can do the following. After defining A and AT (transpose of A), create the augmented matrix \( A_{aug} = Join[AT.A, {AT.b}] \) then use RowReduce[Transpose[Aaug]] and you can read the solution from the output.

For Exercises 15 and 16, see the Numerical Note following Theorem 15 in the text. After defining A and b, execute \((q,r) = QRDecomposition[A]\), a command mentioned in the discussion for Section 6.5. After determining q and r, use the command LinearSolve[r, Transpose[q].b] to solve \( Rx = Q^T b \).

The command \( A = Join[{{AI}, {A2}}] \) can be used in Exercise 26 to create a partitioned matrix whose top block is AI and whose bottom block is A2.

SECTION 6.6 Least Squares Solutions and Functions of Vectors

Once you create the design matrix X and the observation vector y, your computations for the least-squares solutions are the same as those described the discussion for Section 6.5. Here, A and b are replaced by X and y, respectively. When X has linearly independent columns, the Mathematica commands XT=Transpose[X], the LinearSolve[XT.X, XT.b] will solve the system. If the columns of X are linearly dependent, instead of using LinearSolve, after defining XT, you need to first create the augmented matrix \( X_{aug} = Join[XT.X, {XT.b}] \) and then apply RowReduce[Transpose[Xaug]] and read the solution from the output.

To construct the design matrix for certain exercises in this section, you may need Mathematica's ability to compute functions of vectors. If x is a vector and k is a positive integer, then \( x^k \) is a vector the same size as x whose entries are the kth powers of the entries in x. The function Cos[x]k was mentioned in the Mathematica notes in Section 4.3; Sin[x] is the expression for the sine function. The exponential function, Exp[x], and the natural logarithm function, Log[x], also act on each entry in x. The entries in the vector Exp[-.02 x], for example, are computed by applying the function \( e^{-0.02x} \) to the corresponding entries in x. You can enter \( e^{-0.02x} \) directly from the keyboard in Mathematica by selecting the □.
expression followed by the script \( e \) symbol from the Basic Math Input Palette. It should be noted that \( \text{Log}[x] \) automatically assumes the natural log. There is no “\( \ln \)” command. If you want the base-10 logarithmic function, you need the command \( \text{Log}[10, x] \)

### SECTION 6.7 Integrate Command

The \textit{Mathematica} command \texttt{Integrate[f[t], \{t, a, b\}]} will attempt to compute \( \int_a^b f(t) dt \) when \( f \) is integrable on \([a, b]\). Using the Basic Math Input Palette, you can enter the integral in a more natural way: \( \int_a^b f(t) dt \). To solve Exercise 28 with the help of \textit{Mathematica}, begin by creating an inner product function where \( \langle f, g \rangle = \int_a^b f(t) g(t) dt \). The assignment statement for \( \text{ip}[f, g] \) below assumes that \( t \) is the variable of integration. In the \texttt{Integrate} command, put a space between \( f \) and \( g \) for multiplication.

```mathematica
Clear[ip, f, g, t, p1, p2, p3, p4, q1, q2, q3, q4];
ip[f_, g_] := Integrate[f g, \{t, 0, 2\pi\}];
\{p1, p2, p3, p4\} = Table[Cos[t] k, \{k, 0, 3\}];
```

Then perform the Gram-Schmidt process in a similar fashion as illustrated in the \textit{Mathematica} notes for Section 6.4, but using the newly defined inner product. The input steps are shown here:

\[
\begin{align*}
q1 &= p1; \\
q2 &= p2 - \frac{\text{ip[p2, q1]}}{\text{ip[q1, q1]}} q1; \\
q3 &= p3 - \frac{\text{ip[p3, q1]}}{\text{ip[q1, q1]}} (q1 - \frac{\text{ip[p3, q2]}}{\text{ip[q2, q2]}} q2); \\
q4 &= p4 - \frac{\text{ip[p4, q1]}}{\text{ip[q1, q1]}} (q1 - \frac{\text{ip[p4, q2]}}{\text{ip[q2, q2]}} q2 - \frac{\text{ip[p4, q3]}}{\text{ip[q3, q3]}} q3); \\
\end{align*}
\]

\[
\{q1, q2, q3, q4\} // \text{Simplify}
\]

The output will show the set of orthogonal functions, first as they were computed, and second in simplified form using trigonometric identities.

### SECTION 6.8 Defining Discontinuous Functions and Computing Fourier Series

For Exercise 10, you can define the square wave function in \textit{Mathematica} in more than one way. Using the \texttt{Piecewise} command, you could make the function assignment as a list of rules. Another way to define such functions is to use the \texttt{UnitStep} function. This is a widely used function in mathematics and \texttt{UnitStep[t]} takes on the value of 0 for \( t < 0 \) and 1 for \( t \geq 0 \). This is what the commands would look like

\[
\begin{align*}
\text{f1[t_] := \text{Piecewise}[[\{1, 0 \leq t < \pi\}, \{-1, \pi \leq t < 2\pi\}]]} & \quad \text{or} \quad \text{f2[t_] := \text{UnitStep}[t] - 2 \text{UnitStep}[t - \pi]} \\
\end{align*}
\]

Once you have defined the function, you can compute the Fourier coefficients using integration. Below are input commands for generating the Fourier coefficients for a function \( f[t] \) and creating the series. **You must input a value of \( n \) at the start**, indicating how many terms you want to generate.

\[
n = \text{a0 = 1/(2 \pi)}; \\
\text{a0 = \text{Integrate}[[f[t], \{t, 0, 2\pi\}]] // \text{Simplify}} \\
\text{sinc = Table[1/\pi \text{Integrate}[[f[t] \text{Sin}[k t], \{t, 0, 2\pi\}], \{k, 1, n\}]]} \\
\text{cosc = Table[1/\pi \text{Integrate}[[f[t] \text{Cos}[k t], \{t, 0, 2\pi\}], \{k, 1, n\}]]} \\
\text{fs[t_] = a0 + \text{cosc}.\text{Table}[\text{Cos}[k t], \{k, 1, n\}] + \text{sinc}.\text{Table}[\text{Sin}[k t], \{k, 1, n\}]} \\
\]

Copyright © 2016 Pearson Education, Inc.
This set of commands will output the coefficients and create the Fourier series, naming it \( fs[t] \) and you can plot this function (in red) along with your original function (in blue) with the command:

\[
\text{Plot}[[fs[t], f[t]], \{t, 0, 2\pi\}, \text{PlotStyle} \to \{\text{Red, Blue}\}]
\]

Mathematica also has built-in functions that generate the Fourier series for functions. \texttt{FourierSeries}, \texttt{FourierSinSeries}, \texttt{FourierCosSeries}, \texttt{FourierCoefficients}, etc.

\section*{SECTION 7.1 Orthogonal Diagonalization}

Recall the \texttt{Eigensystem[A]} command as it was used in the discussion for Section 5.3. It generated the eigenvalues and corresponding eigenvectors for a matrix \( A \). That allowed us to define a matrix \( P \) whose columns were the eigenvectors of \( A \), and a diagonal matrix \( mD \) whose entries were the eigenvalues of \( A \), such that \( A = P.mD.P^{-1} \). The columns of \( P \) are orthogonal, but we must normalize them now. We have done this before, but we will introduce a new command \texttt{Normalize}. We can apply this command to a vector to turn it into a unit vector. We will demonstrate this using the matrix in Exercise 21.

\[
A = \begin{bmatrix} 4 & 3 & 1 & 1 \\ 3 & 4 & 1 & 1 \\ 1 & 1 & 3 & 1 \end{bmatrix}
\]
\[
\{\text{evals, evvecs} = \text{Eigensystem}[A]//\text{N};
\text{evecsn=}\text{Table}[\text{Normalize}[\text{evvecs}[[i]]], \{i, 1, 3\}];
\text{MatrixForm}[P = \text{Transpose}[\text{evecsn}]]
\text{MatrixForm}[mD = \text{DiagonalMatrix}[\text{evals}];
\text{MatrixForm}[P.mD.\text{Inverse}[P]]
\]

The output gives

\[
P = \begin{bmatrix} 0.56690 & -0.42032 & -0.707107 & -0.044290 \\ 0.56690 & -0.42032 & 0.707107 & -0.044290 \\ 0.48656 & 0.70990 & 0. & -0.509218 \\ 0.34715 & 0.37777 & 0. & 0.858355 \end{bmatrix},
\]
\[
mD = \begin{bmatrix} 8.4707 & 0. & 0. & 0. \\ 0. & 4.4123 & 0. & 0. \\ 0. & 0. & 1. & 0. \\ 0. & 0. & 0. & -0.8829 \end{bmatrix},
\]

\[
P.mD.\text{Inverse}[P] = \begin{bmatrix} 4 & 3 & 1 & 1 \\ 3 & 4 & 1 & 1 \\ 1 & 1 & 3 & 1 \end{bmatrix},
\]

verifying that we have the orthogonal diagonalization of \( A \).

\section*{SECTION 7.4 The Singular Value Decomposition}

The built-in Mathematica command \texttt{SingularValueList[A]} outputs the list of singular values of the matrix \( A \), and \texttt{SingularValueDecomposition[A]}, outputs the three matrices discussed in Theorem 10 in Section 7.4. Unlike in the \texttt{Eigensystem[A]} command, the matrix \( A \) does not have to be a square matrix in order to apply these commands.

The \texttt{SingularValueList} command was discussed in the discussion for Section 2.3 when indicating how to find the Condition number for a matrix. Since this command outputs the singular values in decreasing order, the commands, \texttt{sv=\text{SingularValueList}[A]//\text{N} and sv[[1]]/sv[[-1]]} output the list of singular values and the condition number of the matrix \( A \). Numerical computation is often needed for simplification.

The following series of input commands can lead you through the steps in Example 4.

\[
A = \begin{bmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{bmatrix};
\text{sv=}\text{SingularValueList}[A]
\text{\{evals, evvecs\}=\text{Eigensystem}[\text{Transpose}[A].A]}
\]

Before the \( \Sigma \) (sigma) matrix can be determined, the singular values are needed. There is only one here.

\[
v=\text{Transpose}[[\text{Normalize}[\text{evvecs}[[1]]]], \text{Normalize}[\text{evvecs}[[2]]]]];
\text{sigma=}\{\text{sv[[1]]},0,0,0\}
\text{u1=}1/\text{sv[[1]]} A.v[[1]]
\]
Solve[u1.{a,b,c}==0,{a,b,c}]

The output of the Solve command is \{c \rightarrow -(a/2)+b\} and from this, you can get two vectors that are perpendicular to u1. If you let a = 2 and b = 0, you get c = -1. If a = 0 and b = 1, then c = 1. You also need to select u3 so it is perpendicular to u2.

\[\begin{align*}
u2 &= \text{Normalize}[\{2,0,-1\}] \\
u3 &= \{0, 1, 1\}; \\
u3 &= \text{Normalize}[u33-u33.u2 u2] \\
u &= \text{Transpose}[\{u1,u2,u3\}] \\
u\cdot\sigma\cdot\text{Transpose}[v]
\end{align*}\]

and the output for the last input will verify that the product of these three matrices is A. The matrices for u and v might vary slightly from those in the text in terms of the ordering of the rows of u or factors of -1 in certain rows, but that is to be expected. If you execute the command

\[\text{SingularValueDecomposition}[A]\]

the output will be very similar to the matrices found in Example 4 in your text.

