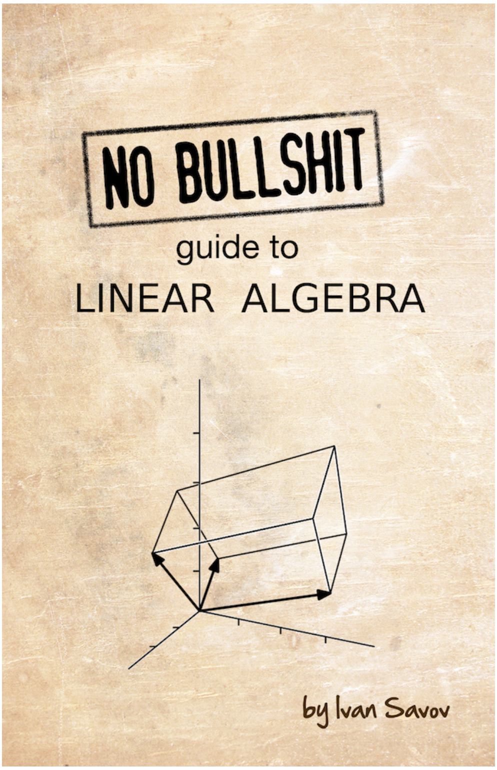
# No Bullshit Guide to Linear Algebra

Extended book preview




The full book has 596 pages and includes 128 exercises and 200 solved problems. This preview has been chosen to showcase some key aspects of the book (chapter intros, formulas, explanations, and figures). You can buy the full book for only US$29 at https://gum.co/noBSLA .

**No bullshit guide to linear algebra**
by Ivan Savov



Published by Minireference Co.
Montréal, Québec, Canada
`minireference.com | @minireference | fb.me/noBSguide`
For inquiries, contact the author at `ivan@minireference.com`

*Library and Archives Canada Cataloguing in Publication*

Savov, Ivan, 1982-, author
No bullshit guide to linear algebra / Ivan Savov. — Second edition.

ISBN 978-0-9920010-2-5 (softcover)

1. Algebras, Linear–Textbooks. 2. Mathematics–Textbooks.
3. Textbooks. I. Title.

QA184.2.S29 2017 512′.5 C2017-902491-4

Cover image: Jim Hefferon and Nicholas Longo, CC-BY-SA-2.5
`http://commons.wikimedia.org/wiki/File:Linalg_parallelepiped.png`

Second edition
v2.2
git commits `master@1054:67a22f1 + vers/LA@1456:ac2c497`

`ISBN 978-0-9920010-2-5`

10 9 8 7 6 5 4 3 2 1

# Contents

# Concept maps

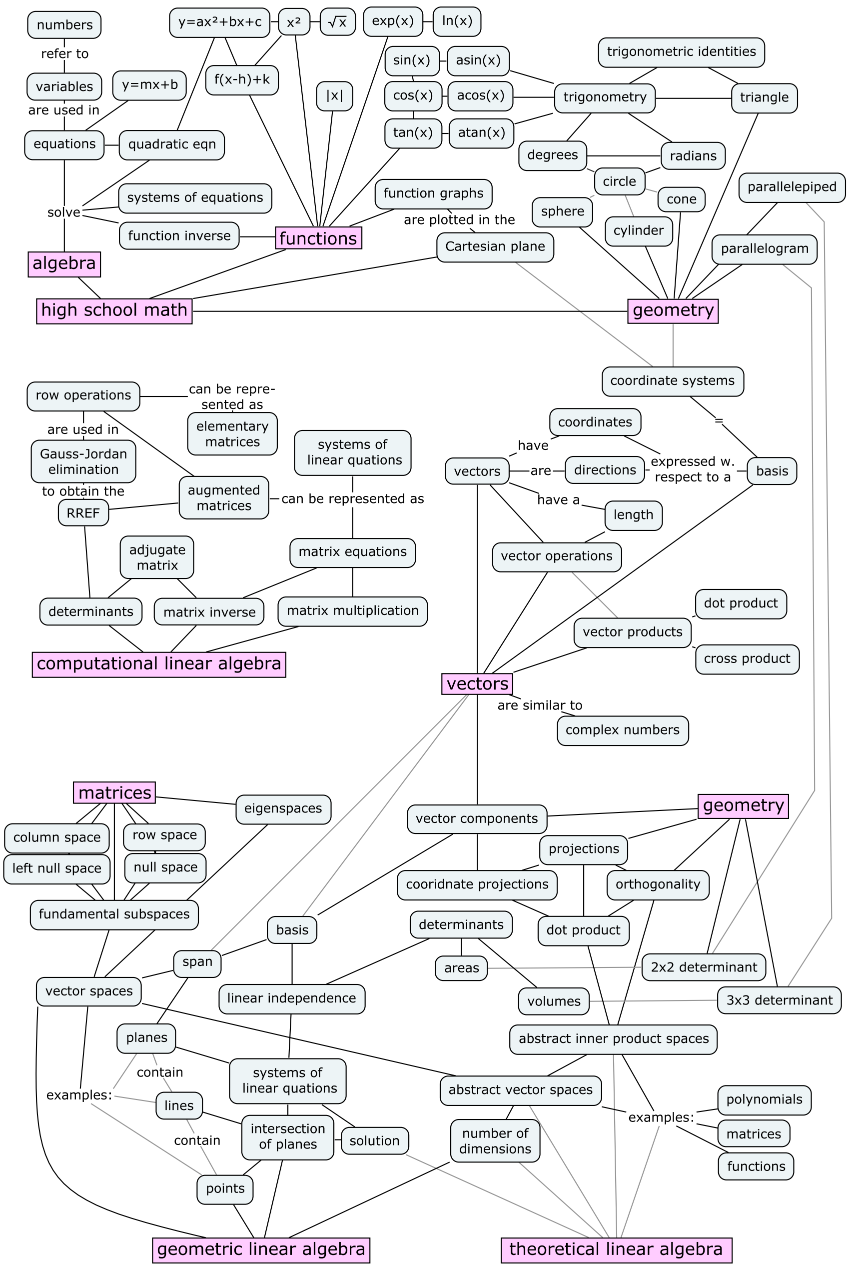



**Figure 1:** This diagram shows the concepts and topics covered in this book.

Consult the index on page 577 to find the exact location in the book where each concept is defined.

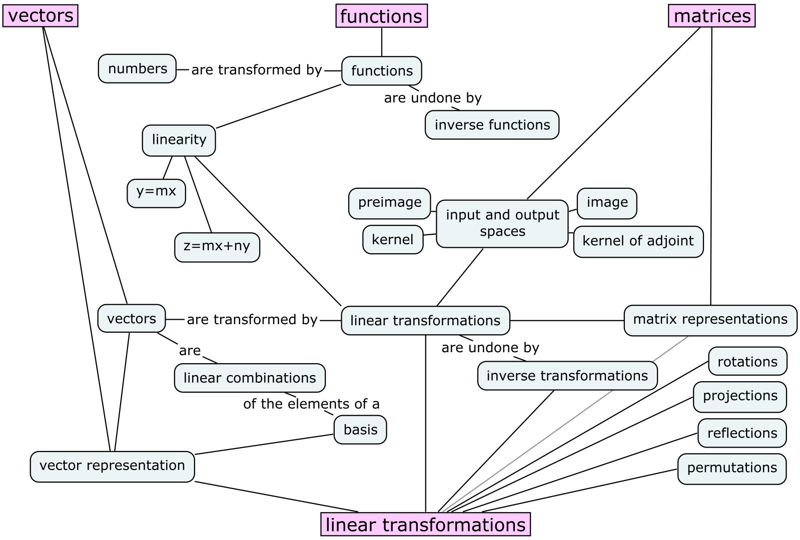



**Figure 2:** Chapter 5 is about linear transformations and their properties.

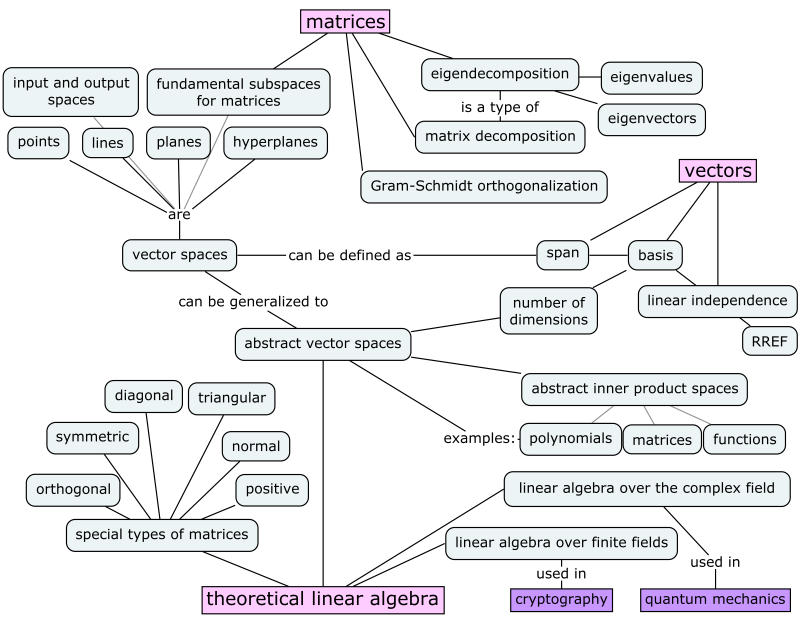



**Figure 3:** Chapter 6 covers theoretical aspects of linear algebra.

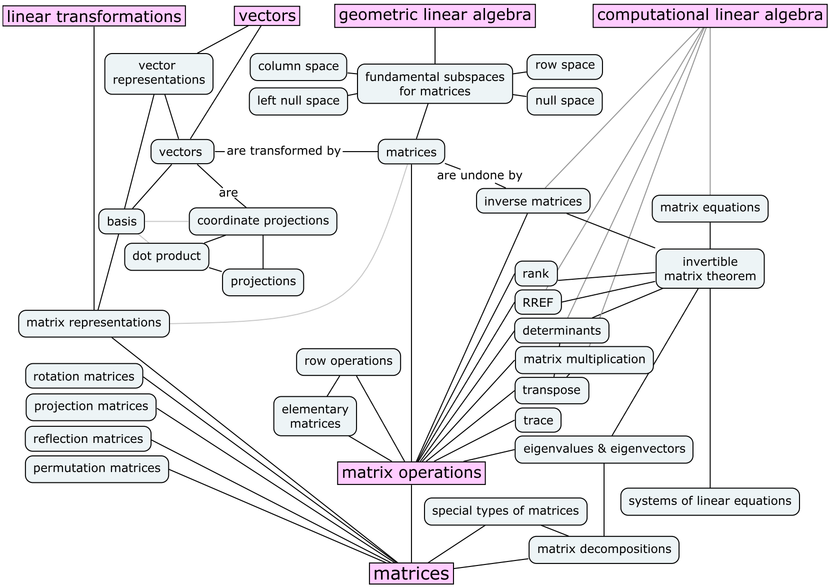



**Figure 4:** Matrix computations play an important role throughout this book. Matrices are used to represent linear transformations, systems of linear equations, and various geometric operations.

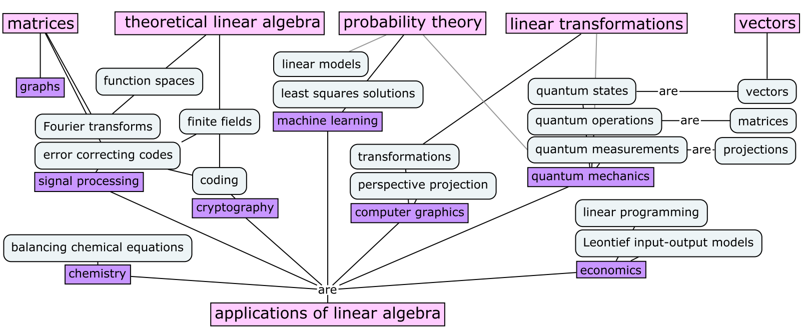



**Figure 5:** The book concludes with three chapters on linear algebra applications. In Chapter 7 we'll discuss applications to science, economics, business, computing, and signal processing. In Chapter 8 we'll explain probability theory, and finally in Chapter 9 we'll introduce quantum mechanics.

You can annotate the concept maps with your current knowledge of each concept to keep track of your progress. Add a single dot (•) next to all concepts you've heard of, two dots (••) next to concepts you think you know, and three dots (•••) next to concepts you've used in exercises and problems. By collecting some dots every week, you'll be able to move through the material in no time at all.

If you don't want to mark up your book, you can download a printable version of the concept maps here: `bit.ly/LAcmaps`.

# Preface

This is a book about linear algebra and its applications. The material is presented at the level of a first-year university course, in an approachable style that cuts to the point. It covers both practical and theoretical aspects of linear algebra, with extra emphasis on explaining the connections between concepts and building a solid understanding of the material.

This book is designed to **give readers access to advanced math modelling tools regardless of their academic background**. Since the book includes all the prerequisites needed to learn linear algebra, it's suitable for readers of any skill level—including those who don't feel comfortable with fundamental math concepts.

## Why learn linear algebra?

Linear algebra is one of the most fundamental and all-around useful subjects in mathematics. The practical skills learned by studying linear algebra—such as manipulating vectors and matrices—form an essential foundation for applications in physics, computer science, statistics, machine learning, and many other fields of scientific study. Learning linear algebra can also be a lot of fun. Readers will experience numerous *knowledge buzz* moments when they feel math ideas "click" into place and understand the connections between concepts.

The tools of linear algebra form a bridge to more advanced areas of mathematics. For example, learning about *abstract vector spaces* will help readers recognize the common "vector space structure" in seemingly unrelated mathematical objects like matrices, polynomials, and functions. Linear algebra techniques apply not only to standard vectors, but to *all* mathematical objects that are vector-like!

## What's in this book?

Each section is a self-contained tutorial that covers the definitions, formulas, and explanations associated with a single topic. Check out the concept maps on the preceding pages to see the book's many topics and the connections between them.

The book begins with a review chapter on numbers, algebra, sets, equations, functions, geometry, and trigonometry (Chapter 1). If you haven't previously studied these concepts, or if you feel your math skills are a little "rusty," read this chapter and work through the exercises and problems provided. If you feel confident in your high school math abilities, jump straight to Chapter 2, where the linear algebra begins.

Chapters 2–6 cover the core topics of linear algebra: vectors, bases, analytical geometry, matrices, linear transformations, matrix representations, vector spaces, inner product spaces, eigenvectors, and matrix decompositions. These chapters contain the material required for every university-level linear algebra course. Each section contains plenty of exercises so you can test your understanding as you read; and each chapter concludes with an extensive list of problems for further practice.

Chapters 7, 8, and 9 discuss various applications of linear algebra. Though this material isn't likely to appear on any final exam, these chapters serve to demonstrate the power of linear algebra techniques and their relevance to many areas of science. The mini-course on quantum mechanics (Chapter 9) is unique to this book. Read this chapter to understand the fascinating laws of physics that govern the behaviour of atoms and photons.

## Is this book for you?

The quick pace and lively explanations in this book provide interesting reading for students and non-students alike. Whether you're learning linear algebra for a course, reviewing material as a prerequisite for more advanced topics, or generally curious about the subject, this book will help you find your way in the land of linear algebra.

Students and educators can use this book as the main textbook for any university-level linear algebra course. It contains everything students need to know to prepare for a linear algebra final exam. Don't be fooled by the book's small size compared to other textbooks: it's all in here. The text is compact because we've distilled the essentials and removed the unnecessary crud.

## Publisher

The starting point of the **No Bullshit Guide** textbook series dates back to my student days when I was required to purchase expensive course textbooks, which were long and tedious to read. I said to myself, "Something must be done," and started a publishing company to produce textbooks that explain math and physics concepts clearly, concisely, and affordably.

Minireference Co. is a publisher specializing in math and science. Our goal is to make advanced math modelling tools accessible to everyone by producing affordable textbooks that explain math concepts clearly and concisely. It's time for a break from traditional textbooks that are expensive, heavy, and tedious to read. The books in the **No Bullshit Guide** series are an efficient and enjoyable alternative for learning mathematical subjects.

## About the author

I have been teaching math and physics for more than 17 years as a private tutor. Through this experience, I learned to break complicated ideas into smaller, interconnected chunks that are easy to understand. An interesting feedback loop occurs when students learn concepts in small, manageable chunks: they experience *knowledge buzz* whenever concepts "click" into place, and this excitement motivates them to continue learning more. I know this from first-hand experience, both as a teacher and as a student. I completed my undergraduate studies in electrical engineering, then stayed on to earn a M.Sc. in physics, and a Ph.D. in computer science from McGill University. Linear algebra played a central role throughout my studies. With this book, I want to share with you some of what I've learned about this expansive subject.

Ivan Savov
Montreal, 2020

# Introduction

In recent years we've seen countless advances in science and technology. Modern science and engineering fields have developed advanced models for understanding the real world, predicting the outcomes of experiments, and building useful technology. Although we're still far from obtaining a "theory of everything" that can fully explain reality and predict the future, we do have a significant understanding of the natural world on many levels: physical, chemical, biological, ecological, psychological, and social. And, since mathematical models are leveraged throughout these fields of study, anyone interested in contributing to scientific and technological advances must also understand mathematics.

The linear algebra techniques you'll learn in this book are some of the most powerful mathematical modelling tools that exist. At the core of linear algebra lies a very simple idea: *linearity*. A function $f$ is *linear* if it obeys the equation

$$f(a\mathbf{x}_1 + b\mathbf{x}_2) \;=\; af(\mathbf{x}_1) + bf(\mathbf{x}_2),$$

where $\mathbf{x}_1$ and $\mathbf{x}_2$ are any two inputs of the function. We use the term *linear combination* to describe any expression constructed from a set of variables by multiplying each variable by a constant and adding the results. In the above equation, the linear combination $a\mathbf{x}_1 + b\mathbf{x}_2$ of the inputs $\mathbf{x}_1$ and $\mathbf{x}_2$ is transformed into the linear combination $af(\mathbf{x}_1) + bf(\mathbf{x}_2)$ of the outputs of the function $f(\mathbf{x}_1)$ and $f(\mathbf{x}_2)$. **Essentially, linear functions transform a linear combination of inputs into the *same* linear combination of outputs.** If the input to the linear function $f$ consists of five parts $\mathbf{x}_1$ and three parts $\mathbf{x}_2$, then the output of the function will consist of five parts $f(\mathbf{x}_1)$ and three parts $f(\mathbf{x}_2)$. That's it, that's all! Now you know everything there is to know about linear algebra. The rest of the book is just details.