In Examples 7 and 8, mention is made of a pseudoinverse. For a square matrix, the \textit{Mathematica} command \texttt{PseudoInverse} gives the Moore–Penrose inverse. \texttt{PseudoInverse[A]} also works on rectangular matrices. For a system of equations, \(Ax = b\), \texttt{PseudoInverse[A].b} gives the least squares solution of the system.

**SECTION 7.5 Computing Principal Components**

The \textit{Mathematica} command \texttt{Mean[v]} outputs the mean of the components of the vector \(v\). To generalize our code, we will also use the command \texttt{Length[v]} that determines the length of a list. Consider the data given in Exercise 2. Use the following set of commands to turn the data \(X\) into mean-deviation form, and then find its covariance matrix and principal components as requested in Exercise 4. \textit{Mathematica} also has a \texttt{PrincipalComponents[mat]} command that transforms elements of a matrix into unscaled principal components.

\[\begin{align*}
\text{mat} &= \{(1,5,2,6,7,3),\{3,11,6,8,15,11\}\}; \\
\text{meanlist} &= \text{Table}[\text{Mean}[\text{mat}[[i]],\{i, 1, 2\}] \\
\text{Bmat} &= \text{Transpose}[\text{Table}[\text{Transpose}[\text{mat}[[i]]]-\text{meanlist},\{i, 1, 6\}]] \\
\text{Smat} &= \text{Bmat.\text{Transpose}[\text{Bmat}]} / (\text{Length[\text{mat}[[1]]]}-1) \\
\text{MatrixForm}[[\text{evals, evecs}]=\text{Eigensystem[Smat//N]]}
\end{align*}\]

The output will show the mean list, the B matrix, the S covariance matrix, and the eigensystem for the S matrix. From the eigensystem result, the answer for Exercise 8 can be determine, using the eigenvector for the principal eigenvalue to identify the components of \(y_1\) and computing the ratio of the principal eigenvalue to the sum of the two eigenvalues to determine how much of the variance can be explained by \(y_1\). See the Numerical Note at the end of Section 7.5 of the text as it explains the value in using \texttt{SingularValueDecomposition} matrices to compute principal components of the data for data sets.

**SECTION 8.3 Convex Combinations**

For Exercises 24-25, see the discussion of the \textit{Mathematica} command \texttt{BezierCurve} under Section 8.6.
SECTION 8.5  Polytopes

For finding a vector which minimizes a quantity subject to linear constraints, *Mathematica* contains the built-in command `LinearProgramming`. Even a method of solution can be specified, such as “InteriorPoint”, “Simplex”, or “RevisedSimplex”. Suppose we wished to minimize the function $2x + y$ over the convex region described in Exercise 8 of your text. The commands would be as follows:

```mathematica
A = {{2, 1}, {1, 1}, {1, 2}};
b = {8, 6, 7};
LinearProgramming[{{2, 1}, A, b]
```

The output is the optimal solution to the system.

```
Out[3] = {2, 4}
```

In earlier versions of *Mathematica*, many of the linear algebra commands were available only in special packages that had to be loaded before executing the commands. That is no longer the case, but there are still packages for special purpose operations in *Mathematica*. You can see a list of those packages by selecting Function Navigator on the Help menu and then scrolling down to the Add-Ons & Packages. One package of interest that pertains to Section 8.5 is called “Polytopes”. This package contains functions that give geometrical characteristics of regular polygons. Polygons are identified by name, such as, Digon, Pentagon, Hexagon, Heptagon, Octagon, Nonagon, Decagon, Undecagon, Dodecagon, in function arguments and in results.

The package is loaded using the command `Needs["Polytopes"]`. The symbol ` at the end of the word Polytopes can be found on the upper left side of your keyboard to the left of 1. The `Needs` command has no output, but once it is loaded you can call on commands such as `Vertices` which gives the coordinates of the vertices of a polytope and those points can be plotted. Other commands are `NumberOfVertices`, `NumberOfEdges`, `NumberOfFaces`, `Faces`, `InscribedRadius`, `CircumscribedRadius`. Following is an example of using this package. The `AppendTo` statement is needed because it inserts the first vertex to the end of the list to close the polygon.

```mathematica
Needs["Polytopes"]
vdec = Vertices[Decagon] // N
AppendTo[vdec, vdec[[1]]]
ListPlot[vdec, AspectRatio -> 1, Joined -> True]
```

The option `AspectRatio -> 1` plots the horizontal and vertical points in a 1:1 ratio. The default ratio is the reciprocal of the Golden Ratio. The option `Joined -> True` will connect the points with lines.

SECTION 8.6  Curves and Surfaces

*Mathematica* supports state-of-the-art splines for use both in graphics and computational applications. It allows not just cubic splines, but splines of any degree – for curves, surfaces, and in general manifolds of any dimension. It can represent splines as explicit piecewise symbolic functions. Some of the built-in commands in *Mathematica* that pertain to this section include `BezierCurve` and `BezierFunction` that generate curves or surfaces of higher dimension manifolds and functions for those curves or surfaces. `BezierCurve` by default represents a composite cubic, but the `SplineDegree` option allows one to vary the degree of the polynomials for the curves. As was stated in Exercise 25 from Section 8.3, a Bezier curve of degree $k$ is determined by $k+1$ control points.

*Mathematica* commands `BSplineCurve` and `BSplineSurface` are extensions of Bezier curves and surfaces. A B-spline curve of degree $m$ with $n$ control points consist of $n - m$ Bezier curve segments. These segments all have $C^2$ continuity at the join points. Some Exercises in Section 8.6 compare a B-spline with a Bezier curve. The differences can be demonstrated using the following sets of commands. Here, we
use the **Graphics** command to display Bezier (thin curve) and B-spline (thick curve) curves of degree two, together with the control points and segments connecting the control points.

```mathematica
pts = {{1,1}, {2,3}, {3,-1}, {4,1}, {5,0}};
g1=Graphics[{BezierCurve[pts, SplineDegree \[Rule] 2],
    Dashed, Line[pts], PointSize[.03], Point[pts]}];
g2=Graphics[{Thick, BSplineCurve[pts, SplineDegree \[Rule] 2],
    Dashed, Line[pts], PointSize[.03], Point[pts]}];
Show[g1, g2]
```

If you remove the SplineDegree → 2 specification and re-execute the commands, you would get cubic splines. By default, if **BezierCurve[pts]** is applied to a set of five points, it will find a cubic with which to approximate the first four points, then uses a straight line between the fourth and fifth points.
INDEX OF MATHEMATICA COMMANDS

+ (matrix addition), MM9
- (matrix subtraction), MM9
:= (delayed equals), MM9, MM15, MM20
% (last output), MM11-12
_ (underscore), MM8-9, MM14-15, MM20
^ (power), MM14-15
. (matrix multiplication), MM7 and following
. (inner product), MM22 and following
/(division), MM11, MM20, MM24
//(postfix command), MM2 and following
/. (evaluated at), MM18
;(semicolon), MM5 and following
== (equation equal), MM15 and following
= (set equal), MM2 and following
[[ ]] (double brackets), MM3 and following
? or ?? (question), MM10
Abs, MM21
Append, MM7 and following
AppendTo, MM27
BezierCurve, MM27-28
BSplineCurve, MM27-28
CharacteristicPolynomial, MM18
Clear, MM5, MM8, MM20
CoFactor, MM14
ConstantArray, MM8, MM10, MM15
Cos, MM14
Do, MM17, MM21
Det, MM14
DiagonalMatrix, MM10
Eigensystem, MM17 and following
Eigenvalues, MM17, MM21
Eigenvectors, MM17, MM21
Exp, MM23
FourierSeries, MM25
Graphics, MM28
Hilbert, MM11
I (imaginary number i), MM19
IdentityMatrix, MM10, MM12
Im, MM19, MM21
Integrate, MM24
Inverse, MM11, MM13
Join, MM10, MM23
Length, MM26

LinearProgramming, MM27
LinearSolve, MM12, MM21-23
ListPlot, MM20, MM27
Log, MM23=24
LUDecomposition, MM11
MatrixForm, MM3 and following
MatrixRank, MM13, MM15
MatrixPower, MM9, MM16, MM21
Max, MM21
Mean, MM26
Minors, MM13
N, MM10
Norm, MM22
Normalize, MM25
NullSpace, MM13, MM15
Orthogonalize, MM22
Piecewise, MM24
Plot, MM15 and following
PlotStyle, MM18, MM20
PrincipalComponents, MM26
Projection, MM22
PseudoInverse, MM26
QRDecomposition, MM23
Random, MM17
RandomInteger, MM10-11
Re, MM19, MM21
RowReduce, MM7, MM13, MM15-16, MM23
Show, MM18, MM20, MM28
Simplify, MM19, MM24
Sin, MM23-24
SingularValueDecomposition, MM25-26
SingularValueList, MM10-11, MM25
Sqrt, MM22
Solve, MM16-17
Table, MM9 and following
TableForm, MM9
Transpose, MM6 and following
UnitStep, MM24

Special Packages
“Combinatorica”, MM14
“Polytopes”, MM26
Notes for the TI-84 Family and the TI-89

VARIOUS MODELS FROM THE TI-84 FAMILY

The TI-84 Family of calculators include the TI-83+, the TI-84+ and the TI-84+C with full color display. Throughout this study guide, the differences in the functionality of the various models will be noted.

GETTING STARTED WITH A TI-83+ CALCULATOR

Entering Matrices: There are several ways to enter matrices in the TI-83+. One method consists of entering the matrix directly as an expression, while another method is through the use of the matrix editor found in the 2 > EDIT menu. Below are examples that illustrate the creation of matrices and vectors using these methods. On the TI-83+, a (column) vector is simply an \( n \times 1 \) column matrix. For additional information, consult the TI-83+ Guidebook.