## Linear models are super useful

A significant proportion of the math models used in science describe *linear relationships* between quantities. Mathematicians, scientists, engineers, and business analysts develop and use linear models to make sense of the systems they study. Linear models are popular because they are **easy to describe mathematically**. We can obtain the parameters of a linear model for a real-world system by analyzing the system's behaviour for relatively few inputs. Let's illustrate this important point with an example.

**Example** You're visiting an art gallery. Inside, the screen of a tablet computer is being projected onto a giant wall. Anything you draw on the tablet instantly appears projected onto the wall. However, the tablet's user interface doesn't give any indication about how to hold the tablet "right side up." How can you find the correct orientation of the tablet so your drawing won't appear rotated or upside-down?

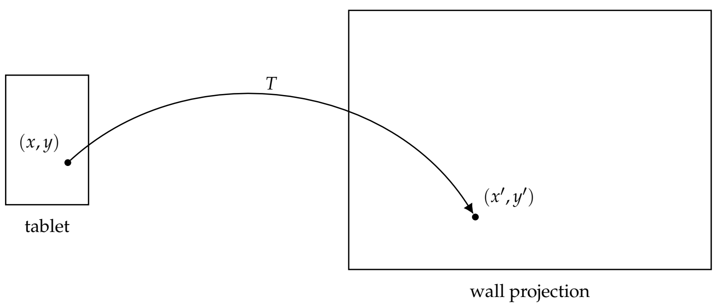



**Figure 6:** An unknown linear transformation $T$ maps "tablet coordinates" to "wall coordinates." How can we characterize $T$?

The tablet's screen is a two-dimensional *input space* described by coordinates $(x, y)$ and the wall projection is a two-dimensional *output space* described by wall coordinates $(x', y')$. You're looking for the unknown transformation $T$ that maps the pixels of the tablet screen (the input space) to the projection on the wall (the output space):

$$(x, y) \xrightarrow{T} (x', y').$$

This task is directly analogous to the tasks scientists and engineers face every day when trying to model real-world systems by observing how systems transform inputs to outputs. If the unknown transformation $T$ is linear, you can learn what it is very quickly, using only two swipes on the tablet screen.

To understand how $T$ transforms screen coordinates $(x, y)$ to wall coordinates $(x', y')$, you can use this two-step "probing" procedure:

1. Draw a horizontal line on the tablet to represent the $x$-direction in the input space $\rightarrow = (1,0)$. You observe the output $\nearrow$ projected on the wall. This tells you horizontal lines are transformed to northeast diagonal lines in the wall-projection space.

2. Draw a vertical line in the $y$-direction $\uparrow = (0,1)$ on the tablet. You observe the output $\nwarrow$ appears on the wall. This means vertical lines on the tablet screen turn into northwest diagonal lines when projected on the wall.

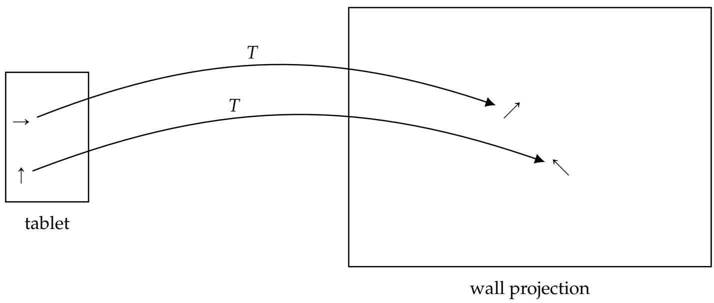



**Figure 7:** Drawing a short horizontal arrow $\rightarrow$ on the tablet screen results in a northeast diagonal projection on the wall $\nearrow$. Drawing a vertical arrow $\uparrow$ on the tablet results in a northwest diagonal line projected on the wall $\nwarrow$.

Here comes the interesting part: now that you know the outputs $\nearrow$ and $\nwarrow$ produced for the two input directions, you can **predict the linear transformation's output for any other input.** Let's look at the math equations that show why this is true.

Suppose you want to predict what will appear on the wall if you draw a line on the tablet in the direction $(3,2)$. The coordinates $(3,2)$ describe a swipe with length 3 units in the $x$-direction and 2 units in the $y$-direction. The input coordinates $(3,2)$ can be written as $3(1,0)+2(0,1) = 3 \rightarrow + 2 \uparrow$. Because you know $T$ is linear, the wall projection of this input will have a length equal to 3 times the $x$-direction output $\nearrow$ plus 2 times the $y$-direction output $\nwarrow$:

$$T\big(3\rightarrow + 2\uparrow\big) \;=\; 3T\big(\rightarrow\big) + 2T\big(\uparrow\big) \;=\; 3\nearrow + 2\nwarrow .$$

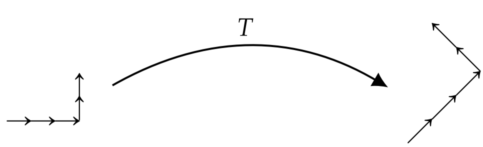



**Figure 8:** The linear transformation $T$ maps the input $3\rightarrow + 2\uparrow$ to the output $3T\big(\rightarrow\big) + 2T\big(\uparrow\big) = 3\nearrow + 2\nwarrow$.

Knowing that the input $\rightarrow$ produces the output $\nearrow$ and the input $\uparrow$ produces the output $\nwarrow$ allows you to determine the linear transformation's output for all other inputs. Every input $(a, b)$ can be written as a linear combination: $(a, b) = a(1, 0) + b(0, 1) = a \rightarrow + b\uparrow$. Since you know $T$ is linear, you know the corresponding output will be

$$T\big(a \rightarrow + b\uparrow\big) \;=\; aT\big(\rightarrow\big) + bT\big(\uparrow\big) \;=\; a \nearrow + b \nwarrow .$$

Since you can predict the output of $T$ for all possible inputs, you have obtained a complete characterization of the linear transformation $T$.

The probing procedure we used to characterize the two-dimensional tablet-to-wall linear transformation (denoted $T : \mathbb{R}^2 \to \mathbb{R}^2$) can be used to study arbitrary linear transformations with $n$-dimensional inputs and $m$-dimensional outputs (denoted $T : \mathbb{R}^n \to \mathbb{R}^m$). **Knowing the outputs of a linear transformation $T$ for all "directions" in its input space gives us a complete characterization of $T$.**

**TL;DR** The linear property allows us to analyze multidimensional systems and processes by studying their effects on a small set of inputs. This is the essential reason linear models are used so widely in science. Without this linear structure, characterizing the behaviour of unknown input-output systems would be a much harder task.

## Linear transformations

Linear transformations will be a central topic throughout this book. You can think of linear transformations as "vector functions" and understand their properties as analogous to the properties of the regular functions you're familiar with. The action of a function on a number is similar to the action of a linear transformation on a vector:

$$\begin{aligned}
\text{function } f : \mathbb{R} \to \mathbb{R} \;&\Leftrightarrow\; \text{linear transformation } T : \mathbb{R}^n \to \mathbb{R}^m \\
\text{input } x \in \mathbb{R} \;&\Leftrightarrow\; \text{input } \vec{x} \in \mathbb{R}^n \\
\text{output } f(x) \in \mathbb{R} \;&\Leftrightarrow\; \text{output } T(\vec{x}) \in \mathbb{R}^m \\
\text{inverse function } f^{-1} \;&\Leftrightarrow\; \text{inverse transformation } T^{-1} \\
\text{roots of } f \;&\Leftrightarrow\; \text{kernel of } T
\end{aligned}$$

Studying linear algebra will expose you to many new topics associated with linear transformations. You'll learn about concepts like vector spaces, projections, rotations, and orthogonalization procedures. Indeed, a first linear algebra course introduces many advanced, abstract ideas; yet all the new ideas you'll encounter can be seen as extensions of ideas you're already familiar with. Linear algebra is the vector-upgrade to your high school knowledge of functions.

## Prerequisites

To understand linear algebra, you must have some preliminary knowledge of fundamental math concepts like numbers, equations, and functions. For example, you should be able to tell me the meaning of the parameters $m$ and $b$ in the equation $f(x) = mx + b$. If you do not feel confident about your basic math skills, don't worry. Chapter 1 is a prerequisites chapter specially designed to help bring you quickly up to speed on the high school math material.

## Executive summary

The book is organized into nine chapters. Chapters 2 through 6 are the core of linear algebra. Chapters 7 through 9 contain optional reading about linear algebra applications. The concept maps on pages v, vi, and vii illustrate the connections between the topics we'll cover. I know the maps may seem informationally intimidating at first sight, but don't worry—the book is split into tiny chunks, and we'll navigate the material step by step. It will be like Mario World, but in $n$ dimensions and with a lot of bonus levels.

Chapter 2 is a general introduction to linear algebra. Linear algebra is the math of vectors and matrices, so we'll start by defining the mathematical operations we can perform on vectors and matrices.

In Chapter 3, we'll tackle the computational aspects of linear algebra. By the end of this chapter you'll know how to solve systems of equations, transform a matrix into its *reduced row echelon form*, compute the product of two matrices, and find the *determinant* and the *inverse* of a square matrix. Each of these computational tasks can be tedious to carry out by hand and can require lots of steps. There is no way around this; we must do the grunt work before we get to the cool stuff.

In Chapter 4, we'll review the properties and the equations that describe basic geometric objects like points, lines, and planes. We'll learn how to compute projections onto vectors, projections onto planes, and distances between objects. We'll also review the meaning of vector coordinates, which are lengths measured with respect to a basis. We'll learn about linear combinations of vectors, the *span* of a set of vectors, and formally define what a *vector space* is.

Chapter 5 is about linear transformations. Armed with the computational tools from Chapter 3 and the geometric intuition from Chapter 4, we can tackle the core subject of linear algebra: linear transformations. We'll explore in detail the correspondence between linear transformations ($T : \mathbb{R}^n \to \mathbb{R}^m$) and their representation as $m \times n$ matrices. We'll also learn how the entries in a matrix representation depend on the choice of basis for the input and output spaces of the transformation. Section 5.4 on the invertible matrix theorem serves as a midway checkpoint for your understanding of linear algebra. This theorem connects several seemingly disparate concepts: reduced row echelon forms, matrix inverses, and determinants. The invertible matrix theorem links all these concepts and highlights the properties of invertible linear transformations that distinguish them from non-invertible transformations.

Chapter 6 covers more advanced theoretical topics of linear algebra. We'll define the *eigenvalues* and the *eigenvectors* of a square matrix. We'll see how the eigenvalues of a matrix tell us important information about the properties of the matrix, and learn about the special names given to different types of matrices based on the properties of their eigenvalues. In Section 6.3 we'll discuss *abstract vector spaces*. Abstract vectors are mathematical objects that—like vectors—have components and can be scaled, added, and subtracted by manipulating their components. Section 6.7 will discuss linear algebra with complex numbers.

In Chapter 7, we'll discuss the applications of linear algebra. If you've done your job learning the material in the first six chapters, you'll get to learn all the cool things you can do with linear algebra. Chapter 8 will introduce the basic concepts of probability theory. Chapter 9 contains an introduction to quantum mechanics.

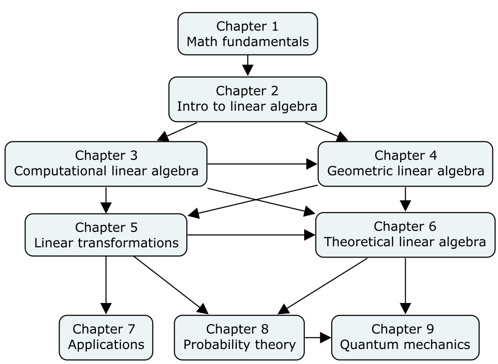



**Figure 9:** The dependency structure of the chapters in the book.

Figure 9 shows the prerequisite structure between the chapters. If you're new to linear algebra, it would be best to read the chapters in order. If you already have some experience with the subject, you can skip ahead to the parts you're interested in. Use the table of contents, the concept maps on pages v–vii, and the index on page 577 to navigate the book. The chapters and sections in the book are self-contained so they will make sense even if you read them out of order.

# Chapter 1

# Math fundamentals

In this chapter we'll review the fundamental ideas of mathematics—the prerequisites for learning linear algebra. We'll define the different types of numbers and the concept of a function, which is a transformation that takes numbers as inputs and produces numbers as outputs. Linear algebra is the extension of these ideas to many dimensions: instead of doing math with numbers and functions, in linear algebra we'll be doing math with vectors and linear transformations.

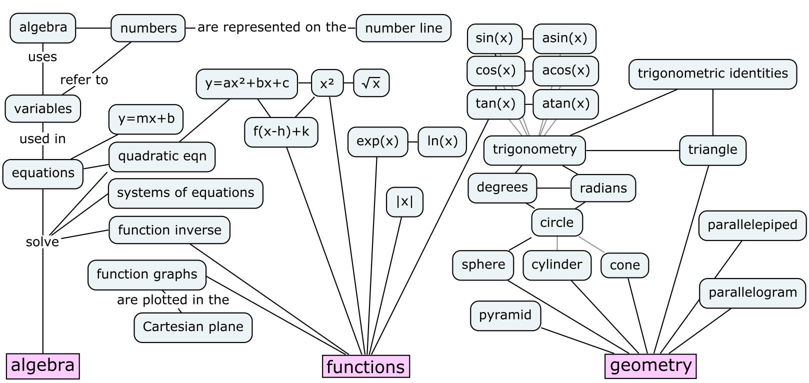



**Figure 1.1:** A concept map showing the mathematical topics covered in this chapter. We'll learn how to solve equations using algebra, how to model the world using functions, and some important facts about geometry. The material in this chapter is required for your understanding of the more advanced topics in this book.

## 1.1 Solving equations

Most math skills boil down to being able to manipulate and solve equations. Solving an equation means finding the value of the unknown in the equation.

Check this shit out:

$$x^2 - 4 = 45.$$

To solve the above equation is to answer the question "What is $x$?" More precisely, we want to find the number that can take the place of $x$ in the equation so that the equality holds. In other words, we're asking,

"Which number times itself minus four gives 45?"

That is quite a mouthful, don't you think? To remedy this verbosity, mathematicians often use specialized symbols to describe math operations. The problem is that these specialized symbols can be very confusing. Sometimes even the simplest math concepts are inaccessible if you don't know what the symbols mean.

What are your feelings about math, dear reader? Are you afraid of it? Do you have anxiety attacks because you think it will be too difficult for you? Chill! Relax, my brothers and sisters. There's nothing to it. Nobody can magically guess the solution to an equation immediately. To find the solution, you must break the problem into simpler steps. Let's walk through this one together.

To find $x$, we can manipulate the original equation, transforming it into a different equation (as true as the first) that looks like this:

$$x = \text{only numbers}.$$

That's what it means to *solve* an equation: the equation is solved because the unknown is isolated on one side, while the constants are grouped on the other side. You can type the numbers on the right-hand side into a calculator and obtain the numerical value of $x$.

By the way, before we continue our discussion, let it be noted: the equality symbol ($=$) means that all that is to the left of $=$ is equal to all that is to the right of $=$. To keep this equality statement true, **for every change you apply to the left side of the equation, you must apply the same change to the right side of the equation**.

To find $x$, we need to manipulate the original equation into its final form, simplifying it step by step until it can't be simplified any further. The only requirement is that the manipulations we make transform one true equation into another true equation. In this example, the first simplifying step is to add the number four to both sides of the equation:

$$x^2 - 4 \ + 4 = 45 \ + 4,$$

which simplifies to

$$x^2 = 49.$$

Now the expression looks simpler, yes? How did I know to perform this operation? I wanted to "undo" the effects of the operation $-4$. We undo an operation by applying its *inverse*. In the case where the operation is the subtraction of some amount, the inverse operation is the addition of the same amount. We'll learn more about function inverses in Section 1.4.

We're getting closer to our goal of *isolating* $x$ on one side of the equation, leaving only numbers on the other side. The next step is to undo the square $x^2$ operation. The inverse operation of squaring a number $x^2$ is to take its square root $\sqrt{\ }$, so that's what we'll do next. We obtain

$$\sqrt{x^2} = \sqrt{49}.$$

Notice how we applied the square root to both sides of the equation? If we don't apply the same operation to both sides, we'll break the equality!

The equation $\sqrt{x^2} = \sqrt{49}$ simplifies to

$$|x| = 7.$$

What's up with the vertical bars around $x$? The notation $|x|$ stands for the *absolute value* of $x$, which is the same as $x$ except we ignore the sign that indicates whether $x$ is positive or negative. For example $|5| = 5$ and $|-5| = 5$, too. The equation $|x| = 7$ indicates that both $x = 7$ and $x = -7$ satisfy the equation $x^2 = 49$. Seven squared is 49, $7^2 = 49$, and negative seven squared is also 49, $(-7)^2 = 49$, because the two negative signs cancel each other out.

The final solutions to the equation $x^2 - 4 = 45$ are

$$x = 7 \qquad \text{and} \qquad x = -7.$$

Yes, there are *two* possible answers. You can check that both of the above values of $x$ satisfy the initial equation $x^2 - 4 = 45$.

If you are comfortable with all the notions of high school math and you feel you could have solved the equation $x^2 - 4 = 45$ on your own, then you can skim through this chapter quickly. If on the other hand you are wondering how the squiggle killed the power two, then this chapter is for you! In the following sections we will review all the essential concepts from high school math that you will need to power through the rest of this book. First, let me tell you about the different kinds of numbers.

## 1.2 Numbers

In the beginning, we must define the main players in the world of math: numbers.

### Definitions

Numbers are the basic objects we use to count, measure, quantify, and calculate things. Mathematicians like to classify the different kinds of number-like objects into categories called *sets*:

- The natural numbers: $\mathbb{N} = \{0, 1, 2, 3, 4, 5, 6, 7, \ldots\}$
- The integers: $\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$
- The rational numbers: $\mathbb{Q} = \{\frac{5}{3}, \frac{22}{7}, 1.5, 0.125, -7, \ldots\}$
- The real numbers: $\mathbb{R} = \{-1, 0, 1, \sqrt{2}, e, \pi,\ 4.94\ldots,\ \ldots\}$
- The complex numbers: $\mathbb{C} = \{-1, 0, 1, i, 1+i, 2+3i, \ldots\}$

These categories of numbers should be somewhat familiar to you. Think of them as neat classification labels for everything that you would normally call a number. Each group in the above list is a *set*. A set is a collection of items of the same kind. Each collection has a name and a precise definition for which items belong in that collection. Note also that each of the sets in the list contains all the sets above it, as illustrated in Figure 1.2. For now, we don't need to go into the details of sets and set notation, but we do need to be aware of the different sets of numbers.

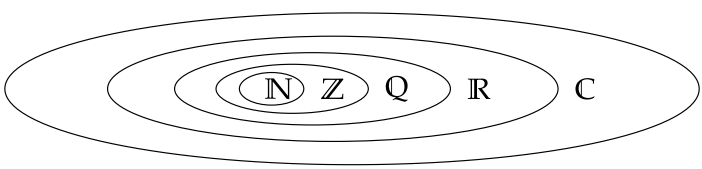



**Figure 1.2:** An illustration of the nested containment structure of the different number sets. The set of natural numbers is contained in the set of integers, which in turn is contained in the set of rational numbers. The set of rational numbers is contained in the set of real numbers, which is contained in the set of complex numbers.

Why do we need so many different sets of numbers? Each set of numbers is associated with more and more advanced mathematical problems.

The simplest numbers are the natural numbers $\mathbb{N}$, which are sufficient for all your math needs if all you're going to do is *count* things. How many goats? Five goats here and six goats there so the total is number 2 corresponds to a line segment of length two, which is a geometric object in a one-dimensional space. If we add a line segment of length two in a second dimension, we obtain a square with area $2^2$ in a two-dimensional space. Adding a third dimension, we obtain a cube with volume $2^3$ in a three-dimensional space. Indeed, raising a base $a$ to the exponent 2 is commonly called "$a$ squared," and raising $a$ to the power of 3 is called "$a$ cubed."

The geometrical analogy about one-dimensional quantities as lengths, two-dimensional quantities as areas, and three-dimensional quantities as volumes is good to keep in mind.

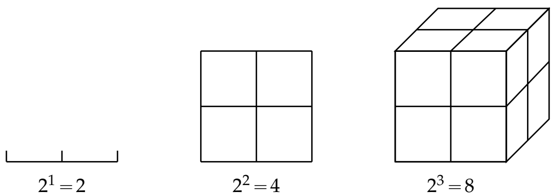



**Figure 1.5:** Geometric interpretation for exponents 1, 2, and 3. A length raised to exponent 2 corresponds to the area of a square. The same length raised to exponent 3 corresponds to the volume of a cube.

Our visual intuition works very well up to three dimensions, but we can use other means of visualizing higher exponents, as demonstrated in Figure 1.6.

## Operator precedence

There is a standard convention for the order in which mathematical operations must be performed. The basic algebra operations have the following precedence:

1. Parentheses
2. Exponents
3. Multiplication and Division
4. Addition and Subtraction

If you're seeing this list for the first time, the acronym PEMDAS and the associated mnemonic "Please Excuse My Dear Aunt Sally," might help you remember the order of operations.

For instance, the expression $5 \cdot 3^2 + 13$ is interpreted as "First find the square of 3, then multiply it by 5, and then add 13." Parentheses are needed to carry out the operations in a different order: to multiply 5 times 3 first and *then* take the square, the equation should read

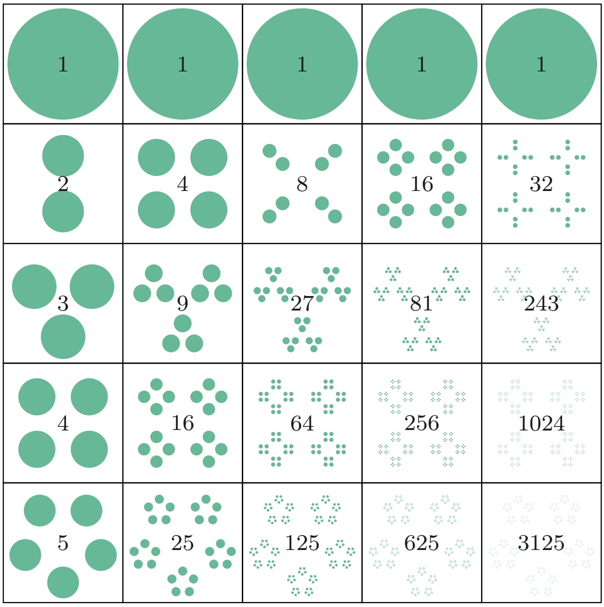



**Figure 1.6:** Visualization of numbers raised to different exponents. Each box in this grid contains $a^n$ dots, where the base $a$ varies from one through five, and the exponent $n$ varies from one through five. In the first row we see that the number $a = 1$ raised to any exponent is equal to itself. The second row corresponds to the base $a = 2$ so the number of dots doubles each time we increase the exponent by one. Starting from $2^1 = 2$ in the first column, we end up with $2^5 = 32$ in the last column. The rest of the rows show how exponentiation works for different bases.

$(5 \cdot 3)^2 + 13$, where parentheses indicate that the square acts on $(5 \cdot 3)$ as a whole and not on 3 alone.

## Exercises

**E1.1** Solve for the unknown $x$ in the following equations:

**a)** $3x + 2 - 5 = 4 + 2$ **b)** $\frac{1}{2}x - 3 = \sqrt{3} + 12 - \sqrt{3}$

**c)** $\frac{7x-4}{2} + 1 = 8 - 2$ **d)** $5x - 2 + 3 = 3x - 5$

**E1.2** Indicate all the number sets the following numbers belong to.

**a)** $-2$ **b)** $\sqrt{-3}$ **c)** $8 \div 4$ **d)** $\frac{5}{3}$ **e)** $\frac{\pi}{2}$

**E1.3** Calculate the values of the following expressions:

**a)** $2^3 3 - 3$ **b)** $2^3(3 - 3)$ **c)** $\frac{4-2}{3^3}(6 \cdot 7 - 41)$

A *perfect square* is a quadratic expression that can be written as the product of repeated factors $(x+p)$:

$$x^2 + 2px + p^2 = (x+p)(x+p) = (x+p)^2.$$

Note $x^2 - 2qx + q^2 = (x-q)^2$ is also a perfect square.

## Completing the square

In this section we'll learn about an ancient algebra technique called *completing the square*, which allows us to rewrite *any* quadratic expression of the form $x^2 + Bx + C$ as a perfect square plus some constant correction factor $(x+p)^2 + k$. This algebra technique was described in one of the first books on *al-jabr* (algebra), written by Al-Khwarizmi around the year 800 CE. The name "completing the square" comes from the ingenious geometric construction used by this procedure. Yes, we can use geometry to solve algebra problems!

We assume the starting point for the procedure is a quadratic expression whose quadratic coefficient is one, $1x^2 + Bx + C$, and use capital letters $B$ and $C$ to denote the linear and constant coefficients. The capital letters are to avoid any confusion with the quadratic expression $ax^2 + bx + c$, for which $a \neq 1$. Note we can always write $ax^2 + bx + c$ as $a(x^2 + \frac{b}{a}x + \frac{c}{a})$ and apply the procedure to the expression inside the brackets, identifying $\frac{b}{a}$ with $B$ and $\frac{c}{a}$ with $C$.

First let's rewrite the quadratic expression $x^2 + Bx + C$ by splitting the linear term into two equal parts:

$$x^2 + \tfrac{B}{2}x + \tfrac{B}{2}x + C.$$

We can interpret the first three terms geometrically as follows: the $x^2$ term corresponds to a square with side length $x$, while the two $\frac{B}{2}x$ terms correspond to rectangles with sides $\frac{B}{2}$ and $x$. See the left side of Figure 1.8 for an illustration.

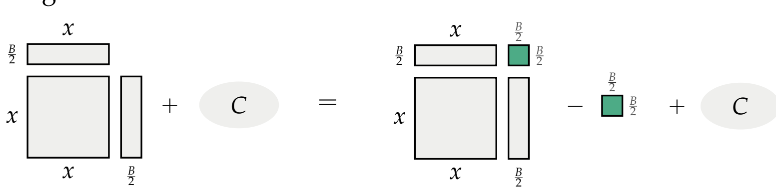



**Figure 1.8:** To complete the square in the expression $x^2 + Bx + C$, we need to add the quantity $(\frac{B}{2})^2$, which corresponds to a square (shown in darker colour) with sides equal to half the coefficient of the linear term. We also subtract $(\frac{B}{2})^2$ so the overall value of the expression remains unchanged.

The square with area $x^2$ and the two rectangles can be positioned to form a larger square with side length $\left(x + \frac{B}{2}\right)$. Note there's a small

**E1.9** Solve for $x$ in the equation $x^4 - 4x^2 + 4 = 0$.
Hint: Use the substitution $y = x^2$.

## 1.7 The Cartesian plane

The Cartesian plane, named after famous philosopher and mathematician René Descartes, is used to visualize pairs of numbers $(x, y)$.

Consider first the *number line* representation for numbers.

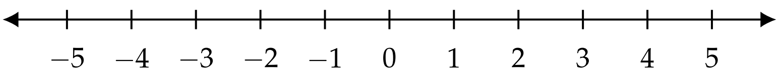



**Figure 1.9:** Every real number $x$ corresponds to a point on the number line. The number line extends indefinitely to the left (toward negative infinity) and to the right (toward positive infinity).

The Cartesian plane is the two-dimensional generalization of the number line. Generally, we call the plane's horizontal axis "the $x$-axis" and its vertical axis "the $y$-axis." We put notches at regular intervals on each axis so we can measure distances.

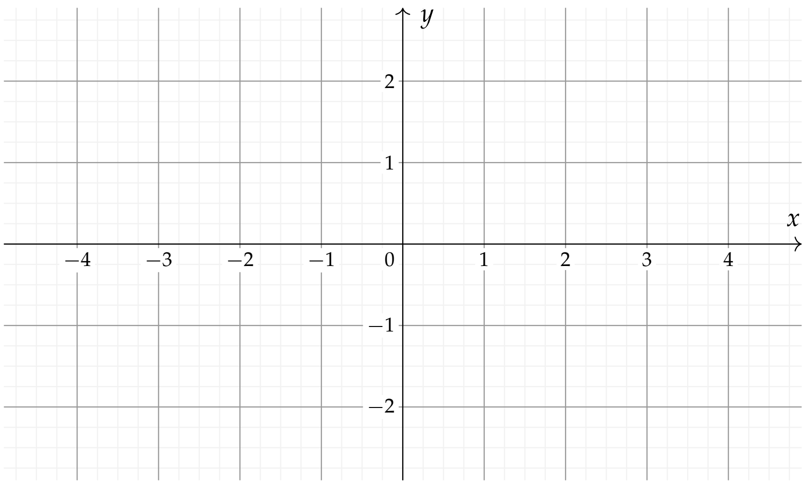



**Figure 1.10:** Every point in the Cartesian plane corresponds to a pair of real numbers $(x, y)$. Points $P = (P_x, P_y)$, vectors $\vec{v} = (v_x, v_y)$, and graphs of functions $(x, f(x))$ live here.

Figure 1.10 is an example of an empty Cartesian coordinate system. Think of the coordinate system as an empty canvas. What can you draw on this canvas?

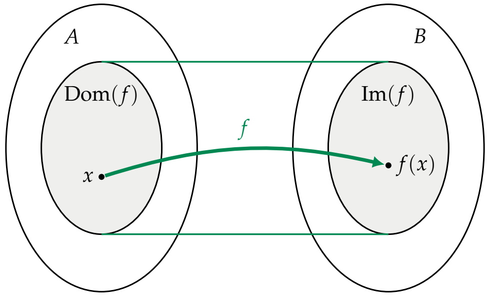



**Figure 1.14:** Illustration of the input and output sets of a function $f\colon A \to B$. The *source set* is denoted $A$ and the *domain* is denoted $\mathrm{Dom}(f)$. Note that the function's domain is a subset of its source set. The *target set* is denoted $B$ and the *image* is denoted $\mathrm{Im}(f)$. The image is a subset of the target set.

The complicated-looking expression between the curly brackets uses *set notation* to define the set of nonnegative numbers $\mathbb{R}_+$. In words, the expression $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geqslant 0\}$ states that "$\mathbb{R}_+$ is defined as the set of all real numbers $x$ such that $x$ is greater than or equal to zero." We'll discuss set notation in more detail in Section 1.16. For now, you can just remember that $\mathbb{R}_+$ represents the set of nonnegative real numbers.

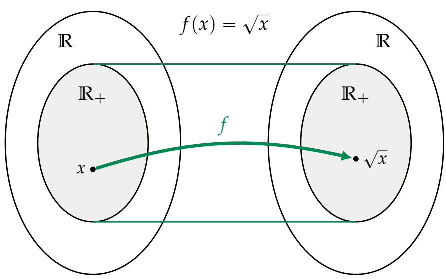



**Figure 1.15:** The input and output sets of the function $f(x) = \sqrt{x}$. The domain of $f$ is the set of nonnegative real numbers $\mathbb{R}_+$ and its image is $\mathbb{R}_+$.

To illustrate the difference between the image of a function and its target set, let's look at the function $f(x) = x^2$ shown in Figure 1.16. The quadratic function is of the form $f\colon \mathbb{R} \to \mathbb{R}$. The function's source set is $\mathbb{R}$ (it takes real numbers as inputs) and its target set is $\mathbb{R}$ (the outputs are real numbers too); however, not all real numbers are possible outputs. The *image* of the function $f(x) = x^2$ consists only of the nonnegative real numbers $\mathbb{R}_+ = \{y \in \mathbb{R} \mid y \geqslant 0\}$, since $f(x) \geqslant 0$ for all $x$.

## 1.10 Geometry

The word "geometry" comes from the Greek roots *geo*, which means "earth," and *metron*, which means "measurement." This name is linked to one of the early applications of geometry, which was to measure the total amount of land contained within a certain boundary region. Over the years, the study of geometry evolved to be more abstract. Instead of developing formulas for calculating the area of specific regions of land, mathematicians developed general area formulas that apply to *all* regions that have a particular shape.

In this section we'll present formulas for calculating the perimeters, areas, and volumes for various shapes (also called "figures") commonly encountered in the real world. For two-dimensional figures, the main quantities of interest are the figures' areas and the figures' perimeters (the length of the walk around the figure). For three-dimensional figures, the quantities of interest are the surface area (how much paint it would take to cover all sides of the figure), and volume (how much water it would take to fill a container of this shape). The formulas presented are by no means an exhaustive list of everything there is to know about geometry, but they represent a core set of facts that you want to add to your toolbox.

### Triangles

The area of a triangle is equal to $\frac{1}{2}$ times the length of its base times its height:

$$A = \tfrac{1}{2} a h_a.$$

Note that $h_a$ is the height of the triangle *relative to* the side $a$.

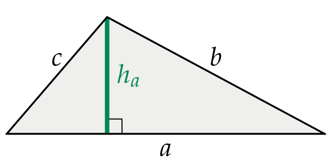



**Figure 1.33:** A triangle with side lengths $a$, $b$, and $c$. The height of the triangle with respect to the side $a$ is denoted $h_a$.

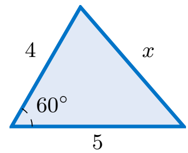
The perimeter of a triangle is given by the sum of its side lengths:

$$P = a + b + c.$$

**Interior angles of a triangle rule** The sum of the inner angles in any triangle is equal to $180^\circ$. Consider a triangle with internal angles $\alpha$, $\beta$ and $\gamma$ as shown in Figure 1.34. We may not know the values of

## Exercises

**E1.13** Find the length of side $x$ in the triangle below.




Hint: Use the cosine rule.

**E1.14** Find the volume and the surface area of a sphere with radius 2.

**E1.15** On a rainy day, Laura brings her bike indoors, and the wet bicycle tires leave a track of water on the floor. What is the length of the water track left by the bike's rear tire (diameter 73 cm) if the wheel makes five full turns along the floor?

# 1.11 Trigonometry

If one of the angles in a triangle is equal to $90^\circ$, we call this triangle a *right-angle triangle*. In this section we'll discuss right-angle triangles in great detail and get to know their properties. We'll learn some fancy new terms like *hypotenuse*, *opposite*, and *adjacent*, which are used to refer to the different sides of a triangle. We'll also use the functions *sine*, *cosine*, and *tangent* to compute the *ratios of lengths* in right triangles.

Understanding triangles and their associated trigonometric functions is of fundamental importance: you'll need this knowledge for your future understanding of mathematical concepts like vectors and complex numbers.

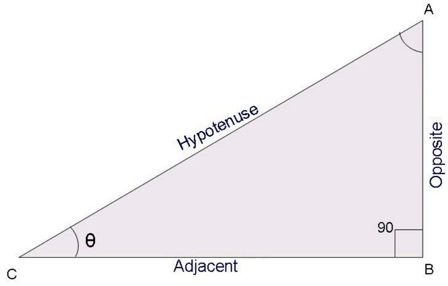



**Figure 1.40:** A right-angle triangle. The angle at the base is denoted $\theta$ and the names of the sides of the triangle are indicated.

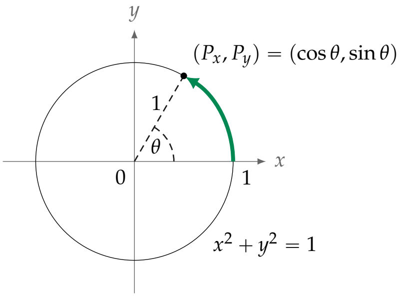



**Figure 1.41:** The unit circle corresponds to the equation $x^2 + y^2 = 1$. The coordinates of the point $P$ on the unit circle are $P_x = \cos\theta$ and $P_y = \sin\theta$.