**Direct Entry:** Begin a matrix with a square bracket, and each row of the matrix with a matching pair of square brackets; separate the entries in the rows with commas. Figure 1 below shows how to enter the matrix \[
\begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{bmatrix}
\] directly and assign it the name \([A]\) from the 2 > NAMES menu. The symbol \( \star \) appears after pressing the \( \star \) key.

**The Matrix Editor:** Press 2 > EDIT. The editor allows you to create a matrix by specifying a variable name, a size—row size and column size—and the entries of the matrix. After scrolling over to EDIT, a screen such as Figure 2 will appear. Select a name by pressing \( \star \) on a particular matrix, input rows, columns, and entries. Your screen should be similar to Figure 3. To view the matrix from the home screen, press 2 \( \lambda \) to return home, and then choose 2 > and select the matrix. You can create vectors in a similar fashion. Simply enter vectors in as \( n \times 1 \) column matrices.

![Figure 1: \([A]\) entered directly](image1)
![Figure 2: Matrix, edit menu](image2)
![Figure 3: Matrix editor](image3)
GETTING STARTED WITH A TI-84+ CALCULATOR

**Entering Matrices:** There are several ways to enter matrices into the TI-84+. The methods described earlier for the TI-83+ will also work for the TI-84+. However, one additional method can be used on the TI-84+ through the use of the tab keys.

**Tab Key Entry:** Press 10 to access the Tab Key matrix menu. A screen such as Figure 4 will appear. Use the arrow keys to select the rows and columns of your matrix. When both row and column size have been selected, place the cursor on OK and press ÷. This action will place a matrix onto the home screen, allowing you to input values. Once the matrix is filled, use the arrows to place the cursor outside of the matrix and assign it the name [A] from the 2 > NAMES menu. The symbol ! appears after pressing the • key. See Figure 5.

![Figure 4: Matrix, tab menu](image1)

![Figure 5: [A] entered through tab menu](image2)

GETTING STARTED WITH A TI-89 CALCULATOR

**Entering Matrices:** There are several ways to enter matrices into the TI-89. One method consists of entering the matrix directly as an expression, another method is through the use of the matrix editor found in the Data/Matrix Editor in the O menu. Below are examples that illustrate the creation of matrices using these methods. On the TI-89, a (column) vector is simply an n×1 column matrix. For additional information, consult the TI-89 Guidebook.

**Direct Entry:** Begin a matrix with a square bracket, and each row of the matrix with a matching pair of square brackets; separate the entries in the rows with commas. Figure 6 below shows how you can enter the matrix \[
\begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{bmatrix}
\] directly and assign to it the name aa—the symbol ! appears after pressing the • key, press ÷ to display the matrix aa on the screen. Matrices can be denoted by nearly any combination of letters and numbers, but double lower-case letters will be used to denote matrices in this text. Entries can either be real or complex. To input complex entries, simply input them in with the form \(a+bi\).

![Figure 6: aa entered directly](image3)
The Matrix Editor: Press 0, scroll down to the Data/Matrix Editor and press +. See Figure 7. You will be asked whether you want to work on the current data item, open a data item, or make a new data item. Choose 3:New to construct a new matrix. See Figure 8.

Use the right arrow to select either Data, Matrix, or List. Choose 2:Matrix by pressing +. Scroll down to Variable, type in the name of the matrix, input number of rows and columns in the appropriate places, and press + twice. See Figure 9. If the variable name is not being used for anything else, the TI-89 will allow you to start inputting entries directly into the matrix. See Figure 10.

Once the matrix has been input, go home by pressing the \( \triangleright \) button. To view the matrix, type in the name that you assigned to the matrix and press +. You can create vectors in a similar fashion. Simply enter vectors in as \( n \times 1 \) column matrices. If you want to change a matrix, select 2:Open instead of 3:New, and open an existing matrix.

STUDY GUIDE NOTES

Data and Program Files

To use your TI calculator with the text, you will need to download to a computer the appropriate TI-data from the website www.pearsonhighered.com/lay. Also available on this site is a selection of supplemental programs that will be beneficial in your study of linear algebra. The files are compressed and must be decompressed by a ZIP or Stuffit program (available from the website) into a single new folder on the computer. Then, transfer the files to your calculator. (Consult your TI guidebook for details.)

TI-84 Family The data for each chapter of the text are stored in separate programs with names such as CHAPTER1 or CHAPTER4. The program that includes the supplemental programs is called ALINEAR. The files for each chapter are very large, so the chapters that are not in use must be either archived or deleted from memory. If other large files are stored into the
calculator’s RAM, they may have to be archived as well. To put data files in archive memory, press 2 \( \triangleleft 5 \) and then choose the file you want to archive. Pressing \( \triangledown \) will send the file to archive memory where it will no longer be accessible. To unarchive a file, follow the same steps with the UnArchive function.

When downloaded to the calculator, these programs will appear in the PRGM menu. To run a program for a particular chapter, press the 8 button, scroll down to the chapter in question and press \( \triangledown \). The program will ask you for a section number and then an exercise number. It will then display where the data for the exercise is located; in matrices ranging from \([A]\) to \([F]\). To run the program ALINEAR, follow the same procedures and select the desired program and follow the directions for what to input.

**TI-89** The data for each chapter of the text are stored in separate programs with names such as CHAPTER1 or CHAPTER4. The program that includes the supplemental programs is called ALINEAR. The files for each chapter are very large, so the chapters that are not in use may have to be either archived or deleted from memory. If other large files are stored into the calculator’s RAM, they may have to be archived as well. However, there is space for every file on a calculator with average memory use. To put data files in archive memory, press 2 \( \triangledown \), choose the file you want to archive, and then press \( \triangledown \). Pressing \( \triangledown \) will send the file to archive memory where it will no longer be accessible. To unarchive a file, follow the same steps with the UnArchive function.

When downloaded to the calculator, these programs will appear in the \( \triangledown \) menu. To run a program for a particular chapter, go to the \( \triangledown \) menu, scroll down to the chapter in question and press \( \triangledown \). The program will ask you for a section number and then an exercise number. It will then display where the data for the exercise is located; in matrices ranging from \( aa \) to \( ff \). To run the program ALINEAR, follow the same procedures and select the desired program and follow the directions for what to input.

**SECTION 1.1 Row Operations**

**TI-84 Family** In this exercise set, the data for each exercise are stored in a matrix \([A]\). Row operations on \([A]\) are performed by the following commands:

\[
\text{rowswap}([A],s,r) \quad \text{Interchanges row } r \text{ and } s \text{ of } [A] \\
*mrow(m,[A],s) \quad \text{Multiplies row } s \text{ of } [A] \text{ by a nonzero scalar } m \\
*row+(m,[A],s,r) \quad \text{Replaces row } r \text{ of } [A] \text{ by } (\text{row } r) + m*(\text{row } s)
\]

These operations are found by pressing 2 \( \triangledown \) MATH and scrolling down to the bottom, Figures 12 and 13.

![Figure 12: Matrix, math menu](image1.png) ![Figure 13: Matrix, math menu extended](image2.png)

If you enter one of these commands, say, \text{rowswap}([A],1,3), then the new matrix, produced from \([A]\), is stored in the ANS (for “answer”). If instead you enter, \text{rowswap}([A],1,3)! [B], the answer is stored in a new matrix [B].

Copyright © 2016 Pearson Education, Inc.
The advantage of giving a new name to each new matrix is that you can easily go back a step if you don’t like what you just did to a matrix. If, instead you enter \( \text{rowswap}([A],1,3)! [A] \), then the result is placed back in \([A]\) and the old \([A]\) is lost. Of course the reverse operation \( \text{rowswap}([A],1,3)! [A] \), will bring back the old \([A]\).

Note: For the simple problems in this section and the next, the multiple \( m \) you need in the command \( \ast\text{row+(m,}[A],s,r) \) will usually be a small integer or fraction that you can compute in your head. In general, \( m \) may not be so easy to compute mentally. The next two paragraphs describe how to handle such a case.

The entry in row \( r \) and column \( c \) of a matrix \([A]\) is denoted by \([A](r,c)\). If the number stored in \([A](r,c)\) is displayed with a decimal point, then the displayed values may be accurate to only about five digits. In this case, use the symbol \([A](r,c)\) instead of the displayed value in calculations.

For instance, if you want to use the entry (pivot) \([A](s,c)\) to change \([A](r,c)\) to 0, enter the commands
\[
-[A](r,c)/[A](s,c)! m
\]
Multiple of row \( s \) to be added to row \( r \)
\[
\ast\text{row+}(m, [A], s, r)! [A]
\]
Adds \( m \) times row \( s \) to row \( r \)
or you can just use the command
\[
\ast\text{row+}(-[A](r,c)/[A](s,c), [A], s, r)! [A]
\]

**TI-89** In this exercise set, the data for each exercise are stored in a matrix \( aa \). Row operations on \( aa \) are performed by the following commands:

- \( \text{mRowAdd}(m, aa, s, r) \): Replaces row \( r \) of \( aa \) by \((row r) + m \times (row s)\)
- \( \text{rowSwap}(aa, r, s) \): Interchanges rows \( r \) and \( s \) of \( aa \)
- \( \text{mRow}(m, aa, r) \): Multiplies row \( r \) of \( aa \) by a nonzero scalar \( m \)

These operations are found by pressing 21, scrolling down to, or pressing 4:Matrix as seen in Figure 16. Scrolling down again to 3:Row ops in Figure 17, and pressing the right arrow shows these four options.

To use the row operations efficiently it is convenient to store matrices in memory. The name of any matrix in your calculator memory can be inserted in place of \( aa \); the letters \( r \), \( m \), and \( s \) stand for numbers you choose. Press + after each command.

If you enter one of these commands, say, \( \text{rowSwap}(aa,1,3) \) then the new matrix, produced from \( aa \), is stored in the matrix \text{ans}(1)\ (for “answer”). If instead you enter \( \text{rowSwap}(aa,1,3)!bb \), then the answer is stored in a new matrix \( bb \). If the new operation is \( \text{mRowAdd}(5, bb, 1, 2)!cc \), then the result of changing \( bb \) is placed in \( cc \), and so on.

The advantage of giving a new name to each new matrix is that you can easily go back a step if you don’t like what you just did to a matrix. If, instead you enter \( \text{mRowAdd}(5, aa, 1, 2)!aa \),
then the result is placed back in aa and the old aa is lost. Of course the reverse operation mRowAdd(-5, aa, 1, 2)\!aa, will bring back the old aa.

Note: For the simple problems in this section and the next, the multiple m you need in the command mRowAdd(m, aa, s, r) will usually be a small integer or fraction that you can compute in your head. In general, m may not be so easy to compute mentally. The next two paragraphs describe how to handle such a case.

The entry in row r and column c of a matrix aa is denoted by aa[r,c]. If the number stored in aa[r,c] is displayed with a decimal point, then the displayed values may be accurate to only about five digits. In this case, use the symbol aa[r,c] instead of the displayed value in calculations.

For instance, if you want to use the pivot entry aa[s,c] to change aa[r,c] to 0, enter the commands
\[-aa[r,c]/aa[s,c]\!m\]
mRowAdd(m, aa, s, r)!aa  Adds m times row s to row r
or you can just use the command
\[-aa[r,c]/aa[s,c],aa[s,c],aa,s,r]!aa

SECTION 1.3 Constructing a Matrix

TI-84 Family The augment( operator concatenates two matrices or a matrix and a vector. For instance, if [A] is a matrix and [B] is a vector b, then the augmented matrix for the equation \( Ax = b \) is given by augment([A], [B]). To construct this, press 2 > MATH and select augment(. Fill in [A] and [B] separated by a comma and press +. Matrices must have the same number of rows in order to augment them together. Given a vector equation such as \( x_1a_1 + x_2a_2 + x_3a_3 = b \), you might store \( a_1 \) in [A], \( a_2 \) in [B], \( a_3 \) in [C], and b in [D]. Then the augmented matrix for this system is created by the command
\[ \text{augment(augment(augment([A], [B]), [C]), [D])}!E \]

Excercises 11-14, 25-28, and 21 can be solved using the commands rowswap(, *row(, and row+, described in notes for Section 1.1.

TI-89 The augment( operator concatenates two matrices or a matrix and a vector. For instance, if aa is a matrix and bb is a vector b, then the augmented matrix for the equation \( Ax = b \) is given by augment(aa, bb). It can be found by pressing 2 1, scrolling down to, or pressing 4:Matrix. Then scroll down to 7:augment(. Fill in aa and bb separated by a comma and press +. Matrices must have the same number of rows in order to augment them together. Given a vector equation such as \( x_1a_1 + x_2a_2 + x_3a_3 = b \), you might store \( a_1 \) in aa, \( a_2 \) in bb, \( a_3 \) in cc, and b in dd. Then the augmented matrix for this system is created by the command
\[ \text{augment(augment(augment(aa, bb), cc), dd)!ee} \]

Excercises 11-14, 25-28, and 21 can be solved using the commands mRowAdd(, rowSwap(, mRow+, described in notes for Section 1.1.

SECTION 1.4 Gauss and BGauss

TI-84 Family To solve \( Ax = b \), row reduce the matrix \( C = [A \ b] \). Matrix [C], or any other for that matter, can be constructed by the command: augment([A], [B])!C. To speed up row reduction of [C], the program GAUSS (in the ALINEAR program) will use the leading entry in row r of [C] as a
pivot, and use row replacements to create zeros in the pivot column below this pivot entry. The output of GAUSS is stored in the matrix [J]. If you wish, you can assign [J] to some other variable, such as [A] (the first matrix in the list).

For the backward phase of row reduction, use the BGAUSS program, which selects the leading entry in row r as the pivot, and creates zeros in the column above the pivot. Use *row( to create leading 1's in the pivot positions. The output of BGAUSS is stored in the matrix [J]. If you wish, you can assign [J] to some other variable, such as [A] (the first matrix in the list).

To run a program, see the notes on Data and Program Files at the beginning of the study guide notes. Both GAUSS and BGAUSS will prompt the user for the matrix and the row number.

**SECTION 1.5 Zero Matrices**

**TI-89** To solve $A\mathbf{x} = \mathbf{b}$, row reduce the matrix $C = [A \ b]$. Matrix cc, or any other for that matter, can be constructed by the command: `augment(aa,bb)!cc`.

To speed up row reduction of cc, the program GAUSS (in the ALINEAR program) will use the leading entry in row r as a pivot, and use row replacements to create zeros in the pivot column below this pivot entry. The output of GAUSS is stored in the matrix xx.

For the backward phase of row reduction, use the BGAUSS program, which selects the leading entry in row r as the pivot, and creates zeros in the column above the pivot. Use `mRow( to create leading 1's in the pivot positions. The output of BGAUSS is stored in the matrix xx.

To run a program, see the notes on Data and Program Files at the beginning of the study guide notes. Both GAUSS and BGAUSS will prompt the user for the matrix and the row number.

**SECTION 1.6 Rational Format**

**TI-84 Family** Chemical equation-balance problems are studied best using exact or symbolic arithmetic, because the balance variables must be whole numbers (with no round-off allowed). To convert floating-point numbers shown in matrices to rational answers, the command `m++` will...
give the last answer in fractions, if possible. This function will convert a number or a matrix into rational form. Once you find a rational solution of a chemical equation-balance problem, you can multiply the entries in the solution vector by a suitable integer to produce a solution that involves only whole numbers.

**TI-89**  
Chemical equation-balance problems are studied best using exact or symbolic arithmetic, because the balance variables must be whole numbers (with no round-off allowed). To convert floating-point numbers shown in matrices to rational answers, try setting the MODE, PAGE 2, to exact. This will give the answers in fractions, if possible. This setting will convert a number or a matrix into rational form. Once you find a rational solution of a chemical equation-balance problem, you can multiply the entries in the solution vector by a suitable integer to produce a solution that involves only whole numbers.

**SECTION 1.10 Generating a Sequence**

**TI-84 Family** The data program for exercises 9-13 for Section 1.10 stores vectors into the matrix \([B]\). Use the command

\([B]*x +\)

to generate the next value in the sequence. To continue generating the sequence press \(\times +\) repeated times. This will compute the answer stored in memory times \(x\).

**TI-89** The data program for exercises 9-13 for Section 1.10 stores vectors into the matrix \(bb\). Use the command

\(bb*x +\)

to generate the next value in the sequence. To continue generating the sequence press \(\times +\) repeated times. This will compute the answer stored in memory times \(x\).

**SECTION 2.1 Matrix Notation and Operations**

**TI-84 Family** To create a matrix, begin with a square bracket, enter the data row-by-row, with a comma between entries, each row of the matrix must begin and end with square brackets. For instance, the command

\([[1,2,3][4,5,6]]*[A]\)

creates a 2×3 matrix \([A]\). If \([A]\) is \(m×n\), then \(\text{dim}([A])\) (found in the 2 > MATH menu) is the list \(\{m, n\}\). The \((i,j)\)-entry in the matrix \([A]\) is \([A](i,j)\). To extract either the \(j^{th}\) row or the \(j^{th}\) column from a matrix, use program JROW or JCOL respectively. Each program will prompt for a matrix, and row or column number, and display the item in question.

The calculators in the TI-84 Family use the +, −, and * keys to denote matrix addition, subtraction and multiplication, respectively. (Multiplication shows on the screen and in these notes as *.) If \([A]\) is square and \(k\) is a positive integer, \([A]\ ^k\) denotes the \(k^{th}\) power of \([A]\) \(([A]^k, q\) is equivalent to \([A]\ ^2\)). The transpose of \([A]\) is \([A]^\top\) (the \(\top\) operator is in the 2 > MATH menu).

If \(A\) and \(B\) are two column vectors of the same size, their inner product (or dot product) can be computed by \(A^\top*B\) or \(B^\top*A\). In this section you can experiment with random matrices. The
command \texttt{randM(} returns a random matrix with integer entries in the interval [-9,9]. Go to the 2 > \texttt{MATH} menu to activate the command; it requires that you enter the two arguments—row size and column size—then close the parenthesis and press \( \). (To save the random matrix into memory press the \( \) key and enter the name of the matrix before you press \( \).)

\textbf{TI-89} To create a matrix, begin with a square bracket, enter the data row-by-row, with a comma between entries, each row of the matrix must begin and end with square brackets. For instance, the command

\[
\begin{bmatrix}
[1,2,3] \\
[4,5,6]
\end{bmatrix}
\]

creates a 2\( \times \)3 matrix aa. If aa is \( m \times n \), then \texttt{dim(aa)} (found by pressing 2 \( \right\), and scrolling down to, or pressing 4:Matrix, and scrolling down to \texttt{I:dimensions} and pressing the right arrow.) is the list \{m,n\}. The \([i,j]\)-entry in the matrix aa is aa\([i,j]\]. aa\([i]\) is the \( i^{th} \) row of the matrix aa, \textit{given as a vector}. To extract the \( j^{th} \) column of aa, use the program \texttt{COL} from \texttt{ALINEAR}. The program will prompt for a matrix, and column number, it will return the \( j^{th} \) column as a vector.

The TI-89 uses the \( \leftrightarrow \), \(|\), and \( \pi \) keys to denote matrix addition, subtraction and multiplication, respectively. (Multiplication shows on the screen and in these notes as \( \ast \).) If aa is square and \( k \) is a positive integer, aa \( Zk \) denotes the \( k^{th} \) power of aa. The transpose of aa is aa\( ^{\ast} \) (the \( ^{\ast} \) operator is found by pressing 2 \( \right\), and scrolling down to, or pressing 4:Matrix).

If u and v are two vectors of the same size, then \texttt{dotP(u,v)} is their inner product (the \texttt{dotP(} operator is found by pressing 2 \( \right\), and scrolling down to, or pressing 4:Matrix). Then scroll down to \texttt{L:Vector ops} and press the right arrow). The vectors can either be input as \( n \times 1 \) matrices or \( 1 \times n \) matrices when computing dot products.

In this section you can experiment with random matrices. The command \texttt{randMat(} returns a random matrix with integer entries in the interval \([-9,9]\). Press 2 \( \right\), and scrolling down to, or pressing 4:Matrix, and scroll down to \texttt{E:randMat(} It requires that you enter the two arguments—row size and column size—then close the parenthesis and press \( \). (To save the random matrix into memory press the \( \) key and enter the name of the matrix before you press \( \)).

\textbf{SECTION 2.2 The Identity Matrix and \( A^{-1} \)}

\textbf{TI-84 Family} The \( n \times n \) identity matrix is \texttt{identity(n)} (the \texttt{identity(} command is in the 2 > \texttt{MATH} menu). If \([A]\) is an \( n \times n \) matrix, then the command

\[
\texttt{augment([A],identity(n))}![B]
\]

creates the augmented matrix \( B = A I \). Use the \texttt{GAUSS} and \texttt{BGAUSS} programs, and the \texttt{*row(} operation to reduce the augmented matrix \( A I \) completely.