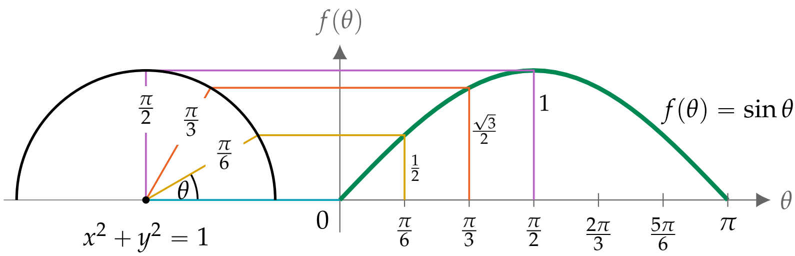



**Figure 1.42:** The function $f(\theta) = \sin\theta$ describes the vertical position of a point $P$ that travels along the unit circle. The graph shows the values of the function $f(\theta) = \sin\theta$ for angles between $\theta = 0$ and $\theta = \pi$.

Figure 1.42 shows the graph of the function $f(\theta) = \sin\theta$. The values $\sin\theta$ for the angles 0, $\frac{\pi}{6}$ (30°), $\frac{\pi}{3}$ (60°), and $\frac{\pi}{2}$ (90°) are marked. There are three values to remember: $\sin\theta = 0$ when $\theta = 0$, $\sin\theta = \frac{1}{2}$ when $\theta = \frac{\pi}{6}$ (30°), and $\sin\theta = 1$ when $\theta = \frac{\pi}{2}$ (90°). See Figure 1.26 (page 60) for a graph of $\sin\theta$ that shows a complete cycle around the circle. Also see Figure 1.29 (page 62) for the graph of $\cos\theta$.

Instead of trying to memorize the values of the functions $\cos\theta$ and $\sin\theta$ separately, it's easier to remember them as a combined "package" $(\cos\theta, \sin\theta)$, which describes the $x$- and $y$-coordinates of the point $P$ for the angle $\theta$. Figure 1.43 shows the values of $\cos\theta$ and $\sin\theta$ for the angles 0, $\frac{\pi}{6}$ (30°), $\frac{\pi}{4}$ (45°), $\frac{\pi}{3}$ (60°), and $\frac{\pi}{2}$ (90°). These are the most common angles that often show up on homework and exam questions. For each angle, the $x$-coordinate (the first number in the bracket) is $\cos\theta$, and the $y$-coordinate (the second number in the bracket) is $\sin\theta$.

## Vector as arrows

So far, we described how to perform algebraic operations on vectors in terms of their components. Vector operations can also be interpreted geometrically, as operations on arrows in the Cartesian plane.

**Vector addition** The sum of two vectors corresponds to the combined displacement of the two vectors. Figure 1.49 illustrates the addition of two vectors, $\vec{v}_1 = (3,0)$ and $\vec{v}_2 = (2,2)$. The sum of the two vectors is the vector $\vec{v}_1 + \vec{v}_2 = (3,0) + (2,2) = (5,2)$.

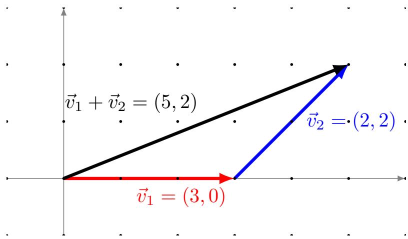



**Figure 1.49:** The addition of the vectors $\vec{v}_1$ and $\vec{v}_2$ produces the vector $(5,2)$.

**Vector subtraction** Before we describe vector subtraction, note that multiplying a vector by a scale factor $\alpha = -1$ gives a vector of the same length as the original, but pointing in the opposite direction.

This fact is useful if you want to subtract two vectors using the graphical approach. Subtracting a vector is the same as adding the negative of the vector:

$$\vec{w} - \vec{v}_1 = \vec{w} + (-\vec{v}_1) = \vec{v}_2.$$

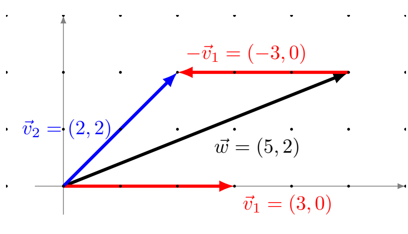



**Figure 1.50:** The vector subtraction $\vec{w} - \vec{v}_1$ is equivalent to the vector addition $\vec{w} + (-\vec{v}_1)$, where $(-\vec{v}_1)$ is like $\vec{v}_1$ but points in the opposite direction.

Figure 1.50 illustrates the graphical procedure for subtracting the vector $\vec{v}_1 = (3,0)$ from the vector $\vec{w} = (5,2)$. Subtraction of $\vec{v}_1 = (3,0)$ is the same as addition of $-\vec{v}_1 = (-3,0)$.

**Vectors and vector coordinates**

One final point we need to clarify is the difference between real-world vector quantities like the velocity of a tennis ball $\vec{v}$ and its mathematical representation as a coordinate vector $(v_x, v_y, v_z)$. If you know the coordinate vector $(v_x, v_y, v_z)$ then you know what the real-world velocity is, right? Not quite.

Let's say you're doing a physics research project on tennis serves. You define an $xyz$-coordinate system for the tennis court, which allows you to represent the ball's velocity $\vec{v}$ as a triple of components $(v_x, v_y, v_z)$ interpreted as: "The ball is moving with velocity $v_x$ units in the $x$-direction, $v_y$ units in the $y$-direction, and $v_z$ units in the $z$-direction."

Suppose you want to describe the velocity vector $\vec{v}$ to a fellow physicist via text message. Referring to your sheet of calculations, you find the values $\vec{v} = (60, 3, -2)$, which you know were measured in metres per second. You send this message:

```
The velocity is (60,3,-2) measured in metres per second.
```

A few minutes later the following reply comes back:

```
Wait whaaat? What coordinate system are you using?
```

Indeed the information you sent is incomplete. Vector components depend on the coordinate system in which the vectors are represented. The triple of numbers $(60, 3, -2)$ only makes sense once you know the directions of the axes in the $xyz$-coordinate system. Realizing your mistake, you send a text with all the required information:

```
Using the coordinate system centred at the south post of
the net, with the x-axis pointing east along the court,
the y-axis pointing north along the net, and the z-axis
pointing up, the velocity is (60,3,-2) in metres per second.
```

A few seconds later, you get the reply:

```
OK got it now. Thx!
```

This hypothetical situation illustrates the importance of the coordinate systems for describing vectors. If you don't know what the coordinate system is, knowing the coordinates $(v_x, v_y, v_z)$ doesn't tell you much. Only when you know the directions of the unit vectors $\hat{\imath}$, $\hat{\jmath}$, and $\hat{k}$ can you interpret the instructions $\vec{v} = v_x\hat{\imath} + v_y\hat{\jmath} + v_z\hat{k}$.

It turns out, using the $xyz$-coordinate system with the three vectors $\{\hat{\imath}, \hat{\jmath}, \hat{k}\}$ is just one of many possible ways we can represent vectors. We can represent a vector $\vec{v}$ as coordinates $(v_1, v_2, v_3)$ with respect to any *basis* $\{\hat{e}_1, \hat{e}_2, \hat{e}_3\}$ using the expression $\vec{v} = v_1\hat{e}_1 + v_2\hat{e}_2 +$

$v_3\hat{e}_3$, which corresponds to the instructions: "Move $v_1$ units in the direction of $\hat{e}_1$, move $v_2$ units in the direction of $\hat{e}_2$, and move $v_3$ units in the direction of $\hat{e}_3$."

What's a basis, you ask? I'm glad you asked, because this is the subject of the next section.

## Basis

One of the most important concepts in the study of vectors is the concept of a *basis*. Consider the three-dimensional vector space $\mathbb{R}^3$. A *basis* for $\mathbb{R}^3$ is a set of vectors $\{\hat{e}_1, \hat{e}_2, \hat{e}_3\}$ that can be used as a coordinate system for $\mathbb{R}^3$. If the set of vectors $\{\hat{e}_1, \hat{e}_2, \hat{e}_3\}$ is a basis, then you can *represent* any vector $\vec{v} \in \mathbb{R}^3$ as coordinates $(v_1, v_2, v_3)$ *with respect to* that basis:

$$\vec{v} = v_1\hat{e}_1 + v_2\hat{e}_2 + v_3\hat{e}_3.$$

The vector $\vec{v}$ is obtained by measuring out a distance $v_1$ in the $\hat{e}_1$ direction, a distance $v_2$ in the $\hat{e}_2$ direction, and a distance $v_3$ in the $\hat{e}_3$ direction.

You are already familiar with the *standard* basis $\{\hat{\imath}, \hat{\jmath}, \hat{k}\}$, which is associated with the $xyz$-coordinate system. You know that any vector $\vec{v} \in \mathbb{R}^3$ can be expressed as a triple $(v_x, v_y, v_z)$ with respect to the basis $\{\hat{\imath}, \hat{\jmath}, \hat{k}\}$ through the formula $\vec{v} = v_x\hat{\imath} + v_y\hat{\jmath} + v_z\hat{k}$. The whole point of this section is to let you know that other bases (coordinate systems) exist, and to get you into the habit of asking, "With respect to which coordinate system?" every time you see a coordinate vector $(a, b, c)$.

### An analogy

Let's start with a simple example of a basis. If you look at the HTML source code behind any web page, you're sure to find at least one mention of the colour stylesheet directive such as `color:#336699;`. The numbers should be interpreted as a triple of values $(33, 66, 99)$, each value describing the amount of red, green, and blue needed to create a given colour. Let us call the colour described by the triple $(33, 66, 99)$ CoolBlue. This convention for colour representation is called the RGB colour model and we can think of it as the *RGB basis*. A basis is a set of elements that can be combined together to express something more complicated. In our case, the **R**, **G**, and **B** elements are pure colours that can create any colour when mixed appropriately. Schematically, we can write this mixing idea as

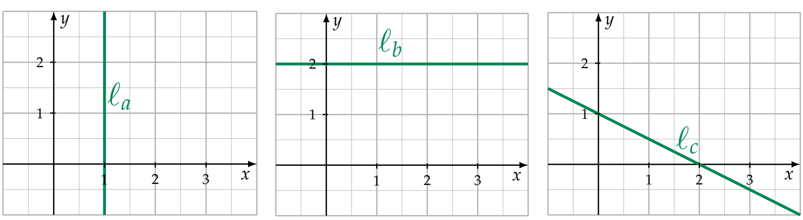
$$\text{CoolBlue} = (33, 66, 99)_{RGB} = 33\mathbf{R} + 66\mathbf{G} + 99\mathbf{B},$$

## Geometric solution

Solving a system of two linear equations in two unknowns can be understood geometrically as finding the point of intersection between two lines in the Cartesian plane. In this section we'll explore this correspondence between algebra and geometry to develop yet another way of solving systems of linear equations.

The algebraic equation $ax + by = c$ containing the unknowns $x$ and $y$ can be interpreted as a *constraint* equation on the set of possible values for the variables $x$ and $y$. We can visualize this constraint geometrically by considering the coordinate pairs $(x, y)$ that lie in the Cartesian plane. Recall that every point in the Cartesian plane can be represented as a coordinate pair $(x, y)$, where $x$ and $y$ are the coordinates of the point.

Figure 1.59 shows the geometrical representation of three equations. The line $\ell_a$ corresponds to the set of points $(x, y)$ that satisfy the equation $x = 1$, the line $\ell_b$ is the set of points $(x, y)$ that satisfy the equation $y = 2$, and the line $\ell_c$ corresponds to the set of points that satisfy $x + 2y = 2$.




(a) $x = 1$ (b) $y = 2$ (c) $x + 2y = 2$

**Figure 1.59:** Graphical representations of three linear equations.

You can convince yourself that the geometric lines shown in Figure 1.59 are equivalent to the algebraic equations by considering individual points $(x, y)$ in the plane. For example, the points $(1, 0)$, $(1, 1)$, and $(1, 2)$ are all part of the line $\ell_a$ since they satisfy the equation $x = 1$. For the line $\ell_c$, you can verify that the line's $x$-intercept $(2, 0)$ and its $y$-intercept $(0, 1)$ both satisfy the equation $x + 2y = 2$.

The Cartesian plane as a whole corresponds to the set $\mathbb{R}^2$, which describes all possible pairs of coordinates. To understand the equivalence between the algebraic equation $ax + by = c$ and the line $\ell$ in the Cartesian plane, we can use the following precise math notation:

$$\ell : \{(x, y) \in \mathbb{R}^2 \mid ax + by = c\}.$$

In words, this means that the line $\ell$ is defined as the subset of the pairs of real numbers $(x, y)$ that satisfy the equation $ax + by = c$.

Figure 1.60 shows the graphical representation of the line $\ell$.

You don't have to take my word for it, though! Think about it and convince yourself that all points on the line $\ell$ shown in Figure 1.60 satisfy the equation $ax + by = c$. For example, you can check that the $x$-intercept $(\frac{c}{a}, 0)$ and the $y$-intercept $(0, \frac{c}{b})$ satisfy the equation $ax + by = c$.

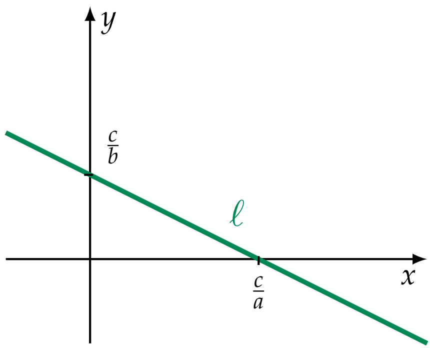



**Figure 1.60:** Graphical representation of the equation $ax + by = c$.

Solving the system of two equations

$$\begin{aligned} a_1x + b_1y &= c_1, \\ a_2x + b_2y &= c_2, \end{aligned}$$

corresponds to finding the intersection of the lines $\ell_1$ and $\ell_2$ that represent each equation. The pair $(x, y)$ that satisfies both algebraic equations simultaneously is equivalent to the point $(x, y)$ that is the intersection of lines $\ell_1$ and $\ell_2$, as illustrated in Figure 1.61.

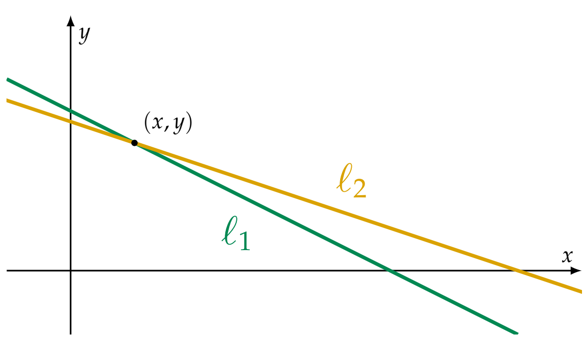



**Figure 1.61:** The point $(x, y)$ that lies at the intersection of lines $\ell_1$ and $\ell_2$.

**Example** Let's see how we can use the geometric interpretation to solve the system of equations

$$\begin{aligned} x + 2y &= 5, \\ 3x + 9y &= 21. \end{aligned}$$

We've already seen three different *algebraic* techniques for finding the solution to this system of equations; now let's see a *geometric* approach for finding the solution. I'm not kidding you, we're going to solve the exact same system of equations a fourth time!

The first step is to draw the lines that correspond to each of the equations using pen and paper or a graphing calculator. The second step is to find the coordinates of the point where the two lines intersect as shown in Figure 1.62. The point $(1, 2)$ that lies on both lines $\ell_1$ and $\ell_2$ corresponds to the $x$ and $y$ values that satisfy both equations simultaneously.

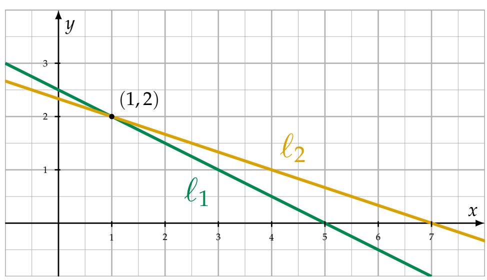



**Figure 1.62:** The line $\ell_1$ with equations $x + 2y = 5$ intersects the line $\ell_2$ with equation $3x + 9y = 21$ at the point $(1, 2)$.

Visit the webpage at `www.desmos.com/calculator/exikik615f` to play with an interactive version of the graphs shown in Figure 1.62. Try changing the equations and see how the graphs change.

## Exercises

**E1.23** Plot the lines $\ell_a$, $\ell_b$, and $\ell_c$ shown in Figure 1.59 (page 106) using the Desmos graphing calculator. Use the graphical representation of these lines to find: **a)** the intersection of lines $\ell_c$ and $\ell_a$, **b)** the intersection of $\ell_a$ and $\ell_b$, and **c)** the intersection of lines $\ell_b$ and $\ell_c$.

**E1.24** Solve the system of equations simultaneously for $x$ and $y$:

$$\begin{aligned} 2x + 4y &= 16, \\ 5x - y &= 7. \end{aligned}$$

**E1.25** Solve the system of equations for the unknowns $x$, $y$, and $z$:

$$\begin{aligned} 2x + y - 4z &= 28, \\ x + y + z &= 8, \\ 2x - y - 6z &= 22. \end{aligned}$$

**E1.26** Solve for $p$ and $q$ given the equations $p + q = 10$ and $p - q = 4$.

## 1.16 Set notation

A *set* is the mathematically precise notion for describing a group of objects. You don't need to know about sets to perform simple math; but more advanced topics require an understanding of what sets are and how to denote set membership, set operations, and set containment relations. This section introduces all the relevant concepts.

### Definitions

- *set*: a collection of mathematical objects
- $S, T$: the usual variable names for sets
- $s \in S$: this statement is read "$s$ is an element of $S$" or "$s$ is in $S$"
- $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$: some important number sets: the naturals, the integers, the rationals, and the real numbers, respectively.
- $\varnothing$: the *empty set* is a set that contains no elements
- $\{ \ldots\ldots \}$: the curly brackets are used to define sets, and the expression inside the curly brackets describes the set contents.

Set operations:

- $S \cup T$: the *union* of two sets. The union of $S$ and $T$ corresponds to the elements in either $S$ or $T$.
- $S \cap T$: the *intersection* of the two sets. The intersection of $S$ and $T$ corresponds to the elements that are in both $S$ and $T$.
- $S \backslash T$: *set difference* or *set minus*. The set difference $S \backslash T$ corresponds to the elements of $S$ that are not in $T$.

Set relations:

- $\subset$: is a strict subset of
- $\subseteq$: is a subset of or equal to

## 1.17 Math problems

We've now reached the first section of problems in this book. The purpose of these problems is to give you a way to comprehensively practice your math fundamentals. Knowing how to solve math problems is a very useful skill to develop. At times, honing your math chops might seem like tough mental work, but at the end of each problem, you'll gain a stronger foothold on all the topics you've been learning about. You'll also experience a small *achievement buzz* after each problem you vanquish.

Sit down and take a crack at these practice problems today, or another time when you're properly caffeinated. If you make time for some math practice, you'll develop long-lasting comprehension and true math fluency.

Without solving any problems, you're likely to forget most of what you've learned in the next few months. You might still remember the big ideas, but the details will be fuzzy and faded. By solving some of the practice problems, you'll remember a lot more stuff. Don't break the pace now: with math, it's very much *use it or lose it*!

Make sure you put your phone away while you're working on the problems. You don't need fancy technology to do math; grab a pen and some paper from the printer and you'll be fine. The great mathematicians like Descartes, Hilbert, Leibniz, and Noether did most of their work with pen and paper and they did well. Spend some time with math the way they did.

**P1.1** Solve for $x$ in the equation $x^2 - 9 = 7$.

**P1.2** Solve for $x$ in the equation $\cos^{-1}\left(\frac{x}{A}\right) - \phi = \omega t$.

**P1.3** Solve for $x$ in the equation $\frac{1}{x} = \frac{1}{a} + \frac{1}{b}$.

**P1.4** Use a calculator to find the values of the following expressions:

**a)** $\sqrt[4]{3^3}$ **b)** $2^{10}$ **c)** $7^{\frac{1}{4}} - 10$ **d)** $\frac{1}{2}\ln(e^{22})$

**P1.5** Compute the following expressions involving fractions:

**a)** $\dfrac{1}{2} + \dfrac{1}{4}$ **b)** $\dfrac{4}{7} - \dfrac{23}{5}$ **c)** $1\frac{3}{4} + 1\frac{31}{32}$

**P1.6** Use the basic rules of algebra to simplify the following expressions:

**a)** $ab\,\dfrac{1}{a}\,b^2cb^{-3}$ **b)** $\dfrac{abc}{bca}$ **c)** $\dfrac{27a^2}{\sqrt{9abba}}$

**d)** $\dfrac{a(b+c)-ca}{b}$ **e)** $\dfrac{a}{c\sqrt[3]{b}}\,\dfrac{b^{\frac{4}{3}}}{a^2}$ **f)** $(x+a)(x+b) - x(a+b)$

# Chapter 2

# Intro to linear algebra

The first chapter reviewed core ideas of mathematics. Now that we're done with the prerequisites, we can begin the main discussion of linear algebra: the study of vectors and matrices.

## 2.1 Definitions

Vectors and matrices are the objects of study in linear algebra, and in this chapter we'll define them and learn the basic operations we can perform on them.

We denote the set of $n$-dimensional vectors with real coefficients as $\mathbb{R}^n$. A vector $\vec{v} \in \mathbb{R}^n$ is an $n$-tuple of real numbers.[1] For example, a three-dimensional vector is defined as a triple of numbers:

$$\vec{v} \stackrel{\text{def}}{=} (v_1, v_2, v_3).$$

To specify the vector $\vec{v}$, we must specify the values for its three components, $v_1$, $v_2$, and $v_3$. We'll use the terms *components* and *coordinates* interchangeably throughout the book.

A matrix $A \in \mathbb{R}^{m \times n}$ is a rectangular array of real numbers with $m$ rows and $n$ columns. For example, a $3 \times 2$ matrix is defined like this:

$$A \stackrel{\text{def}}{=} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix}.$$

To specify the matrix $A$, we need to specify the values of its six *entries*: $a_{11}$, $a_{12}$, $a_{21}$, $a_{22}$, $a_{31}$, and $a_{32}$.

[1] The notation "$s \in S$" is read "$s$ is an element of $S$" or "$s$ in $S$."

In the remainder of this chapter we'll learn about the mathematical operations we can perform on vectors and matrices. Many problems in science, business, and technology can be described in terms of vectors and matrices, so it's important you understand how to work with these math objects.

## Context

To illustrate what's new about vectors and matrices, let's begin by reviewing the properties of something more familiar: the set of real numbers $\mathbb{R}$. The basic operations for real numbers are:

- Addition (denoted $+$)
- Subtraction, the inverse of addition (denoted $-$)
- Multiplication (denoted implicitly)
- Division, the inverse of multiplication (denoted by fractions)

You're familiar with these operations and know how to use them to evaluate math expressions and solve equations.

You should also be familiar with *functions* that take real numbers as inputs and give real numbers as outputs, denoted $f : \mathbb{R} \to \mathbb{R}$. Recall that, by definition, the *inverse function* $f^{-1}$ *undoes* the effect of $f$. If you are given $f(x)$ and want to find $x$, you can use the inverse function as follows: $f^{-1}(f(x)) = x$. For example, the function $f(x) = \ln(x)$ has the inverse $f^{-1}(x) = e^x$, and the inverse of $g(x) = \sqrt{x}$ is $g^{-1}(x) = x^2$.

Having reviewed the basic operations for real numbers $\mathbb{R}$, let's now introduce the basic operations for vectors $\mathbb{R}^n$ and matrices $\mathbb{R}^{m\times n}$.

### Vector operations

The operations we can perform on vectors are:

- Addition (denoted $+$)
- Subtraction, the inverse of addition (denoted $-$)
- Scaling (denoted implicitly)
- Dot product (denoted $\cdot$)
- Cross product (denoted $\times$)

We'll discuss each of these vector operations in Section 2.2. Although you should already be familiar with vectors and vector operations from Section 1.13, it's worth revisiting these concepts in greater depth, because vectors are the foundation of linear algebra.

### Matrix operations

The mathematical operations defined for matrices $A$ and $B$ are:

- Addition (denoted $A + B$)
- Subtraction, the inverse of addition (denoted $A - B$)
- Scaling by a constant $\alpha$ (denoted $\alpha A$)
- Matrix product (denoted $AB$)
- Matrix-vector product (denoted $A\vec{v}$)
- Matrix inverse (denoted $A^{-1}$)
- Trace (denoted $\mathrm{Tr}(A)$)
- Determinant (denoted $\det(A)$ or $|A|$)

We'll define each of these operations in Section 2.3, and we'll learn about the various computational, geometric, and theoretical considerations associated with these matrix operations throughout the remainder of the book.

Let's now examine one important matrix operation in closer detail: the matrix-vector product $A\vec{x}$.

### Matrix-vector product

Consider the matrix $A \in \mathbb{R}^{m\times n}$ and the vector $\vec{v} \in \mathbb{R}^n$. The matrix-vector product $A\vec{x}$ produces a linear combination of the columns of the matrix $A$ with coefficients $\vec{x}$. For example, the product of a $3 \times 2$ matrix $A$ and a $2 \times 1$ vector $\vec{x}$ results in a $3 \times 1$ vector, which we'll denote $\vec{y}$:

$$\vec{y} = A\vec{x},$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \underbrace{\begin{bmatrix} x_1a_{11} + x_2a_{12} \\ x_1a_{21} + x_2a_{22} \\ x_1a_{31} + x_2a_{32} \end{bmatrix}}_{\text{row picture}} = \underbrace{x_1 \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \end{bmatrix} + x_2 \begin{bmatrix} a_{12} \\ a_{22} \\ a_{32} \end{bmatrix}}_{\text{column picture}}.$$

The key thing to observe in the above formula is the dual interpretation of the matrix-vector product $A\vec{x}$ in the "row picture" and in the "column picture." In the row picture, we obtain the vector $\vec{y}$ by computing the dot product of the vector $\vec{x}$ with each of the rows of the matrix $A$. In the column picture, we interpret the vector $\vec{y}$ as $x_1$ times the first column of $A$ plus $x_2$ times the second column of $A$. In other words, $\vec{y}$ is a linear combination of the columns of $A$. For example, if you want to obtain the linear combination consisting of three times the first column of $A$ and four times the second column of $A$, you can multiply $A$ by the vector $\vec{x} = \left[\begin{smallmatrix} 3 \\ 4 \end{smallmatrix}\right]$.

### Linear combinations as matrix products

Consider some set of vectors $\{\vec{e}_1, \vec{e}_2\}$, and a third vector $\vec{y}$ that is a *linear combination* of the vectors $\vec{e}_1$ and $\vec{e}_2$:

$$\vec{y} \;=\; \alpha\vec{e}_1 \;+\; \beta\vec{e}_2.$$

The numbers $\alpha, \beta \in \mathbb{R}$ are the coefficients in this linear combination.

The matrix-vector product is defined expressly for the purpose of studying linear combinations. We can describe the linear combination $\vec{y} = \alpha\vec{e}_1 + \beta\vec{e}_2$ as the following matrix-vector product:

$$\vec{y} \;=\; \begin{bmatrix} | & | \\ \vec{e}_1 & \vec{e}_2 \\ | & | \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

The matrix $E$ has $\vec{e}_1$ and $\vec{e}_2$ as columns. The dimensions of the matrix $E$ will be $n \times 2$, where $n$ is the dimension of the vectors $\vec{e}_1$, $\vec{e}_2$, and $\vec{y}$.

### Linear transformations

Dear readers, we've reached the key notion in the study of linear algebra. This is the crux. The essential fibre. The main idea. I know you're ready to handle it because you're familiar with functions of a real variable $f : \mathbb{R} \to \mathbb{R}$, and you just learned the definition of the matrix-vector product (in which the variables were chosen to subliminally remind you of the standard conventions for the function input $x$ and the function output $y = f(x)$). Without further ado, I present to you the concept of a *linear transformation*.

The matrix-vector product corresponds to the abstract notion of a *linear transformation*, which is one of the key notions in the study of linear algebra. Multiplication by a matrix $A \in \mathbb{R}^{m\times n}$ can be thought of as computing a linear transformation $T_A$ that takes $n$-vectors as inputs and produces $m$-vectors as outputs:

$$T_A : \mathbb{R}^n \to \mathbb{R}^m.$$

Instead of writing $\vec{y} = T_A(\vec{x})$ to denote the linear transformation $T_A$ applied to the vector $\vec{x}$, we can write $\vec{y} = A\vec{x}$. Since the matrix $A$ has $m$ rows, the result of the matrix-vector product is an $m$-vector. Applying the linear transformation $T_A$ to the vector $\vec{x}$ corresponds to the product of the matrix $A$ and the column vector $\vec{x}$. We say $T_A$ is *represented by* the matrix $A$.

**Inverse** When a matrix $A$ is square and invertible, there exists an inverse matrix $A^{-1}$ which *undoes* the effect of $A$ to restore the original input vector:

$$A^{-1}(A\vec{x}) = A^{-1}A\vec{x} = \vec{x}.$$

Using the matrix inverse $A^{-1}$ to undo the effects of the matrix $A$ is analogous to using the inverse function $f^{-1}$ to undo the effects of the function $f$.

**Example 1** Consider the linear transformation that multiplies the first components of input vectors by 3 and multiplies the second components by 5, as described by the matrix

$$A = \begin{bmatrix} 3 & 0 \\ 0 & 5 \end{bmatrix}, \qquad A\vec{x} = \begin{bmatrix} 3 & 0 \\ 0 & 5 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3x_1 \\ 5x_2 \end{bmatrix}.$$

The inverse of the matrix $A$ is

$$A^{-1} = \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{5} \end{bmatrix}, \qquad A^{-1}(A\vec{x}) = \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{5} \end{bmatrix}\begin{bmatrix} 3x_1 \\ 5x_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \vec{x}.$$

The inverse matrix multiplies the first component by $\frac{1}{3}$ and the second component by $\frac{1}{5}$, which effectively undoes what $A$ did.

**Example 2** Things get a little more complicated when matrices *mix* the different components of the input vector, as in this example:

$$B = \begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix}, \text{ which acts as } B\vec{x} = \begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_1 + 2x_2 \\ 3x_2 \end{bmatrix}.$$

Make sure you understand how to compute $B\vec{x}$ using both the *row picture* and the *column picture* of the matrix-vector product.

The inverse of the matrix $B$ is the matrix

$$B^{-1} = \begin{bmatrix} 1 & \frac{-2}{3} \\ 0 & \frac{1}{3} \end{bmatrix}.$$

Multiplication by the matrix $B^{-1}$ is the "undo action" for multiplication by $B$:

$$B^{-1}(B\vec{x}) = \begin{bmatrix} 1 & \frac{-2}{3} \\ 0 & \frac{1}{3} \end{bmatrix}\begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 & \frac{-2}{3} \\ 0 & \frac{1}{3} \end{bmatrix}\begin{bmatrix} x_1 + 2x_2 \\ 3x_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \vec{x}.$$

By definition, the inverse $A^{-1}$ *undoes* the effects of the matrix $A$. The cumulative effect of applying $A^{-1}$ after $A$ is the *identity matrix* $\mathbb{1}$, which has 1s on the diagonal and 0s everywhere else:

$$A^{-1}A\vec{x} = \mathbb{1}\vec{x} = \vec{x} \qquad \Rightarrow \qquad A^{-1}A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \mathbb{1}.$$

Note that $\mathbb{1}\vec{x} = \vec{x}$ for any vector $\vec{x}$.

We'll discuss matrix inverses and how to compute them in more detail later (Section 3.5). For now, it's important you know they exist.

## An overview of linear algebra

In the remainder of the book, we'll learn all about the properties of vectors and matrices. Matrix-vector products play an important role in linear algebra because of their relation to *linear transformations*.

Functions are transformations from an input space (the domain) to an output space (the image). A linear transformation $T : \mathbb{R}^n \to \mathbb{R}^m$ is a function that takes $n$-vectors as inputs and produces $m$-vectors as outputs. If the function $T$ is linear, the output $\vec{y} = T(\vec{x})$ of $T$ applied to $\vec{x}$ can be computed as the matrix-vector product $A_T\vec{x}$, for some matrix $A_T \in \mathbb{R}^{m\times n}$. We say $T$ is *represented by* the matrix $A_T$. Equivalently, every matrix $A \in \mathbb{R}^{m\times n}$ corresponds to some linear transformation $T_A : \mathbb{R}^n \to \mathbb{R}^m$. Given the equivalence between matrices and linear transformations, we can reinterpret the statement "linear algebra is about vectors and matrices" by saying "linear algebra is about vectors and linear transformations."

You can adapt your existing knowledge about functions to the world of linear transformations. The action of a function on a number is similar to the action of a linear transformation on a vector. The table below summarizes several useful correspondences between functions and linear transformations.

| | | |
|---|---|---|
| function $f : \mathbb{R} \to \mathbb{R}$ | $\Leftrightarrow$ | linear transformation $T_A : \mathbb{R}^n \to \mathbb{R}^m$ represented by the matrix $A \in \mathbb{R}^{m\times n}$ |
| input $x \in \mathbb{R}$ | $\Leftrightarrow$ | input $\vec{x} \in \mathbb{R}^n$ |
| output $f(x) \in \mathbb{R}$ | $\Leftrightarrow$ | output $T_A(\vec{x}) = A\vec{x} \in \mathbb{R}^m$ |
| $g \circ f\,(x) = g(f(x))$ | $\Leftrightarrow$ | $T_B(T_A(\vec{x})) = BA\vec{x}$ |
| function inverse $f^{-1}$ | $\Leftrightarrow$ | matrix inverse $A^{-1}$ |
| roots of $f$ | $\Leftrightarrow$ | kernel of $T_A$ = null space of $A = \mathcal{N}(A)$ |
| image of $f$ | $\Leftrightarrow$ | image of $T_A$ = column space of $A = \mathcal{C}(A)$ |

**Table 2.1:** Correspondences between functions and linear transformations.

This table of correspondences serves as a roadmap for the rest of the material in this book. You'll notice the table introduces several new linear algebra concepts like *kernel*, *null space*, and *column space*, but not too many. You can totally do this!

Remember to always connect the new concepts of linear algebra to concepts you're already familiar with. For example, the roots of a function $f(x)$ are the set of inputs for which the function's output is zero. Similarly, the *kernel* of a linear transformation $T$ is the set of inputs that $T$ sends to the zero vector. The roots of a function $f : \mathbb{R} \to \mathbb{R}$ and the kernel of a linear transformation $T_A : \mathbb{R}^n \to \mathbb{R}^m$ are essentially the same concept; we're just upgrading functions to vector inputs.

In Chapter 1, I explained why functions are useful tools for modelling the real world. Well, linear algebra is the "vector upgrade" to your real-world modelling skills. With linear algebra you'll be able to model complex relationships between multivariable inputs and multivariable outputs. To build modelling skills, you must first develop your geometric intuition about lines, planes, vectors, bases, linear transformations, vector spaces, vector subspaces, etc. It's a lot of work, but the effort you invest will pay dividends.

## Links

[ Linear algebra lecture series by Prof. Strang from MIT ]
`http://bit.ly/1ayRcrj` (row and column picture example)

[ A system of equations in the row picture and column picture ]
`https://www.youtube.com/watch?v=uNKDw46_Ev4`

## Exercises

**E2.1** Find the inverse matrix $A^{-1}$ for the matrix $A = \left[\begin{smallmatrix} 7 & 0 \\ 0 & 2 \end{smallmatrix}\right]$. Verify that $A^{-1}(A\vec{v}) = \vec{v}$ for any vector $\vec{v} = \left[\begin{smallmatrix} v_1 \\ v_2 \end{smallmatrix}\right]$.