There are other commands that can be used to row reduce matrices, invert matrices, and solve equations \( Ax = b \), but they are not discussed here because they will not help you to learn the concepts in this section.

\textbf{TI-89} To get an \( n \times n \) identity matrix press \texttt{identity(n)} (the command is found by pressing 2 \( \right\), and scrolling down to, or pressing 4:Matrix). If aa is an \( n \times n \) matrix, then the command

\[
\texttt{augment(aa,identity(n))}!bb
\]

creates the augmented matrix \( B = A I \). Use the \texttt{GAUSS} and \texttt{BGAUSS} programs, and the \texttt{mRow(} operation to reduce the augmented matrix \( A I \) completely.

Copyright © 2016 Pearson Education, Inc.
There are other commands that can be used to row reduce matrices, invert matrices, and solve equations $Ax = b$, but they are not discussed here because they will not help you to learn the concepts in this section.

SECTION 2.3 Characterizations of Invertible Matrices

**TI-84 Family** Determining whether a specific numerical matrix is invertible is not always a simple matter. A fast and fairly reliable method is to enter the matrix $[A]$ and press $[A]^{-1}$, which computes the inverse of $[A]$. If the matrix in question is not invertible, the calculator will either display “SINGULAR MAT.” or “ERR:INVALID DIM”.

**TI-89** Determining whether a specific numerical matrix is invertible is not always a simple matter. A fast and fairly reliable method is to enter the matrix $aa$ and press $Z \bullet \bullet$, which computes the inverse of $aa$. If the matrix in question is not invertible, the calculator will either display “Singular matrix” or “Error: Dimension”.

SECTION 2.4 Partitioned Matrices

**TI-84 Family** The calculators in the TI-84 family use partitioned matrix notation. For example, if $[A]$, $[B]$, $[C]$, $[D]$, $[E]$, and $[F]$ are matrices of appropriate sizes, then the command

$$
\text{augment(augment(augment([A],[B]),[C]),[E]),[F])} \rightarrow [G]
$$

creates a larger matrix of the form $[G]=
\begin{bmatrix}
A & B & C \\
D & E & F
\end{bmatrix}$.

Once $[G]$ is formed, there is no record of the partition that was used to create $[G]$. For instance, although $[B]$ was the $(1, 2)$-block used to form $[G]$, the number $[G](1,2)$ is the same as the $(1, 2)$-entry of $[A]$.

**TI-89** The TI-89 uses partitioned matrix notation. For example, if $aa$, $bb$, $cc$, $dd$, $ee$, and $ff$ are matrices of appropriate sizes, then the command

$$
\text{augment(augment(augment(aa,bb),cc),[F])} \rightarrow [G]
$$

creates a larger matrix of the form $[G]=
\begin{bmatrix}
A & B & C \\
D & E & F
\end{bmatrix}$.

Once $[G]$ is formed, there is no record of the partition that was used to create $[G]$. For instance, although $[B]$ was the $[1, 2]$-block used to form $[G]$, the number $[G][1,2]$ is the same as the $[1, 2]$-entry of $aa$.

SECTION 2.5 LU Factorization

**TI-84 Family** Row reduction of $A$ using the GAUSS program will produce the intermediate matrices needed for an LU factorization of $A$. You can try this on the matrix in Example 2. The matrices in (5) in the text are produced by running the GAUSS program on the following matrices and rows:

- $[A]$ row 1 Returns a matrix $[J]$ with zeroes below the first pivot
- $[J]$ row 2 Returns a matrix $[J]$ with zeroes below pivots 1 and 2
- $[J]$ row 3 Returns an echelon form, $[J]$
You can copy the information from your screen onto your paper, and divide by the pivot entries to produce $L$ as in the text. For most text exercises, the pivots are integers and so are displayed accurately.

The $\text{ALINEAR}$ program $\text{LU}$ produces $U$ and an $n \times 2n$ matrix $J$ for some square matrices, but it does not handle the general case. From $J$, you can perform row swaps to make $L$ and $P$. The left half of the matrix will form $L$ and the right half will form $P$.

**TI-89** Row reduction of $A$ using the $\text{GAUSS}$ program will produce the intermediate matrices needed for an LU factorization of $A$. You can try this on the matrix in Example 2. The matrices in (5) in the text are produced by running the $\text{GAUSS}$ program on the following matrices and rows:

- $\text{aa row 1}$ Returns a matrix $xx$ with zeros below the first pivot
- $\text{xx row 2}$ Returns a matrix $xx$ with zeros below pivots 1 and 2
- $\text{xx row 3}$ Returns an echelon form, $xx$

You can copy the information from your screen onto your paper, and divide by the pivot entries to produce $L$ as in the text. For most text exercises, the pivots are integers and so are displayed accurately.

To access the TI-89 command $\text{LU}$, press $\text{2} \text{\leftarrow MATH}$, and scroll down to, or press $4:\text{Matrix}$. Then scroll down to $\text{B:LU}$. The command $\text{LU aa,11,uu,pp}$ produces a permuted LU factorization for some square matrices $aa$, but it does not handle the general case.

**SECTION 2.8 rref**

**TI-84 Family** The command $\text{rref(}$, found in the $2 > \text{MATH}$ menu, produces the reduced row echelon form of a given matrix. This form gives you enough information to write down a basis for $\text{Col } A$, and the homogeneous equations that describe $\text{Nul } A$. (Don't forget that $A$ is a coefficient matrix, not an augmented matrix.) The command $\text{rref(}$ does not work if the number of columns in $A$ is less than the number of rows in $A$. If this is the case, you can augment $A$ with enough columns of zeros, and work with a square matrix. Use the command as usual, then ignore the columns of zeros when you interpret the output.

**TI-89** The command $\text{rref(}$, found by pressing $2 \text{ \leftarrow MATH}$, scrolling down to, or pressing $4:\text{Matrix}$, and then pressing $4:\text{rref(}$ produces the reduced row echelon form of a given matrix. This form gives you enough information to write down a basis for $\text{Col } A$, and the homogeneous equations that describe $\text{Nul } A$. (Don't forget that $A$ is a coefficient matrix, not an augmented matrix.) The command $\text{rref(}$ does not work if the number of columns in $A$ is less than the number of rows in $A$. If this is the case, you can augment $A$ with enough columns of zeros, and work with a square matrix. Use the command as usual, then ignore the columns of zeros when you interpret the output.

**SECTION 2.9 Rank**

**TI-84 Family** With the calculators in the TI-84 Family, you should use an echelon form of $A$ to determine the rank of $A$. You can row reduce $A$ using $\text{GAUSS}$, or you can use one of the commands $\text{rref(}$ or $\text{ref(}$ from the $2 > \text{MATH}$ menu. The $\text{ref([A])}$ command produces a row echelon form of $[A]$. (Don't forget that $A$ is a coefficient matrix, not an augmented matrix.) Both $\text{rref(}$ and $\text{ref(}$ only work on matrices with at least as many columns as rows. However, you can
augment \( A \) with enough columns of zeros to make a square matrix, if necessary, and then ignore the columns of zeros when you interpret the output. With each of these three methods, roundoff error from an extremely small pivot entry can sometimes produce an incorrect echelon form.

**TI-89** With a TI-89, you should use an echelon form of \( A \) to determine the rank of \( A \). You can row reduce \( A \) using \texttt{GAUSS} , or you can use one of the commands \texttt{rref} or \texttt{ref} found by pressing 2 1, and scrolling down to, or pressing 4:Matrix. The \texttt{ref(aa)} command produces a row echelon form of \( aa \). (Don’t forget that \( A \) is a coefficient matrix, not an augmented matrix.) Both \texttt{rref} and \texttt{ref} only work on matrices with at least as many columns as rows. However, you can augment \( A \) with enough columns of zeros to make a square matrix, if necessary, and then ignore the columns of zeros when you interpret the output. With each of these three methods, roundoff error from an extremely small pivot entry can sometimes produce an incorrect echelon form.

**SECTION 3.2 Computing Determinants**

**TI-84 Family** To compute \( \text{det} A \), use the \texttt{GAUSS} program and \texttt{rowswap(}, repeatedly, to reduce \( A \) to a matrix \( U \), which is an echelon form of \( A \). Keep track of how many times you swap rows. Then, except possibly for a factor of \( \pm 1 \), the determinant is obtained by using the program \texttt{MDIAV} (also found in the \texttt{ALINEAR} program) to extract the diagonal of \( U \), followed by the \texttt{PRDCT} program to compute the product of the diagonal. You can, of course, use the \texttt{det(} selection from the 2 \( > \) \texttt{MATH} menu) to check your work, but the longer sequence of commands helps you to think about the process of computing \( \text{det} A \).

**TI-89** To compute \( \text{det} A \), use the \texttt{GAUSS} program and \texttt{rowSwap(}, repeatedly, to reduce \( A \) to a matrix \( U \), which is an echelon form of \( A \). Keep track of how many times you swap rows. Then, except possibly for a factor of \( \pm 1 \), the determinant is obtained by using the program \texttt{MDIAV} (also found in the \texttt{ALINEAR} program) to extract the diagonal of \( U \), followed by the \texttt{PRDCT} program to compute the product of the diagonal. You can, of course, use the \texttt{det(} command found by pressing 2 1, and scrolling down to, or pressing 4:Matrix, to check your work, but the longer sequence of commands helps you to think about the process of computing \( \text{det} A \).

**SECTION 4.3 ref and rref**

**TI-84 Family** With the calculators in the TI-84 Family, you should use an echelon form of \( A \) to determine the rank of \( A \). You can row reduce \( A \) using \texttt{GAUSS} , or you can use one of the commands \texttt{rref(} or \texttt{ref(} from the 2 \( > \) \texttt{MATH} menu. The \texttt{ref([A])} command produces a row echelon form of \( [A] \). (Don’t forget that \( A \) is a coefficient matrix, not an augmented matrix.) Both \texttt{rref(} and \texttt{ref(} only work on matrices with at least as many columns as rows. However, you can augment \( A \) with enough columns of zeros to make a square matrix, if necessary, and then ignore the columns of zeros when you interpret the output. With each of these three methods, roundoff error from an extremely small pivot entry can sometimes produce an incorrect echelon form.

**TI-89** With a TI-89, you should use an echelon form of \( A \) to determine the rank of \( A \). You can row reduce \( A \) using \texttt{GAUSS} , or you can use one of the commands \texttt{rref(} or \texttt{ref(} found by pressing 2 1, and scrolling down to, or pressing 4:Matrix. The \texttt{ref(aa)} command produces a row echelon form of \( aa \). (Don’t forget that \( A \) is a coefficient matrix, not an augmented matrix.) Both \texttt{rref(} and
ref( only work on matrices with at least as many columns as rows. However, you can augment \( A \) with enough columns of zeros to make a square matrix, if necessary, and then ignore the columns of zeros when you interpret the output. With each of these three methods, roundoff error from an extremely small pivot entry can sometimes produce an incorrect echelon form.