**E2.2** Given the matrices $A = \left[\begin{smallmatrix} 1 & 3 \\ 4 & 5 \end{smallmatrix}\right]$ and $B = \left[\begin{smallmatrix} -1 & 0 \\ 3 & 3 \end{smallmatrix}\right]$, and the vectors $\vec{v} = \left[\begin{smallmatrix} 1 \\ 2 \end{smallmatrix}\right]$ and $\vec{w} = \left[\begin{smallmatrix} -3 \\ -4 \end{smallmatrix}\right]$, compute the following expressions.

**a)** $A\vec{v}$ **b)** $B\vec{v}$ **c)** $A(B\vec{v})$ **d)** $B(A\vec{v})$ **e)** $A\vec{w}$ **f)** $B\vec{w}$

**E2.3** Find the components $v_1$ and $v_2$ of the vector $\vec{v} = \left[\begin{smallmatrix} v_1 \\ v_2 \end{smallmatrix}\right]$ so that $E\vec{v} = 3\vec{e}_2 - 2\vec{e}_1$, where $E$ is the following matrix:

$$E = \begin{bmatrix} | & | \\ \vec{e}_1 & \vec{e}_2 \\ | & | \end{bmatrix}.$$

## What next?

We won't bring geometry, vector spaces, algorithms, and the applications of linear algebra into the mix all at once. Instead, let's start with the basics. Since linear algebra is about vectors and matrices, let's define vectors and matrices precisely, and describe the math operations we can perform on them.

## 2.2 Vector operations

Section 1.13 introduced some basic notions about vectors. Understanding vectors is so important for linear algebra that it's worth going beyond the rudimentary understanding of vectors as "directional quantities," and so we took the time to describe vectors more abstractly—as math objects. With vectors defined, our next step is to specify their properties and the operations we can perform on vectors. This is what this section is all about.

### Definitions

Consider the vectors $\vec{u} = (u_1, u_2, u_3)$ and $\vec{v} = (v_1, v_2, v_3)$, and an arbitrary constant $\alpha \in \mathbb{R}$. Vector algebra can be summarized as the following operations:

- Addition: $\vec{u} + \vec{v} \stackrel{\text{def}}{=} (u_1 + v_1, u_2 + v_2, u_3 + v_3)$
- Subtraction: $\vec{u} - \vec{v} \stackrel{\text{def}}{=} (u_1 - v_1, u_2 - v_2, u_3 - v_3)$
- Scaling: $\alpha\vec{u} \stackrel{\text{def}}{=} (\alpha u_1, \alpha u_2, \alpha u_3)$
- Dot product: $\vec{u} \cdot \vec{v} \stackrel{\text{def}}{=} u_1 v_1 + u_2 v_2 + u_3 v_3$
- Cross product: $\vec{u} \times \vec{v} \stackrel{\text{def}}{=} (u_2 v_3 - u_3 v_2,\ u_3 v_1 - u_1 v_3,\ u_1 v_2 - u_2 v_1)$
- Length: $\|\vec{u}\| \stackrel{\text{def}}{=} \sqrt{u_1^2 + u_2^2 + u_3^2}$

In the next few pages we'll see what these operations can do for us.

### Notation

The set of real numbers is denoted $\mathbb{R}$. An $n$-dimensional real vector consists of $n$ real numbers slapped together in a bracket. We denote the set of three-dimensional vectors as $\mathbb{R}^3 \stackrel{\text{def}}{=} (\mathbb{R}, \mathbb{R}, \mathbb{R})$. Similarly, the set of $n$-dimensional real vectors is denoted $\mathbb{R}^n$.

When learning about new math operations, it's important to keep track of the types of inputs and the types of outputs of each operation. In computer science, this information is called the *type signature*. For example, the type signature "op : $\mathbb{R} \to \mathbb{R}$" tells us the math operation op takes real numbers as inputs and produces real numbers as outputs. Certain operations take pairs of numbers as inputs, like addition for example: $\text{add}(a, b)$ (usually denoted $a + b$). The type

## Exercises

**E2.4** Given the vectors $\vec{u} = (1,1,0)$ and $\vec{v} = (0,0,3)$, compute the following vector expressions:

**a)** $\vec{u} + \vec{v}$ **b)** $\vec{u} - \vec{v}$ **c)** $3\vec{u} + \vec{v}$ **d)** $\|\vec{u}\|$

**E2.5** Given $\vec{v} = (1,2,3)$ and $\vec{w} = (0,1,1)$, compute the following vector products: **a)** $\vec{v} \cdot \vec{w}$; **b)** $\vec{v} \times \vec{w}$; **c)** $\vec{w} \times \vec{v}$; **d)** $\vec{w} \times \vec{w}$.

**E2.6** For each of the following vectors, $\vec{v}_1 = 10\angle 10°$, $\vec{v}_2 = 10\angle 30°$, $\vec{v}_3 = 10\angle 60°$, $\vec{v}_4 = 10\angle 120°$, complete the following tasks:

**a)** Draw the vector in a Cartesian plane.
**b)** Compute the vector's $x$- and $y$-coordinates.
**c)** Compute the projection of the vector in the direction $\hat{\imath}$. Your answer should be a vector quantity.
**d)** Compute the projection of the vector in the direction $\hat{\jmath}$.
**e)** Compute the projection of the vector in the direction $\vec{d} = (1,1)$, and find the length of the projection.

Hint: Recall the formula for the projection of the vector $\vec{v}$ in the direction $\vec{d}$ is defined as $\Pi_{\vec{d}}(\vec{v}) = \left(\frac{\vec{d}\cdot\vec{v}}{\|\vec{d}\|^2}\right)\vec{d}$.

# 2.3 Matrix operations

A matrix is a two-dimensional array (a table) of numbers. Consider the $m$ by $n$ matrix $A \in \mathbb{R}^{m\times n}$. We denote the matrix as a whole $A$ and refer to its individual entries as $a_{ij}$, where $a_{ij}$ is the number in the $i^{\text{th}}$ row and the $j^{\text{th}}$ column of $A$. What are the mathematical operations we can perform on this matrix?

## Addition and subtraction

The matrix addition and subtraction operations take pairs of matrices as inputs and produce matrices as outputs:

$$+ : \mathbb{R}^{m\times n} \times \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n} \quad \text{and} \quad - : \mathbb{R}^{m\times n} \times \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}.$$

Addition and subtraction are performed as follows:

$$C = A \pm B \quad \Leftrightarrow \quad c_{ij} = a_{ij} \pm b_{ij},\ \forall i \in [1,\ldots,m], j \in [1,\ldots,n].$$

For example, addition for two $3\times 2$ matrices is expressed as

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} \\ a_{21}+b_{21} & a_{22}+b_{22} \\ a_{31}+b_{31} & a_{32}+b_{32} \end{bmatrix}.$$

Matrices must have the same dimensions to be added or subtracted.

## Exercises

**E2.7** Given the matrices $A = \begin{bmatrix} 3 & 4 \\ 2 & 1 \end{bmatrix}$, $B = \begin{bmatrix} -1 & 0 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix}$, and $C = \begin{bmatrix} -2 & 3 & 0 \\ 2 & -2 & 1 \end{bmatrix}$, compute the expressions.

**a)** $A^{\mathsf{T}}$ **b)** $C^{\mathsf{T}}$ **c)** $A^2$ **d)** $AB$ **e)** $AC$ **f)** $BA$ **g)** $C^{\mathsf{T}}A$
**h)** $\det(A)$ **i)** $\det(B)$ **j)** $\det(C)$ **k)** $\det(A^{\mathsf{T}})$ **l)** $\det(AA^{-1})$
**m)** $\mathrm{Tr}(A)$ **n)** $\mathrm{Tr}(A^{\mathsf{T}})$

Hint: Some of these expressions may not exist.

**E2.8** Given the $1 \times 3$ matrices (row vectors) $\vec{u} = (1,2,3)$ and $\vec{v} = (2,-1,0)$, compute the following products:

**a)** $\vec{u}\,\vec{u}^{\mathsf{T}}$ **b)** $\vec{v}\,\vec{v}^{\mathsf{T}}$ **c)** $\vec{u}\,\vec{v}^{\mathsf{T}}$ **d)** $\vec{u}^{\mathsf{T}}\vec{u}$ **e)** $\vec{v}^{\mathsf{T}}\vec{v}$ **f)** $\vec{u}^{\mathsf{T}}\vec{v}$

Hint: The transpose of a $1 \times 3$ row vector is a $3 \times 1$ column vector.

**E2.9** Find the unknowns $\alpha$ and $\beta$ in the equation $\begin{bmatrix} 2 & \alpha \\ \beta & -3 \end{bmatrix}\begin{bmatrix} 1 \\ 4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$.

# 2.4 Linearity

What is linearity and why is this entire book dedicated to learning about it? Consider an arbitrary function that contains terms with different powers of the input variable $x$:

$$f(x) = \underbrace{a/x}_{\text{one-over-}x} + \underbrace{b}_{\text{constant}} + \underbrace{mx}_{\text{linear term}} + \underbrace{qx^2}_{\text{quadratic}} + \underbrace{cx^3}_{\text{cubic}}.$$

The term $mx$ is the *linear* term in this expression—it contains $x$ raised to the first power. All the other terms are *nonlinear*. The linear term is special because changes in the value of the input $x$ lead to proportional changes in the value of $mx$. If the input is $2x$, the linear term will have value $2mx$. If the input is $100x$, the linear term will have value $100mx$. This input-output proportionality does not hold for nonlinear terms.

In this section we'll discuss the special properties of expressions and equations containing only linear terms.

## Introduction

A single-variable function takes as its input a real number $x$ and outputs a real number $f(x)$. The type signature of this function is

$$f\colon \mathbb{R} \to \mathbb{R}.$$

# 2.5 Overview of linear algebra

In linear algebra, you'll learn new computational techniques and develop new ways of thinking about math. With these new tools, you'll be able to use linear algebra techniques for many applications. Let's look at what lies ahead in this book.

## Computational linear algebra

The first steps toward understanding linear algebra will seem a little tedious. In Chapter 3 you'll develop basic skills for manipulating vectors and matrices. Matrices and vectors have many components and performing operations on them involves many arithmetic steps—there is no way to circumvent this complexity. Make sure you understand the basic algebra rules (how to add, subtract, and multiply vectors and matrices) because they are a prerequisite for learning more advanced material. You should be able to perform all the matrix algebra operations with pen and paper for small and medium-sized matrices.

The good news is, with the exception of your homework assignments and final exam, you won't have to carry out matrix algebra by hand. It is much more convenient to use a computer for large matrix calculations. The more you develop your matrix algebra skills, the deeper you'll be able to delve into the advanced topics.

## Geometric linear algebra

So far, we've described vectors and matrices as arrays of numbers. This is fine for the purpose of doing *algebra* on vectors and matrices, but this description is not sufficient for understanding their geometric properties. The components of a vector $\vec{v} \in \mathbb{R}^n$ can be thought of as distances measured along a coordinate system with $n$ axes. The vector $\vec{v}$ can therefore be said to "point" in a particular direction with respect to the coordinate system. The fun part of linear algebra starts when you learn about the geometric interpretation of the algebraic operations on vectors and matrices.

Consider some unit vector that specifies a direction of interest $\hat{r}$. Suppose we're given some other vector $\vec{v}$, and we're asked to find *how much of $\vec{v}$ is in the $\hat{r}$ direction*. The answer is computed using the dot product: $v_r = \vec{v} \cdot \hat{r} = \|\vec{v}\| \cos\theta$, where $\theta$ is the angle between $\vec{v}$ and $\hat{r}$. The technical term for the quantity $v_r$ is "the length of the projection of $\vec{v}$ in the $\hat{r}$ direction." By "projection," I mean we ignore all parts of $\vec{v}$ that are not in the $\hat{r}$ direction. Projections are used in mechanics to calculate the $x$- and $y$-components of forces in force diagrams. In Section 4.2 we'll learn how to calculate all kinds of projections using the dot product.

To further consider the geometric aspects of vector operations, imagine the following situation. Suppose I gave you two vectors $\vec{u}$ and $\vec{v}$, and asked you to find a third vector $\vec{w}$ that is perpendicular to both $\vec{u}$ and $\vec{v}$. A priori this sounds like a complicated question to answer, but in fact the required vector $\vec{w}$ can easily be obtained by computing the cross product $\vec{w} = \vec{u} \times \vec{v}$.

In Section 4.1 we'll learn how to describe lines and planes in terms of points, direction vectors, and normal vectors. Consider the following geometric problem: given the equations of two planes in $\mathbb{R}^3$, find the equation of the line where the two planes intersect. There is an algebraic procedure called *Gauss–Jordan elimination* we can use to find the solution.

The determinant of a matrix has a geometric interpretation (Section 3.4). The determinant tells us something about the relative orientation of the vectors that make up the rows of the matrix. If the determinant of a matrix is zero, it means the rows are not *linearly independent*, in other words, at least one of the rows can be written in terms of the other rows. Linear independence, as we'll see shortly, is an important property for vectors to have. The determinant is a convenient way to test whether vectors are linearly independent.

As you learn about geometric linear algebra, practice *visualizing* each new concept you learn about. Always keep a picture in your head of what is going on. The relationships between two-dimensional vectors can be represented in vector diagrams. Three-dimensional vectors can be visualized by pointing pens and pencils in different directions. Most of the intuition you build about vectors in two and three dimensions are applicable to vectors with more dimensions.

## Theoretical linear algebra

Linear algebra will teach you how to reason about vectors and matrices in an abstract way. By thinking abstractly, you'll be able to extend your geometric intuition of two and three-dimensional problems to problems in higher dimensions. Much *knowledge buzz* awaits as you learn about new mathematical ideas and develop new ways of thinking.

You're no doubt familiar with the normal coordinate system made of two orthogonal axes: the $x$-axis and the $y$-axis. A vector $\vec{v} \in \mathbb{R}^2$ is specified in terms of its coordinates $(v_x, v_y)$ with respect to these axes. When we say $\vec{v} = (v_x, v_y)$, what we really mean is $\vec{v} = v_x\hat{\imath} + v_y\hat{\jmath}$, where $\hat{\imath}$ and $\hat{\jmath}$ are unit vectors that point along the $x$- and $y$-axes. As it turns out, we can use many other kinds of coordinate systems to represent vectors. A *basis* for $\mathbb{R}^2$ is any set of two vectors $\{\hat{e}_1, \hat{e}_2\}$ that allows us to express all vectors $\vec{v} \in \mathbb{R}^2$ as linear combinations of the basis vectors: $\vec{v} = v_1\hat{e}_1 + v_2\hat{e}_2$. The same vector $\vec{v}$ corresponds to two different coordinate pairs, depending on which basis is used for the description: $\vec{v} = (v_x, v_y)$ in the basis $\{\hat{\imath}, \hat{\jmath}\}$ and $\vec{v} = (v_1, v_2)$ in the basis $\{\hat{e}_1, \hat{e}_2\}$. We'll learn about the properties of bases in great detail in the coming chapters. The choice of basis plays a fundamental role in all aspects of linear algebra.

In the text above, I explained that computing the product between a matrix and a vector $A\vec{x} = \vec{y}$ can be thought of as a linear transformation, with input $\vec{x}$ and output $\vec{y}$. Any linear transformation (Section 5.1) can be represented (Section 5.2) as a multiplication by a matrix $A$. Conversely, every $m \times n$ matrix $A \in \mathbb{R}^{m\times n}$ can be thought of as performing a linear transformation $T_A : \mathbb{R}^n \to \mathbb{R}^m$. The equivalence between matrices and linear transformations allows us to identify certain matrix properties with properties of linear transformations. For example, the *column space* $\mathcal{C}(A)$ of the matrix $A$ (the set of vectors that can be written as a combination of the columns of $A$) corresponds to the image space of the linear transformation $T_A$ (the set of possible outputs of $T_A$).

The eigenvalues and eigenvectors of matrices (Section 6.1) allow us to describe the actions of matrices in a natural way. The set of eigenvectors of a matrix are special input vectors for which the action of the matrix is described as a *scaling*. When a matrix acts on one of its eigenvectors, the output is a vector in the same direction as the input vector scaled by a constant. The scaling constant is the *eigenvalue* (own value) associated with this eigenvector. By specifying all the eigenvectors and eigenvalues of a matrix, it is possible to obtain a complete description of what the matrix does. Thinking of matrices in terms of their eigenvalues and eigenvectors is a powerful technique for describing their properties and has many applications.

Linear algebra is useful because linear algebra techniques can be applied to all kinds of "vector-like" objects. The abstract concept of a vector space (Section 6.3) captures precisely what it means for some class of mathematical objects to be "vector-like." For example, the set of polynomials of degree at most two, denoted $P_2(x)$, consists of all functions of the form $f(x) = a_0 + a_1x + a_2x^2$. Polynomials are vector-like because it's possible to describe each polynomial in terms of its coefficients $(a_0, a_1, a_2)$. Furthermore, the sum of two polynomials and the multiplication of a polynomial by a constant both correspond to vector-like calculations of coefficients. Once you realize polynomials are vector-like, you'll be able to use notions like *linear independence*, *dimension*, and *basis* when working with polynomials.

## Useful linear algebra

One of the most useful skills you'll learn in linear algebra is the ability to solve systems of linear equations. Many real-world problems are expressed as linear equations in multiple unknown quantities. You can solve for $n$ unknowns simultaneously if you have a set of $n$ linear equations that relate the unknowns. To solve this system of equations, eliminate the variables one by one using basic techniques such as substitution and subtraction (see Section 1.15); however, the procedure will be slow and tedious for many unknowns. If the system of equations is linear, it can be expressed as an *augmented matrix* built from the coefficients in the equations. You can then use the Gauss–Jordan elimination algorithm to solve for the $n$ unknowns (Section 3.1). The key benefit of the augmented matrix approach is that it allows you to focus on the coefficients without worrying about the variable names. This saves time when you must solve for many unknowns. Another approach for solving $n$ linear equations in $n$ unknowns is to express the system of equations as a matrix equation (Section 3.2) and then solve the matrix equation by computing the matrix inverse (Section 3.5).