**SECTION 4.4 The Inverse Operator -1**

**TI-84 Family** If the equation \( A \mathbf{x} = \mathbf{b} \) has a unique solution and \( A \) is a square matrix, the calculators in the TI-84 Family will automatically produce \( \mathbf{x} \) if you use the command 

\[ [A]^{-1} \cdot [B] \cdot \mathbf{x} \]

In this section, the equation will probably have the form \( P \mathbf{u} = \mathbf{x} \), with \( \mathbf{u} \) the \( B \)-coordinate vector of \( \mathbf{x} \), and the command will be \([P]^{-1} \cdot [X] \cdot \mathbf{u}\).

**TI-89** If the equation \( A \mathbf{x} = \mathbf{b} \) has a unique solution and \( A \) is a square matrix, the TI-89 will automatically produce \( \mathbf{x} \) if you use the command 

\[ a^{-1} \cdot b \cdot \mathbf{x} \]

In this section, the equation will probably have the form \( P \mathbf{u} = \mathbf{x} \), with \( \mathbf{u} \) the \( B \)-coordinate vector of \( \mathbf{x} \), and the command will be \([P]^{-1} \cdot [X] \cdot \mathbf{u}\).

**SECTION 4.6 Rank and Random Matrices**

**TI-84 Family** With the calculators in the TI-84 Family, you should use an echelon form of \( A \) to determine the rank of \( A \). You can row reduce \( A \) using \texttt{GAUSS}, or you can use one of the commands \texttt{rref()} or \texttt{ref()} from the 2 \( > \) \texttt{MATH} menu.

In this section you can experiment with random matrices. The command \texttt{randM()} returns a random matrix with integer entries in the interval \([-9, 9]\). Go to the 2 \( > \) \texttt{MATH} menu to activate the command; it requires that you enter the two arguments—row size and column size then close the parenthesis and press \( + \). (To save the random matrix into memory press the \( \star \) key and enter the name of the matrix before you press \( + \).)

**TI-89** With a TI-89, you should use an echelon form of \( A \) to determine the rank of \( A \). You can row reduce \( A \) using \texttt{GAUSS}, or you can use one of the commands \texttt{rref()} or \texttt{ref()} found by pressing 2 \( 1 \), and scrolling down to, or pressing 4:Matrix.

In this section you can experiment with random matrices. The command \texttt{randMat()} returns a random matrix with integer entries in the interval \([-9, 9]\). Press 2 \( 1 \), scroll down to, or press 4:Matrix, and finally scroll down to \texttt{E:randMat()} It requires that you enter the two arguments—row size and column size—then close the parenthesis and press \( + \). (To save the random matrix into memory press the \( \star \) key and enter the name of the matrix before you press \( + \).)

**SECTION 4.7 Change of Coordinates Matrix**

**TI-84 Family** The \texttt{rref()} command will completely reduce a matrix of the form \([c_1 \ c_2 \ b_1 \ b_2]\) to the desired form. See the note for Section 1.3 on how to construct this matrix.
**TI-89** The \texttt{rref(} command will completely reduce a matrix of the form $\begin{bmatrix} c_1 & c_2 & b_1 & b_2 \end{bmatrix}$ to the desired form. See the note for Section 1.3 on how to construct this matrix.

**SECTION 4.8 Roots**

**TI-84 Family** To find the roots of a polynomial, use the \texttt{solve(} function. This can be found by pressing 2 N “s” and then scroll down. (While looking in the catalog, the calculator automatically puts the alpha lock on.) The command should look similar to this:

\texttt{solve(polyomial,variable,guess,[lower bound,upper bound])}

For additional help, consult your TI-84+ Guidebook.

**TI-89** To find the roots of a polynomial, use the \texttt{solve(} function found in the F2:Algebra menu. Press + on the correct function, input the equation and the variable, separated by a comma, close the parentheses, and press +. Refer to your TI-89 Guidebook for questions.

**SECTION 4.9 Markov Chains**

**TI-84 Family** The notes for Section 1.10 contain information that is useful for the exercises in this section.

**TI-89** The notes for Section 1.10 contain information that is useful for the exercises in this section.

**SECTION 5.1 Finding Eigenvectors**

**TI-84 Family** The program \texttt{NULB} (in the \texttt{ALINEAR} program) will simplify your homework by automatically producing a basis for an eigenspace. For example, if $A$ is a $3\times3$ matrix with an eigenvalue 7, first input $[A]$, then use the command

\texttt{[A]-7identity(3)[C]}

to produce the matrix $C = A - 7I$. Then run the \texttt{NULB} program; at the prompt for a matrix, enter $[C]$. The output is a matrix $[J]$ whose columns form a basis for the eigenspace of $A$ corresponding to $\lambda = 7$. In general \texttt{identity(k)} (in the 2 > MATH menu) produces the $k\times k$ identity matrix and \texttt{NULB} produces a matrix whose columns form a basis for Nul $C$ (the same basis you would get if you started with \texttt{rref([C])} and calculated by hand).

Remarks:
1. The program \texttt{NULB} uses the command \texttt{rref()} and requires that the number of columns of the input matrix be greater than or equal to the number of rows.
2. If the numbers in the basis matrix $B$ are messy, try the sequence of keystrokes \texttt{m ++} which shows the last answer in fractions, if possible.

**TI-89** The program \texttt{NULB} (in the \texttt{ALINEAR} program) will simplify your homework by automatically producing a basis for an eigenspace. For example, if $aa$ is a $3\times3$ matrix with an eigenvalue 7, first input $aa$, then use the command:

\texttt{aa-7identity(3)[cc]}
to produce the matrix 

\[ C = A - 7I \] 

Then run the \texttt{NULB} program; at the prompt for a matrix, enter \texttt{cc}. The output is a matrix \( xx \) whose columns form a basis for the eigenspace of \( A \) corresponding to \( \lambda = 7 \). In general identity(\( k \)) (found by pressing \( \texttt{2nd} \) \( \texttt{9} \), and scrolling down to, or pressing \( \texttt{4:Matrix} \)) produces the \( k \times k \) identity matrix and \texttt{NULB} produces a matrix whose columns form a basis for \( \text{Nul} \) \( C \) (the same basis you would get if you started with \( \text{rref}(\texttt{cc}) \) and calculated by hand).

Remarks:
1. The program \texttt{NULB} uses the command \( \text{rref}() \) and requires that the number of columns of the input matrix be greater than or equal to the number of rows.
2. If the numbers in the basis matrix \( B \) are messy, try setting the \texttt{MODE, PAGE 2}, to exact.

**SECTION 5.2 CHARA**

**TI-84 Family** You can use the \texttt{CHARA} program to check your answers in Exercises 9–14. Note that if \( A \) is \( n \times n \), running this program produces a vector listing the coefficients of the characteristic polynomial of \( A \), in order of decreasing powers of \( \lambda \), beginning with \( \lambda^n \). If the polynomial is of odd degree, the coefficients are multiplied by \(-1\), to make \(+1\) the coefficient of \( \lambda^n \) this corresponds to finding the determinant of \( \lambda I - A \). (\texttt{CHARA} works for matrices up to size 3\( \times \)3.)

**TI-89** You can use the \texttt{CHARA} program to check your answers in Exercises 9–14. Note that if \( A \) is \( n \times n \), running this program produces a vector listing the coefficients of the characteristic polynomial of \( A \), in order of decreasing powers of \( \lambda \), beginning with \( \lambda^n \). If the polynomial is of odd degree, the coefficients are multiplied by \(-1\), to make \(+1\) the coefficient of \( \lambda^n \) this corresponds to finding the determinant of \( \lambda I - A \). (\texttt{CHARA} works for matrices up to size 3\( \times \)3.)

**SECTION 5.3 Diagonalization**

**TI-84 Family** To practice the diagonalization procedure in this section, you should use \texttt{NULB} to produce eigenvectors. For Exercises 33-36, you have to find the eigenvalues first. See notes from Section 5.1

The program \texttt{EIGEN} calculates eigenvalues for a given square matrix with less than four rows. For each eigenvalue, use \texttt{NULB} on \( A - \lambda I \) to create a matrix whose column(s) give a basis for the eigenspace corresponding to the given eigenvalue.

**TI-89** The command \texttt{eigvl(aa)}, found by pressing \( \texttt{2nd} \) \( \texttt{9} \), scrolling down to, or pressing \( \texttt{4:Matrix} \), and finally scrolling down to \( \texttt{9:eigvl}() \) produces a list of the eigenvalues of the matrix \( aa \). To learn the diagonalization procedure, you should use the method of Section 5.1 to produce the eigenvectors. For each eigenvalue, use \texttt{NULB} on \( A - \lambda I \) to create a matrix whose column(s) give a basis for the eigenspace corresponding to the given eigenvalue.

In later work you can automate the diagonalization process. The command \texttt{eigvc\( pp \)}, found by pressing \( \texttt{2nd} \) \( \texttt{9} \), scrolling down to, or pressing \( \texttt{4:Matrix} \), and finally scrolling down to \( \texttt{A:eigvc}() \) produces a matrix \( pp \) (or \( P \)) such that \( AP = PD \), where \( D \) is the diagonal matrix you can create from the list of eigenvalues, with the eigenvalues in the diagonal. If \( A \) happens to be diagonalizable, then \( P \) will be invertible. In any case, \( P \) is likely to be quite different from what you construct for your homework. The columns of \( P \) may be scaled.
SECTION 5.5 Complex Eigenvalues

TI-84 Family The calculators in the TI-84 Family do not compute complex eigenvalues and eigenvectors directly.

TI-89 The \texttt{eigVl()} and \texttt{eigVc()} commands (mentioned in Section 5.3) also work for matrices with complex eigenvalues. In this case the resulting list of eigenvalues or the matrix containing the eigenvectors as columns have some complex entries. To view these complex values in the form \(a+bi\), the Complex format in the \texttt{MODE} menu must be set to rectangular.

SECTION 5.6 Plotting Discrete Trajectories

TI-84 Family Given a vector \(B\) (representing a point \(x_k\)) and a matrix \(A\) (the transition matrix) the command \([A][B][B]\) will compute the next point on the trajectory. Use the \(2\) keys to repeat the command over and over.