In Section 6.6 you'll learn how to *decompose* a matrix into a product of simpler matrices. Matrix decompositions are often performed for computational reasons: certain problems are easier to solve on a computer when the matrix is expressed in terms of its simpler constituents. Other decompositions, like the decomposition of a matrix into its eigenvalues and eigenvectors, give you valuable information about the properties of the matrix. Google's original PageRank algorithm for ranking webpages by "importance" can be explained as the search for an eigenvector of a matrix. The matrix in question contains information about all hyperlinks that exist between webpages. The eigenvector we're looking for corresponds to a vector that describes the relative importance of each page. So when I tell you eigenvectors are *valuable information*, I'm not kidding: a little 350-billion-dollar company called Google started as an eigenvector idea.

The techniques of linear algebra find applications in many areas of science and technology. We'll discuss applications such as *modelling* multidimensional real-world problems, finding *approximate solutions* to equations (curve fitting), solving constrained optimization problems using *linear programming*, and many other in Chapter 7. As a special bonus for readers interested in physics, a short introduction to quantum mechanics can be found in Chapter 9; if you have a good grasp of linear algebra, you can understand matrix quantum mechanics at no additional mental cost.

Our journey into the land of linear algebra will continue in the next chapter with the study of computational aspects of linear algebra. We'll learn how to solve large systems of linear equations, practice computing matrix products, discuss matrix determinants, and compute matrix inverses.

## 2.6 Introductory problems

We've been having fun learning about vector and matrix operations, and we've also touched upon linear transformations. I've summarized what linear algebra is about; now it's time for you to put in the effort and check whether you understand the definitions of the operations.

Don't cheat yourself by thinking my summaries are enough; you can't magically understand everything about linear algebra merely by reading about it. Learning doesn't work that way! The only way to truly "get" math—especially advanced math—is to solve problems using the new concepts you've learned. Indeed, the only math I remember from my university days is math that I practiced by solving lots of problems. There's no better way to test whether you understand than testing yourself. Of course, it's your choice whether you'll dedicate the next hour of your life to working through the problems in this section. All I'll say is that you'll have something to show for your efforts; and it's totally worth it.

**P2.1** Which of the following functions are linear?

**a)** $q(x) = x^2$ **b)** $f(x) = g(h(x))$, where $g(x) = \sqrt{3}x$ and $h(x) = -4x$
**c)** $i(x) = \frac{1}{mx}$ **d)** $j(x) = \frac{x-a}{x-b}$

**P2.2** Find the sum of the vector $(1,0,1)$ and the vector $(0,2,2)$.

**P2.3** Your friend is taking a quantum physics class and needs your help answering the following vectors question. "Let $|a\rangle = 1|0\rangle + 3|1\rangle$ and $|b\rangle = 4|0\rangle - 1|1\rangle$. Find $|a\rangle + |b\rangle$."

Hint: The angle-bracket notation describes vectors: $|0\rangle = \hat{\imath}$ and $|1\rangle = \hat{\jmath}$.

**P2.4** Given unit vectors $\hat{\imath} = (1,0,0)$, $\hat{\jmath} = (0,1,0)$, and $\hat{k} = (0,0,1)$, find the following cross products: **a)** $\hat{\imath} \times \hat{\imath}$, **b)** $\hat{\imath} \times \hat{\jmath}$, **c)** $(-\hat{\imath}) \times \hat{k} + \hat{\jmath} \times \hat{\imath}$, **d)** $\hat{k} \times \hat{\jmath} + \hat{\imath} \times \hat{\imath} + \hat{\jmath} \times \hat{k} + \hat{\jmath} \times \hat{\imath}$.

**P2.5** Given $\vec{v} = (2,-1,3)$ and $\vec{w} = (1,0,1)$, compute the following vector products: **a)** $\vec{v} \cdot \vec{w}$, **b)** $\vec{v} \times \vec{w}$, **c)** $\vec{v} \times \vec{v}$, and **d)** $\vec{w} \times \vec{w}$.

**P2.6** Given the vectors $\vec{u} = (1,1,1)$, $\vec{v} = (2,3,1)$, and $\vec{w} = (-1,-1,2)$, compute the following products:

**a)** $\vec{u} \cdot \vec{v}$ **b)** $\vec{u} \cdot \vec{w}$ **c)** $\vec{v} \cdot \vec{w}$

**d)** $\vec{u} \times \vec{v}$ **e)** $\vec{u} \times \vec{w}$ **f)** $\vec{v} \times \vec{w}$

# Chapter 3

# Computational linear algebra

This chapter covers the computational aspects of performing matrix calculations. Understanding matrix computations is important because the rest of the chapters in this book depend on them. Suppose we're given a huge matrix $A \in \mathbb{R}^{n \times n}$ with $n = 1000$. Behind the innocent-looking mathematical notation of the matrix inverse $A^{-1}$, the matrix product $AA$, and the matrix determinant $\det(A)$, are hidden monster computations involving all the $1000 \times 1000 = 1$ million entries of the matrix $A$. Millions of arithmetic operations must be performed...so I hope you have at least a thousand pencils ready!

Okay, calm down. I won't *actually* make you calculate millions of arithmetic operations. In fact, to learn linear algebra, it is sufficient to know how to carry out calculations with $3 \times 3$ and $4 \times 4$ matrices. Yet, even for such moderately sized matrices, computing products, inverses, and determinants by hand are serious computational tasks. If you're ever required to take a linear algebra final exam, you'll need to make sure you can do these calculations quickly. And even if no exam looms in your imminent future, it's still important you practice matrix operations by hand to get a feel for them.

This chapter will introduce you to four important computational tasks involving matrices.

**Gauss–Jordan elimination** Suppose we're trying to solve two equations in two unknowns $x$ and $y$:

$$\begin{aligned} ax + by &= c, \\ dx + ey &= f. \end{aligned}$$

If we add $\alpha$-times the first equation to the second equation, we obtain an equivalent system of equations:

$$\begin{aligned} ax \; + \quad\quad\; by &= \quad\; c \\ (d+\alpha a)x + (e+\alpha b)y &= f+\alpha c. \end{aligned}$$

This is called a *row operation*: we added $\alpha$-times the first row to the second row. Row operations change the coefficients of the system of equations, but leave the solution unchanged. Gauss–Jordan elimination is a systematic procedure for solving systems of linear equations using row operations.

**Matrix multiplication** The product $AB$ between matrices $A \in \mathbb{R}^{m\times \ell}$ and $B \in \mathbb{R}^{\ell \times n}$ is the matrix $C \in \mathbb{R}^{m\times n}$ whose entries $c_{ij}$ are defined by the formula $c_{ij} = \sum_{k=1}^{\ell} a_{ik}b_{kj}$ for all $i \in [1,\ldots,m]$ and $j \in [1,\ldots,n]$. In Section 3.3, we'll unpack this formula and learn its intuitive interpretation: that computing $C = AB$ is computing all the dot products between the rows of $A$ and the columns of $B$.

**Determinant** The determinant of a matrix $A$, denoted $\det(A)$, is an operation that gives us useful information about the linear independence of the rows of the matrix. The determinant is connected to many notions of linear algebra: linear independence, geometry of vectors, solving systems of equations, and matrix invertibility. We'll discuss these aspects of determinants in Section 3.4.

**Matrix inverse** In Section 3.5, we'll build upon our knowledge of Gauss–Jordan elimination, matrix products, and determinants to derive three different procedures for finding the matrix inverse $A^{-1}$.

## 3.1 Reduced row echelon form

In this section we'll learn to solve systems of linear equations using the *Gauss–Jordan elimination* procedure. A system of equations can be represented as a matrix of coefficients. The Gauss–Jordan elimination procedure converts any matrix into its *reduced row echelon form* (RREF). We can use the RREF to easily find the solution (or solutions) of the system of equations.

Heads up: the material covered in this section requires your full-on, caffeinated attention, as the procedures you'll learn are somewhat tedious. Gauss–Jordan elimination involves many repetitive manipulations of arrays of numbers. It's important you follow the step-by-step manipulations, as well as verify each step I present *on your own* with pen and paper. Don't just take my word for it—always verify the steps!

## Solving equations

Suppose you're asked to solve the following system of equations:

$$\begin{aligned} 1x_1 + 2x_2 &= 5 \\ 3x_1 + 9x_2 &= 21. \end{aligned}$$

The standard approach is to use one of the equation-solving tricks we learned in Section 1.15 to combine the equations and find the values of the two unknowns $x_1$ and $x_2$.

Observe that the *names* of the two unknowns are irrelevant to the solution of the system of equations. Indeed, the solution $(x_1, x_2)$ to the above system of equations is the same as the solution $(s, t)$ to the system of equations

$$\begin{aligned} 1s + 2t &= 5 \\ 3s + 9t &= 21. \end{aligned}$$

The important parts of a system of linear equations are the *coefficients* in front of the variables, and the constants on the right side of each equation.

## Augmented matrix

The system of linear equations can be written as an *augmented matrix*:

$$\left[\begin{array}{cc|c} 1 & 2 & 5 \\ 3 & 9 & 21 \end{array}\right].$$

The first column corresponds to the coefficients of the first variable; the second column is for the second variable; and the last column corresponds to the constants on the right side. It is customary to draw a vertical line where the equal signs in the equations would normally appear. This line helps distinguish the coefficients of the equations from the column of constants on the right side.

Once we have the augmented matrix, we can simplify it by using *row operations* (which we'll discuss shortly) on its entries. After simplification by row operations, the augmented matrix will be transformed to

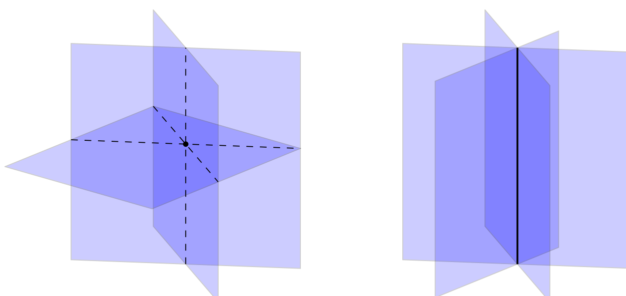
$$\left[\begin{array}{cc|c} 1 & 0 & 1 \\ 0 & 1 & 2 \end{array}\right],$$

which corresponds to the system of equations

$$\begin{aligned} x_1 &= 1 \\ x_2 &= 2. \end{aligned}$$

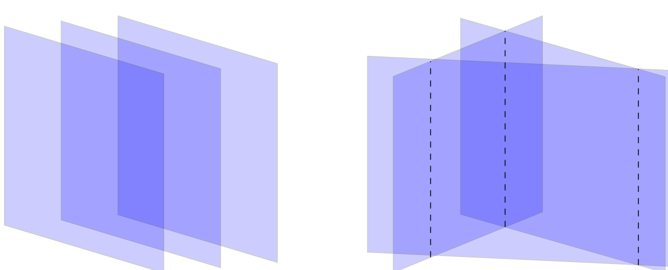
we want to find the points $(x, y, z)$ that satisfy all three equations simultaneously. There are four possibilities for the solution set:

- **One solution**. Three non-parallel planes intersect at a point.
- **Infinitely many solutions**. If one of the plane equations is redundant, the solution corresponds to the intersection of two planes. Two non-parallel planes intersect on a line.
- **Infinitely many solutions in two dimensions**. If two of the equations are redundant, the solution space is a two-dimensional plane.
- **No solution**. If no common points exist at the intersection of all three planes, then the system of equations has no solution.



(a) One solution (b) Infinite solutions

**Figure 3.1:** Three planes can intersect at a unique point, as in figure (a); or along a line, as in figure (b). In the first case, there is a unique point $(x_o, y_o, z_o)$ common to all three planes. In the second case, all points on the line $\{p_o + t\vec{v},\ \forall t \in \mathbb{R}\}$ are shared by the planes.



(a) No solution (b) No solution

**Figure 3.2:** These illustrations depict systems of three equations in three unknowns that have no solution. No common points of intersection exist.

## Computer power

The computer algebra system at `http://live.sympy.org` can be used to compute the reduced row echelon form of any matrix.

Here is an example of how to create a sympy `Matrix` object:

```
>>> from sympy.matrices import Matrix
>>> A = Matrix([[1, 2,  5],      # use SHIFT+ENTER for newline
                [3, 9, 21]])
```

In `Python`, we define lists using the square brackets `[` and `]`. A matrix is defined as a list of lists.

To compute the reduced row echelon form of `A`, call its `rref()` method:

```
>>> A.rref()
( [1, 0, 1] # RREF of A         # locations of pivots
  [0, 1, 2],                    [0, 1]               )
```

The `rref()` method returns a tuple containing the RREF of `A` and an array that tells us the 0-based indices of the columns that contain leading ones. Usually, we'll want to find the RREF of `A` and ignore the pivots; to obtain the RREF without the pivots, select the first (index zero) element in the result of `A.rref()`:

```
>>> Arref = A.rref()[0]
>>> Arref
[1, 0, 1]
[0, 1, 2]
```

The `rref()` method is the fastest way to obtain the reduced row echelon form of a `SymPy` matrix. The computer will apply the Gauss–Jordan elimination procedure and show you the answer. If you want to see the intermediary steps of the elimination procedure, you can also manually apply row operations to the matrix.

**Example** Let's compute the reduced row echelon form of the same augmented matrix by using row operations in `SymPy`:

```
>>> A = Matrix([[1, 2,  5],
                [3, 9, 21]])

>>> A[1,:] = A[1,:] - 3*A[0,:]
>>> A
[1, 2, 5]
[0, 3, 6]
```

We use the notation `A[i,:]` to refer to entire rows of the matrix. The number `i` specifies the 0-based row index: the first row of `A` is `A[0,:]` and the second row is `A[1,:]`. The code example above implements the row operation $R_2 \leftarrow R_2 - 3R_1$.

To obtain the reduced row echelon form of the matrix `A`, we carry out two more row operations, $R_2 \leftarrow \frac{1}{3}R_2$ and $R_1 \leftarrow R_1 - 2R_2$, using the following commands:

```
>>> A[1,:] = S(1)/3*A[1,:]
>>> A[0,:] = A[0,:] - 2*A[1,:]
>>> A
[1, 0, 1]          # the same result as A.rref()[0]
[0, 1, 2]
```

Note we represent the fraction $\frac{1}{3}$ as `S(1)/3` in order to obtain the exact rational expression `Rational(1,3)`. If we were to input $\frac{1}{3}$ as `1/3`, `SymPy` would interpret this either as integer or floating point division, which is not what we want. The single-letter helper function `S` is an alias for the function `sympify`, which ensures a `SymPy` object is produced. Another way to input the exact fraction $\frac{1}{3}$ is `S('1/3')`.

If you need to swap two rows of a matrix, you can use the standard Python tuple assignment syntax. To swap the position of the first and second rows, use

```
>>> A[0,:], A[1,:] = A[1,:], A[0,:]
>>> A
[0, 1, 2]
[1, 0, 1]
```

Using row operations to compute the reduced row echelon form of a matrix allows you to see the intermediary steps of a calculation; which is useful, for instance, when checking the correctness of your homework problems.

There are other applications of matrix methods that use row operations (see Section 7.6), so it's good idea to know how to use `SymPy` for this purpose.

## Discussion

In this section, we learned the Gauss–Jordan elimination procedure for simplifying matrices, which just so happens to be one of the most important computational tools of linear algebra. Beyond being a procedure for finding solutions to systems of linear equations, the Gauss–Jordan elimination algorithms can be used to solve a broad range of other linear algebra problems. Later in the book, we'll use the Gauss–Jordan elimination algorithm to compute inverse matrices (Section 3.5) and to "distill" bases for vector spaces (Section 4.5).

# 3.3 Matrix multiplication

Suppose we're given the matrices

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad \text{and} \qquad B = \begin{bmatrix} e & f \\ g & h \end{bmatrix},$$

and we want to compute the *matrix product* $AB$.

Unlike matrix addition and subtraction, matrix multiplication is *not* performed entry-wise:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} e & f \\ g & h \end{bmatrix} \neq \begin{bmatrix} ae & bf \\ cg & dh \end{bmatrix}.$$

Instead, the matrix product is computed by taking the dot product between each row of the matrix $A$ and each column of the matrix $B$:

$$\begin{array}{c} \vec{r}_1 \rightarrow \\ \vec{r}_2 \rightarrow \end{array} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{array}{c} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \\ \begin{array}{cc} \uparrow & \uparrow \\ \vec{c}_1 & \vec{c}_2 \end{array} \end{array} \begin{array}{l} = \begin{bmatrix} \vec{r}_1 \cdot \vec{c}_1 & \vec{r}_1 \cdot \vec{c}_2 \\ \vec{r}_2 \cdot \vec{c}_1 & \vec{r}_2 \cdot \vec{c}_2 \end{bmatrix} \\ = \begin{bmatrix} ae + bg & af + bh \\ ce + dg & cf + dh \end{bmatrix}. \end{array}$$

Recall the dot product between two vectors $\vec{v}$ and $\vec{w}$ is computed using the formula $\vec{v} \cdot \vec{w} = \sum_i v_i w_i$.

Now, let's look at a picture that shows how to compute the product of a matrix with four rows and another matrix with five columns. To compute the top left entry of the result, take the dot product of the first row of the first matrix and the first column of the second matrix.

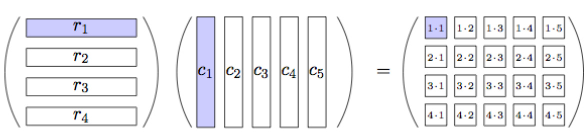



**Figure 3.3:** Matrix multiplication is performed rows-times-columns. The first-row, first-column entry of the product is the dot product of $r_1$ and $c_1$.

Similarly, the entry on the third row and fourth column of the product is computed by taking the dot product of the third row of the first matrix and the fourth column of the second matrix. See Figure 3.4.

For the matrix product to work, the rows of the first matrix must have the same length as the columns of the second matrix.

## Matrix multiplication rules

- Matrix multiplication is associative:

$$(AB)C = A(BC) = ABC.$$

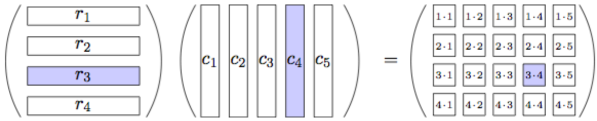



**Figure 3.4:** The third-row, fourth-column entry of the product is computed by taking the dot product of $r_3$ and $c_4$.