TI-89 Given a vector \(x\) (representing a point \(x_k\)) and a matrix \(A\) (the transition matrix) the command \(A\times x\) will compute the next point on the trajectory. Use the \(+\) key to repeat the command over and over.

SECTION 5.8 Power Method and Inverse Power Method

TI-84 Family Set your calculator to display as many decimal places as possible in the \texttt{MODE} menu. The algorithms below assume that \(A\) has a strictly dominant eigenvalue, and the initial vector is \(x\), with largest entry 1 (in magnitude).

The Power Method: When the following steps are executed over and over, the values of \(x\) (also known as matrix \(B\)) approach (in many cases) an eigenvector for a strictly dominant eigenvalue. (The program \texttt{VAMX} prompts for a vector and returns the entry in the vector with the maximum absolute value and stores it in \(M\).)

1. \([A][B][C]\) Stores \(A\times B\) in for \(C\)
2. \texttt{VAMX} Calculates \(M\) (estimate for the eigenvalue)
3. \([C]/M[B]\) Estimate for the eigenvector

As these commands are repeated, the numbers that appear are the \(\mu_k\) that approach the dominant eigenvalue. You can program your calculator to perform this sequence a certain number of times by using a loop structure. See the TI-83+ Guidebook for more information about programming.

The Inverse Power Method: Store the initial estimate for the eigenvalue in the variable \(Z\) and enter the command \([A]-Z\times \text{Identity}(n)[C]\), where \(n\) is the number of columns of \(A\). Then enter the commands

1. \([C]^{-1}[B][D]\) Solves the equation \((A - ZI)y = x\)
2. \texttt{VAMX}
3. \texttt{2Z+i*Z!M} Calculates \(M\) (estimate for the eigenvalue)
4. \([C]/M[B]\) Estimate for the eigenvector

As these commands are repeated, lines (3) and (4) produce the sequences \(\{v_k\}\) and \(\{x_k\}\) described in the text.

Copyright © 2016 Pearson Education, Inc.
**TI-89** Set your calculator to display as many decimal places as possible in the **MODE** menu. The algorithms below assume that $A$ has a strictly dominant eigenvalue, and the initial vector is $x$, with largest entry 1 (in magnitude).

The Power Method: When the following steps are executed over and over, the values of $x$ approach (in many cases) an eigenvector for a strictly dominant eigenvalue. The program `VAMX` prompts for a vector and returns the entry in the vector with the maximum absolute value and stores it in $m$.

1. $aa*bb!cc$ Stores $A \times B$ in for $C$
2. `VAMX` Calculates $m$ (estimate for the eigenvalue)
3. $cc/m!xx$ Estimate for the eigenvector

As these commands are repeated, the numbers that appear are the $\mu_k$ that approach the dominant eigenvalue. You can program your calculator to perform this sequence a certain number of times by using a loop structure. See the TI-89 Guidebook for more information about programming with the TI-89.

The Inverse Power Method: Store the initial estimate for the eigenvalue in the variable $z$ and enter the command $aa-z*identity(n)!yy$, where $n$ is the number of columns of $aa$. Then enter the commands

1. $yy Z \cdot \bullet \ast xx!zz$ Solves the equation $(A - ZI)y = x$
2. `VAMX`
3. $uu Z \cdot \bullet \ast z!m$ Calculates $m$ (estimate for the eigenvalue)
4. $yy/m!xx$ Estimate for the eigenvector

As these commands are repeated, lines (3) and (4) produce the sequences $\{v_k\}$ and $\{x_k\}$ described in the text.

**SECTION 6.1 Inner Product and Norm**

**TI-84 Family** If $A$ and $B$ are two column vectors of the same size, their inner product (or dot product) can be computed by $A^T* B$ or $B^T* A$ (the $T$ operator is in the 2 $\Rightarrow$ MATH menu). To calculate the norm of a vector, you need only take the square root of the dot product of a vector with itself. The keystrokes needed to find the length of the vector stored in $[A]$ are:

$2 \times 2 \ Z \ (1,1)$ Computes the square root of the dot product

See the note for Section 2.1.

**TI-89** The inner product of two vectors is found with the `dotP(` command (found by pressing 2 1, scrolling down to, or pressing 4:Matrix, and then choosing L:Vector ops). The `norm(` command, in the same matrix menu but under H:Norms, produces the length of a vector. See the note for Section 2.1.

---

Copyright © 2016 Pearson Education, Inc.
SECTION 6.2 Orthogonality

TI-84 Family  In Exercises 1–9 and 17–22, the fastest way (counting keystrokes) with the calculators in the TI-84 Family to test a set such as \{u_1, u_2, u_3\} for orthogonality is to use a matrix \( U = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix} \). See the proof of Theorem 6.

For vectors \( y \) and \( u \), the orthogonal projection of \( y \) onto \( u \), (called \([B]\) and \([C]\) respectively) is:

\[ ([B]^T[C]/[C]^T[C])*[C] \]

TI-89  In Exercises 1–9 and 17–22, the fastest way (counting keystrokes) with the TI-89 to test a set such as \{u_1, u_2, u_3\} for orthogonality is to use a matrix \( U = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix} \). See the proof of Theorem 6. For vectors \( y \) and \( u \), the orthogonal projection of \( y \) onto \( u \), denoted \( yy \) and \( uu \), is:

\[ (\text{dotP}(yy,uu)/\text{dotP}(uu,uu))*uu \]

SECTION 6.3 Orthogonal Projections

TI-84 Family  The orthogonal projection of \( y \) onto a single vector was described in the TI-84 Family note for Section 6.2. The orthogonal projection onto the set spanned by an orthogonal set of nonzero vectors is the sum of the one-dimensional projections. Another way to construct this projection is to normalize the orthogonal vectors, (use the \textsc{unitV} program) place them in the columns of a matrix \( A \), and use Theorem 10. For instance if \{\( B, C, D \)\} is an orthogonal set of nonzero vectors, they can be normalized by running the \textsc{unitV} program for each vector. Then, to construct \( A \), input the following commands:

\[
\begin{align*}
\text{augment}([B],[C])![A] & \quad \text{Augments } [B] \text{ and } [C] \text{ together} \\
\text{augment}([A],[D])![A] & \quad \text{Augments } [B] \text{ and } [C] \text{ and } [D] \text{ together}
\end{align*}
\]

The resulting matrix \([A]\) has orthonormal columns, and \([A][A]^T*y\) produces the orthogonal projection of \( y \) onto the subspace spanned by \{\( B, C, D \)\}.

TI-89  The orthogonal projection of \( y \) onto a single vector was described in the TI-89 note for Section 6.2. The orthogonal projection onto the set spanned by an orthogonal set of nonzero vectors is the sum of the one-dimensional projections. Another way to construct this projection is to normalize the orthogonal vectors, (use the \textsc{unitV} by pressing \( 2 \) \text{\normalfont L}, scrolling down to, or pressing \( 4: \text{Matrix} \), and then choosing \( 1: \text{Vector ops} \)) place them in the columns of a matrix \( U \), and use Theorem 10. For instance if \{\( A, B, C \)\} is an orthogonal set of nonzero vectors, they can be normalized by the \textsc{unitV} command for each vector. Then to construct \( U \), input the following commands:

\[
\begin{align*}
\text{augment}(aa,bb)!uu & \quad \text{Augments } aa \text{ and } bb \text{ together} \\
\text{augment}(uu,cc)!uu & \quad \text{Augments } aa \text{ and } bb \text{ and } cc \text{ together}
\end{align*}
\]

The resulting matrix \( uu \) has orthonormal columns, and \( uu*uu^T*y \) produces the orthogonal projection of \( y \) onto the subspace spanned by \{\( A, B, C \)\}.
SECTION 6.4 The Gram-Schmidt Process

**TI-84 Family** The program **PROJ** prompts for two vectors. It produces the orthogonal projection of one onto the other. If you wish to compute the projection of a vector onto the subspace spanned by a set of vectors given as columns of a matrix, **PROJV** will prove to be more useful.

**TI-89** The program **PROJ** prompts for two vectors. It produces the orthogonal projection of one onto the other. If you wish to compute the projection of a vector onto the subspace spanned by a set of vectors given as columns of a matrix, **PROJV** will prove to be more useful.

SECTION 6.5 The Inverse Operator $^{-1}$

**TI-84 Family** The least squares solution to $Ax = b$ is the solution of $A^T Ax = A^T b$. The reduced row echelon form for the augmented matrix for the system is displayed by the command
\[ \text{rref augment}([A]^T [A], [A]^T [B]), \]
from which the set of solutions can be determined by the method described in Section 1.2 of the text.

If the matrix $A$ is square and invertible, then Theorem 14 tells us that the unique least squares solution to $Ax = b$ is given by $(A^T A)^{-1} A^T b$. The solution in this case is displayed explicitly as a $n 	imes 1$ matrix by the command
\[ ([A]^T [A])^{-1} [A]^T [B] \]

For Exercises 15 and 16, see the Numerical Note from this section in the text. For Exercise 26, the command
\[ \text{augment}([A]^T, [B]^T) \]
creates a (partitioned) matrix whose top block is $[A]$ and the bottom block is $[B]$. This command works as long as $[A]$ and $[B]$ have the same number of columns.

**TI-89** The least squares solution to $Ax = b$ is the solution of $A^T Ax = A^T b$. The reduced row echelon form for the augmented matrix for the system is displayed by the command
\[ \text{rref augment}(aa^T aa, aa^T bb), \]
from which the set of solutions can be determined by the method described in Section 1.2 of the text.

If the matrix $A$ is square and invertible, then Theorem 14 tells us that the unique least squares solution to $Ax = b$ is given by $(A^T A)^{-1} A^T b$. The solution in this case is displayed explicitly as a $n 	imes 1$ matrix by the command
\[ (aa^T aa)^{-1} aa^T bb \]

For Exercises 15 and 16, see the Numerical Note from this section in the text. For Exercise 26, the command
\[ \text{augment}(aa^T, bb^T) \]
creates a (partitioned) matrix whose top block is $aa$ and the bottom block is $bb$. This command works as long as $aa$ and $bb$ have the same number of columns.

Copyright © 2016 Pearson Education, Inc.
SECTION 6.6 Least-Squares Solutions, Functions of Vectors

TI-84 Family Once you create the design matrix $X$ and the observation vector $y$, your computations for least-square solutions here are the same as those described in notes for Section 6.5. Here $A$ and $B$ are replaced by $X$ and $y$, respectively. The command
\[
\text{ref augment}([A]^T[A],[A]^T[y])
\]
leads to the general description of all least-squares solutions. When $X$ has linearly independent columns, the command
\[
([A]^T[A])^{-1}[A]^T[y]
\]
creates the least-squares solution. In other courses, you may choose simply to use $[A]^{-1}[B]$, which also produces a least-squares solution, except when $X$ is square and singular (or nearly singular).

TI-89 Once you create the design matrix $X$ and the observation vector $y$, your computations for least-square solutions here are the same as those described in notes for Section 6.5. Here $A$ and $B$ are replaced by $X$ and $y$, respectively. The command
\[
\text{ref augment}(aa^Taa,aa^Tbb)
\]
leads to the general description of all least-squares solutions. When $X$ has linearly independent columns, the command
\[
(aa^Taa)^{-1}aa^Tbb
\]
creates the least-squares solution. In other courses, you may choose simply to use $aa^{-1}bb$, which also produces a least-squares solution, except when $X$ is square and singular (or nearly singular).

SECTION 7.1 Orthogonal Diagonalization

TI-84 Family The program \textsc{Eigen} calculates eigenvalues for a given square matrix with less than four rows. Once the eigenvalues have been found, use the \textsc{Nulb} program to obtain eigenvectors, as in Section 5.3.

TI-89 The \textsc{eigvl} and \textsc{eigvc} keys orthogonally diagonalize any symmetric matrix $A$, but you miss the opportunity to learn the procedure of this section. Use \textsc{eigvl} to find the eigenvalues of the matrix and the \textsc{Nulb} program to obtain eigenvectors, as in Section 5.3.

SECTION 7.4 The Singular Value Decomposition

TI-84 Family The calculators in the TI-84 Family do not have the capabilities to compute eigenvectors for this exercise.

TI-89 The commands \textsc{eigvl}(aa^Taa)e and \textsc{eigvc}(aa^Taa)pp produce the list of eigen-values $e$ and an orthogonal matrix $pp$ of eigenvectors of the matrix $A^TA$, but the eigenvalues may not be in decreasing order. In such a case, you will have to rearrange things to form $V$ and $\Sigma$.
SECTION 8.1 Affine Combinations

**TI-84 Family** To complete example one from section 8.1, construct the matrices \[
[v_2 \ v_3 \ v_4 \ y]
\]
and \[
[v_1 \ v_1 \ v_1 \ v_1]
\]. Then perform \( \text{rref}() \) on the difference between the two matrices to determine the affine combination of the vectors.

**TI-89** To complete example one from section 8.1, construct the matrices \[
[v_2 \ v_3 \ v_4 \ y]
\]
and \[
[v_1 \ v_1 \ v_1 \ v_1]
\]. Then perform \( \text{rref}() \) on the difference between the two matrices to determine the affine combination of the vectors.

SECTION 8.2 Affine Independence

**TI-84 Family** The change of coordinates matrix will help solve example four. See the note in Section 4.7 on how to perform this task.

**TI-89** The change of coordinates matrix will help solve example four. See the note in Section 4.7 on how to perform this task.

SECTION 8.3 Convex Combinations

**TI-84 Family** The orthogonal projection of \( y \) onto a single vector was described in the TI-84 Family note for Section 6.2. The orthogonal projection onto the set spanned by an orthogonal set of nonzero vectors is the sum of the one-dimensional projections. Another way to construct this projection is to normalize the orthogonal vectors, use the \( \text{UNITV} \) program to place them in the columns of a matrix \( A \), and use Theorem 10. For instance if \( \{B, C, D\} \) is an orthogonal set of nonzero vectors, they can be normalized by running the \( \text{UNITV} \) program for each vector. Then, to construct \( A \), input the following commands:

\[
\text{augment}([B],[C])!A \quad \text{Augments } [B] \text{ and } [C] \text{ together}
\]
\[
\text{augment}([A],[D])!A \quad \text{Augments } [B] \text{ and } [C] \text{ and } [D] \text{ together}
\]

The resulting matrix \( A \) has orthonormal columns, and \( A^*A^T y \) produces the orthogonal projection of \( y \) onto the subspace spanned by \( \{B, C, D\} \).

**TI-89** The orthogonal projection of \( y \) onto a single vector was described in the TI-89 note for Section 6.2. The orthogonal projection onto the set spanned by an orthogonal set of nonzero vectors is the sum of the one-dimensional projections. Another way to construct this projection is to normalize the orthogonal vectors, use the \( \text{UNITV} \) command for each vector. Then to construct \( U \), input the following commands:

\[
\text{augment}(aa,bb)!uu \quad \text{Augments } aa \text{ and } bb \text{ together}
\]

Copyright © 2016 Pearson Education, Inc.
## augment(uu,cc)!uu

Augments \(aa\) and \(bb\) and \(cc\) together

The resulting matrix \(uu\) has orthonormal columns, and \(uu^*uu^T*y\) produces the orthogonal projection of \(y\) onto the subspace spanned by \(\{A, B, C\}\).

### Index of TI-84 Family Commands

<table>
<thead>
<tr>
<th><strong>Symbols</strong></th>
<th><strong>Built in Commands</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Alpha key 1 TI-2,7</td>
<td>Archive TI-4</td>
</tr>
<tr>
<td>Addition key (\oplus) TI-8</td>
<td>augment TI-6-10, 12,13,18-22</td>
</tr>
<tr>
<td>Answer key (Z) TI-5,17,18</td>
<td>det TI-12</td>
</tr>
<tr>
<td>Bracket keys ((,) TI-1</td>
<td>dim TI-7,8</td>
</tr>
<tr>
<td>Catalog menu (N) TI-14</td>
<td>Fill TI-7</td>
</tr>
<tr>
<td>Exponent key (\theta) TI-9</td>
<td>Frac TI-8,15</td>
</tr>
<tr>
<td>Enter key (\subseteq) TI-1,2,4,6-9,13,15</td>
<td>identity TI-9,14,17</td>
</tr>
<tr>
<td>Entry key (\langle) TI-16</td>
<td>randM TI-9,13</td>
</tr>
<tr>
<td>Inverse key (\square) TI-10,13,17,19,20</td>
<td>ref TI-12,13,20</td>
</tr>
<tr>
<td>Math menu (\mathbb{M}) TI-8,15</td>
<td>*row TI-4,6,7,9</td>
</tr>
<tr>
<td>Matrix menu (\mathbb{M}) TI-1,2,4,6-9,11-14,18</td>
<td>*row+ TI-4-6</td>
</tr>
<tr>
<td>Memory key (\wedge) TI-4</td>
<td>rowswap TI-4-6,12</td>
</tr>
<tr>
<td>Mode menu TI-16</td>
<td>ref TI-11-15,19,21</td>
</tr>
<tr>
<td>Multiplication key (\downarrow) TI-8,13,17,18,19,20,22</td>
<td>solve TI-14</td>
</tr>
<tr>
<td>Program menu (\mathbb{P}) TI-4</td>
<td>(^T) TI-9,10,18-20,22</td>
</tr>
<tr>
<td>Quit key 5 TI-1</td>
<td>UnArchive TI-4</td>
</tr>
<tr>
<td>Second key (\psi) TI-1,2,4,6-9,11-14,16-18</td>
<td></td>
</tr>
<tr>
<td>Square root key (\sqrt{\phantom{0}}) TI-18</td>
<td></td>
</tr>
<tr>
<td>Squaring key (\sqrt[2]{\phantom{0}}) TI-9</td>
<td></td>
</tr>
<tr>
<td>Store key (\downarrow) TI-1,2,5-10,13,14,16-18,22</td>
<td></td>
</tr>
<tr>
<td>Subtraction key (\neq) TI-8,14,17,21</td>
<td></td>
</tr>
<tr>
<td>Zoom key (\theta) TI-2,7</td>
<td></td>
</tr>
</tbody>
</table>
Supplemental Programs (ALINEAR)
BGAUSS TI-7,9
CHARA TI-15
EIGEN TI-16,21
GAUSS TI-7,9-13
JCOL TI-8
JROW TI-8
LU TI-11
MDIAV TI-12
NULB TI-14-16,21
PRDCT TI-12
PROJ TI-19
PROJV TI-19
UNITV TI-18,21,22
VAMX TI-16,17
# Index of TI-89 Commands

## Symbols
- **Addition key** \( \oplus \) TI-9
- **Answer key** \( \approx \) TI-5
- **Apps menu** \( \mathbb{O} \) TI-2,3
- **Bracket keys** \( \mathbb{H} \) TI-2
- **Exponent key** \( \approx \) TI-9,10,13,17,20
- **Enter key** \( \mathbb{T} \subseteq \) TI-2-6,8,9,14,16
- **F1 key** \( \mathbb{F} \) TI-4
- **F2 key** \( \mathbb{F} \) TI-14
- **Home key** \( \mathbb{V} \) TI-3
- **Math menu** \( \mathbb{[} \) TI-5-7,9-16,18,19,22
- **Mode menu** TI-8,15-17
- **Multiplication key** \( \otimes \) TI-8,9,13,17,18,20-22
- **Negative key** \( \bullet \) TI-10,13,17,20
- **Second key** \( \mathbb{S} \) TI-4-7,9-16,18,19,22
- **Square root key** \( \sqrt{} \) TI-18
- **Squaring key** \( \approx \) TI-9
- **Store key** \( \mathbb{S} \) TI-2,6,7,9,10,13-17,19,21,22
- **Subtraction key** \( \ominus \) TI-9,15,17,22
- **Var-link** \( \mathbb{D} \) TI-4

## Supplemental Programs (ALINEAR)
- **BGAUSS** TI-7,10
- **COL** TI-9
- **CHARA** TI-15
- **GAUSS** TI-7,10-13
- **MDIAV** TI-12
- **NULB** TI-15,16,21
- **PRDCT** TI-12
- **PROJ** TI-19
- **PROJV** TI-19
- **VAMX** TI-17

## Built in Commands
- **Archive** TI-4
- **augment** TI-6,7,10-13,19,20-22
- **det** TI-12
- **dim** TI-7,9
- **dotP** TI-9,18
eigVc TI-16,21
- **eigVL** TI-16,21
- **identity** TI-10,15,17
- **LU** TI-11
- **mRow** TI-5-7,10
- **mRowAdd** TI-5,6
- **norm** TI-18
- **randMat** TI-9,14
- **ref** TI-12,13,20
- **row ops** TI-5
- **rowSwap** TI-5,6,12
- **rref** TI-11-15,19-21
- **solve** TI-14
- \( \mathbb{T} \) TI-9,10,19-21
- **Unarchive** TI-4
- **unitV** TI-19,22
- **vector ops** TI-9,18,19,22