- The touching dimensions of the matrices must be the same. For the triple product $ABC$ to exist, the rows of $A$ must have the same dimension as the columns of $B$, and the rows of $B$ must have the same dimension as the columns of $C$.
- Given two matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times k}$, the product $AB$ is an $m \times k$ matrix.
- Matrix multiplication is not a commutative operation.

$$\boxed{A}\,\boxed{B} \neq \boxed{B}\,\boxed{A}$$

**Figure 3.5:** The order of multiplication matters for matrices: the product $AB$ does not equal the product $BA$.

**Example** Consider the matrices $A \in \mathbb{R}^{2\times 3}$ and $B \in \mathbb{R}^{3\times 2}$. The product $AB = C \in \mathbb{R}^{2\times 2}$ is computed as

$$\underbrace{\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}}_{B} = \begin{bmatrix} 1+6+15 & 2+8+18 \\ 4+15+30 & 8+20+36 \end{bmatrix} = \underbrace{\begin{bmatrix} 22 & 28 \\ 49 & 64 \end{bmatrix}}_{C}.$$

We can also compute the product $BA = D \in \mathbb{R}^{3\times 3}$:

$$\underbrace{\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}}_{A} = \begin{bmatrix} 1+8 & 2+10 & 3+12 \\ 3+16 & 6+20 & 9+24 \\ 5+24 & 10+30 & 15+36 \end{bmatrix} = \underbrace{\begin{bmatrix} 9 & 12 & 15 \\ 19 & 26 & 33 \\ 29 & 40 & 51 \end{bmatrix}}_{D}.$$

In each case, the touching dimensions of the two matrices in the product are the same. Note that $C = AB \neq BA = D$, and in fact, the products $AB$ and $BA$ are matrices with different dimensions.

## 3.5 Matrix inverse

In this section, we'll learn four different approaches for computing the inverse of a matrix. Since knowing how to compute matrix inverses is a pretty useful skill, learning several approaches is hardly overkill. Note that the matrix inverse is *unique*, so no matter which method you use to find the inverse, you'll always obtain the same answer. You can verify your calculations by computing the inverse in different ways and checking that the answers agree.

### Existence of an inverse

Not all matrices are invertible. Given a matrix $A \in \mathbb{R}^{n\times n}$, we can check whether it is invertible by computing its determinant:

$$A^{-1} \text{ exists} \qquad \text{if and only if} \qquad \det(A) \neq 0.$$

Calculating the determinant of a matrix serves as an *invertibility test*. The exact value of the determinant is not important; it could be big or small, positive or negative; as long as the determinant is nonzero, the matrix passes the invertibility test.

### Adjugate matrix approach

The inverse of a $2 \times 2$ matrix can be computed as follows:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}.$$

The above formula is the $2 \times 2$ version of a general formula for obtaining the inverse based on the *adjugate matrix*:

$$A^{-1} = \frac{1}{\det(A)}\text{adj}(A).$$

What is the adjugate matrix, you ask? The adjugate matrix is kind of complicated, so let's proceed step by step. We'll first define a few prerequisite concepts.

In this section, we'll work on a matrix $A \in \mathbb{R}^{3\times 3}$ and refer to its entries as $a_{ij}$, where $i$ is the row index and $j$ is the column index:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

First we'll define three concepts associated with determinants:

# 3.6 Computational problems

We've reached the problem section where you're supposed to practice all the computational techniques of linear algebra. This is not going to be the most exciting three hours of your life, but you'll get through it. You need to know how to solve computational problems by hand and apply the Gauss–Jordan elimination procedure; and you need to know how to multiply matrices, calculate determinants, and find matrix inverses. These computational techniques enable all the advanced procedures we'll develop later in the book. If you skip these practice problems, you'll have trouble later when it comes to mastering more advanced topics that rely on these basic matrix operations as building blocks. Do this important work now, and you'll be on your way to becoming fluent in linear algebra computations... plus, the rest of the book will be much more pleasant.

**P3.1** Mitchell is on a new diet. His target is to eat exactly 25 grams of fat and 32 grams of protein for lunch today. There are two types of food in the fridge, $x$ and $y$. One serving of food $x$ contains one gram of fat and two grams of protein, while a serving of food $y$ contains five grams of fat and one gram of protein. To figure out how many servings of each type of food he should eat, Mitchell writes the following system of equations:

$$\begin{array}{rcrcl} x & + & 5y & = & 25 \\ 2x & + & y & = & 32 \end{array} \qquad \Rightarrow \qquad \left[\begin{array}{cc|c} 1 & 5 & 25 \\ 2 & 1 & 32 \end{array}\right].$$

Help Mitchell find how many servings of $x$ and $y$ he should eat.

Hint: Find the reduced row echelon form of the augmented matrix.

**P3.2** Alice, Bob, and Charlotte are solving this system of equations:

$$\begin{array}{rrl} 3x & +3y & = 6 \\ 2x & +\frac{3}{2}y & = 5 \end{array} \qquad \Rightarrow \qquad \left[\begin{array}{cc|c} 3 & 3 & 6 \\ 2 & \frac{3}{2} & 5 \end{array}\right].$$

Alice follows the standard procedure to obtain a leading one by performing the row operation $R_1 \leftarrow \frac{1}{3}R_1$. Bob starts with a different row operation, applying $R_1 \leftarrow R_1 - R_2$ to obtain a leading one. Charlotte takes a third approach by swapping the first and second rows: $R_1 \leftrightarrow R_2$. Their respective versions of the augmented matrix are shown below.

$$\textbf{a)}\left[\begin{array}{cc|c} 1 & 1 & 2 \\ 2 & \frac{3}{2} & 5 \end{array}\right] \qquad \textbf{b)}\left[\begin{array}{cc|c} 1 & \frac{3}{2} & 1 \\ 2 & \frac{3}{2} & 5 \end{array}\right] \qquad \textbf{c)}\left[\begin{array}{cc|c} 2 & \frac{3}{2} & 5 \\ 3 & 3 & 6 \end{array}\right]$$

Help Alice, Bob, and Charlotte finish solving the system of equations by writing the remaining row operations each of them must perform to bring their version of the augmented matrix into reduced row echelon form.

# Chapter 4

# Geometric aspects of linear algebra

In this section, we'll study geometric objects like lines, planes, and vector spaces. We'll use what we learned about vectors and matrices in the previous chapters to perform geometric calculations such as projections and distance measurements.

Developing your intuition about the geometric problems of linear algebra is very important: of all the things you learn in this course, your geometric intuition will stay with you the longest. Years from now, you may not recall the details of the Gauss–Jordan elimination procedure, but you'll still remember that the solution to three linear equations in three variables corresponds to the intersection of three planes in $\mathbb{R}^3$.

## 4.1 Lines and planes

*Points*, *lines*, and *planes* are the basic building blocks of geometry. In this section, we'll explore these geometric objects, the equations that describe them, and their visual representations.

### Concepts

- $p = (p_x, p_y, p_z)$: a *point* in $\mathbb{R}^3$
- $\vec{v} = (v_x, v_y, v_z)$: a *vector* in $\mathbb{R}^3$
- $\hat{v} = \frac{\vec{v}}{\|\vec{v}\|}$: the *unit vector* in the same direction as the vector $\vec{v}$
- An infinite line $\ell$ is a one-dimensional space defined in one of several possible ways:
  - ▷ $\ell : \{p_o + t\,\vec{v}, t \in \mathbb{R}\}$: a *parametric equation* of a line with direction vector $\vec{v}$ passing through the point $p_o$

▷ $\ell : \left\{ \frac{x-p_{ox}}{v_x} = \frac{y-p_{oy}}{v_y} = \frac{z-p_{oz}}{v_z} \right\}$: a *symmetric equation*

- An infinite plane $P$ is a two-dimensional space defined in one of several possible ways:
  ▷ $P : \{ Ax + By + Cz = D \}$: a *general equation*
  ▷ $P : \{ p_o + s\,\vec{v} + t\,\vec{w},\ s, t \in \mathbb{R} \}$: a *parametric equation*
  ▷ $P : \{ \vec{n} \cdot [(x, y, z) - p_o] = 0 \}$: a *geometric equation* of the plane that contains point $p_o$ and has normal vector $\hat{n}$
- $d(a, b)$: the shortest *distance* between geometric objects $a$ and $b$

## Points

We can specify a point in $\mathbb{R}^3$ by its coordinates $p = (p_x, p_y, p_z)$, which is similar to how we specify vectors. In fact, the two notions are equivalent: we can either talk about the destination point $p$ or the vector $\vec{p}$ that takes us from the origin to the point $p$. This equivalence lets us add and subtract vectors and points. For example, $\vec{d} = q - p$ denotes the displacement vector that takes the point $p$ to the point $q$.

We can also specify a point as the intersection of two lines. As an example in $\mathbb{R}^2$, let's define $p = (p_x, p_y)$ to be the intersection of the lines $x - y = -1$ and $3x + y = 9$. We must solve the two equations simultaneously to find the coordinates of the point $p$. We can use the standard techniques for solving equations to find the answer. The intersection point is $p = (2, 3)$. Note that for two lines to intersect at a point, the lines must not be parallel.

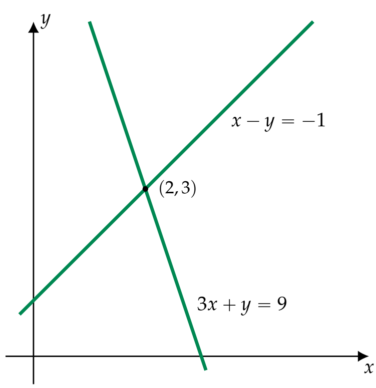



**Figure 4.1:** Two non-parallel lines in $\mathbb{R}^2$ intersect at a point.

**Example 1** Find where the lines $x + 2y = 5$ and $3x + 9y = 21$ intersect. To find the point of intersection, we solve these equations

**Distance between a line and the origin**

The closest distance between the line with equation $\ell : \{p_o + t\,\vec{v}, t \in \mathbb{R}\}$ and the origin $O = (0,0,0)$ is given by the formula

$$d(\ell, O) = \left\| p_o - \frac{p_o \cdot \vec{v}}{\|\vec{v}\|^2}\vec{v} \right\|.$$

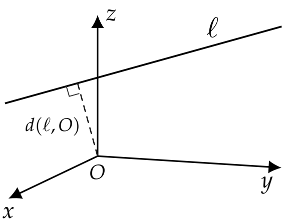



**Figure 4.8:** The closest distance between the line $\ell$ and the origin.

**Example 5** The closest distance between the line $\ell : \{(4,5,6) + t(1,0,1),\ t \in \mathbb{R}\}$ and the origin $O = (0,0,0)$ is calculated as follows:

$$\begin{aligned} d(\ell, O) &= \left\| (4,5,6) - \frac{(4,5,6) \cdot (1,0,1)}{1^2 + 0^2 + 1^2}(1,0,1) \right\| \\ &= \left\| (4,5,6) - \frac{4+0+6}{2}(1,0,1) \right\| \\ &= \|(-1,5,1)\| \ = \ 3\sqrt{3}. \end{aligned}$$

**Distance between a plane and the origin**

The closest distance between the plane with geometric equation $P : \vec{n} \cdot [(x,y,z) - p_o] = 0$ and the origin $O$ is given by

$$d(P, O) = \frac{|\vec{n} \cdot p_o|}{\|\vec{n}\|}.$$

For example, the distance between the plane $P : \quad (-3,0,-4) \cdot [(x,y,z) - (1,2,3)] = 0$ and the origin is computed as

$$d(P, O) = \frac{|(-3,0,-4) \cdot (1,2,3)|}{\|(-3,0,-4)\|} = \frac{|-3-12|}{5} = \frac{15}{5} = 3.$$

look for a one-size-fits-all formula for the different cases; derive the appropriate formula for each case starting from the basic projection operations $\Pi_\ell$ and $\Pi_P$.

# 4.2 Projections

In this section we'll learn to compute projections of vectors onto lines and planes. Given an arbitrary vector, we'll find how much of this vector points in a given direction (projection onto a line). We'll also find the part of the vector that lies in some plane (projection onto a plane). The dot product, $\vec{u} \cdot \vec{v} = u_1 v_1 + u_2 v_2 + u_3 v_3$, will play a central role in these calculations.

**Each projection formula corresponds to a vector diagram**. Vector diagrams, also known as "picture proofs," are used to describe the precise sequence of operations for computing a projection. Focussing on vector diagrams makes it much easier to understand projection and distance formulas. Indeed, the pictures in this section are a heck of a lot more important than the formulas. Be sure you understand each vector diagram, and don't worry about memorizing the corresponding formula. You can easily reproduce the formula by starting from the vector diagram.

## Concepts

- $S \subseteq \mathbb{R}^n$: $S$ is a *vector subspace* of $\mathbb{R}^n$. In this section, we assume $S \subseteq \mathbb{R}^3$. The subspaces of $\mathbb{R}^3$ are lines $\ell$ and planes $P$ that pass through the origin.
- $S^\perp$: the orthogonal complement to $S$, $S^\perp \stackrel{\text{def}}{=} \{\vec{w} \in \mathbb{R}^n \mid \vec{w} \cdot S = 0\}$. The symbol $^\perp$ stands for *perpendicular to*.
- $\Pi_S$: the *projection* onto the subspace $S$.
- $\Pi_{S^\perp}$: the projection onto the subspace $S^\perp$.

## Definitions

Let $S$ be a *vector subspace* of $\mathbb{R}^n$, denoted $S \subseteq \mathbb{R}^n$. In this section, we'll focus on the subspaces of the vector space $\mathbb{R}^3$ because they are easy to visualize and understand intuitively. The vector subspaces of $\mathbb{R}^3$ are lines and planes that pass through the origin. We defer the general discussion of subspaces in $n$ dimensions until Section 4.4.

The projection operation onto the subspace $S$ is a linear transformation that takes as inputs vectors in $\mathbb{R}^3$, and produces outputs in the subspace $S$:

$$\Pi_S : \mathbb{R}^3 \to S.$$

0, which shows the vector $\Pi_{\ell^\perp}(\vec{u})$ is indeed perpendicular to $\ell$. Adding the results of the two projections, we obtain the whole $\vec{u}$: $(\frac{16}{7}, \frac{32}{7}, \frac{48}{7}) + (\frac{12}{7}, \frac{3}{7}, \frac{-6}{7}) = (\frac{28}{7}, \frac{35}{7}, \frac{42}{7}) = (4, 5, 6) = \vec{u}$.

## Projection onto a plane

Now consider the two-dimensional plane $P$ passing through the origin with normal vector $\vec{n}$:

$$P: \quad \{(x, y, z) \in \mathbb{R}^3 \mid \vec{n} \cdot (x, y, z) = 0\}.$$

The perpendicular space $S^\perp$ is a line with direction vector $\vec{n}$:

$$P^\perp: \ \{(x, y, z) \in \mathbb{R}^3 \mid (x, y, z) = \vec{0} + t\,\vec{n}, \ t \in \mathbb{R}\}.$$

Again, the vector space $\mathbb{R}^3$ decomposes into the direct sum of $P$ and $P^\perp$: $\mathbb{R}^3 = P \oplus P^\perp$.

We want to find $\Pi_P$, but it will actually be easier to find $\Pi_{P^\perp}$ first and then compute $\Pi_P(\vec{u})$ using $\Pi_P(\vec{u}) = \vec{v} = \vec{u} - \vec{w}$, where $\vec{w} = \Pi_{P^\perp}(\vec{u})$. See Figure 4.11 for an illustration.

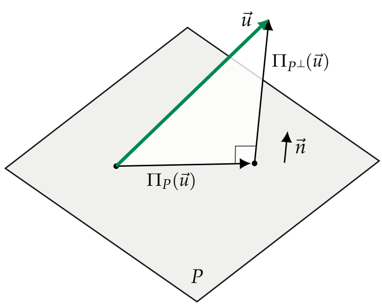



**Figure 4.11:** Any vector $\vec{u}$ can be written as the sum of two projections defined with respect to the plane $P$. The projection $\Pi_P(\vec{u})$ is parallel to the plane $P$, while the projection $\Pi_{P^\perp}(\vec{u})$ is perpendicular to the plane $P$.

Since $P^\perp$ is a line, we know the formula for projecting onto it is

$$\Pi_{P^\perp}(\vec{u}) = \frac{\vec{u} \cdot \vec{n}}{\|\vec{n}\|^2}\vec{n}.$$

We can now obtain the formula for $\Pi_P$:

$$\Pi_P(\vec{u}) = \vec{v} = \vec{u} - \vec{w} = \vec{u} - \Pi_{P^\perp}(\vec{u}) = \vec{u} - \frac{\vec{u} \cdot \vec{n}}{\|\vec{n}\|^2}\vec{n}.$$

# 4.3 Coordinate projections

In science, it's common to express vectors as components: $(v_x, v_y, v_z)$. Thinking of vectors as lists of components is fine for computational purposes, since vector operations require manipulating the components of vectors. However, focussing on a vector's components overlooks an important concept—the *basis* with respect to which the vector's components are expressed.

It's not uncommon for students to have misconceptions about linear algebra due to an incomplete understanding of the fundamental distinction between vectors and their components. Since I want you to have a *thorough* understanding of linear algebra, we'll review—in full detail—the notion of a basis and how to compute vector components with respect to different bases.

Before we begin, let's quickly review what we know about vectors and vector components from previous sections of the book. In Section 1.13, we described vectors in terms of their $x$- and $y$-components. Given a standard $xy$-coordinate system, we can decompose a vector $\vec{F}$ in terms of its components:

$$F_x = \|\vec{F}\| \cos\theta, \qquad F_y = \|\vec{F}\| \sin\theta,$$

where $\theta$ is the angle the vector $\vec{F}$ makes with the $x$-axis. We can express the vector as *coordinates* with respect to the basis $\{\hat{\imath}, \hat{\jmath}\}$ as $\vec{F} = F_x\hat{\imath} + F_y\hat{\jmath} = (F_x, F_y)_{\hat{\imath}\hat{\jmath}}$. The number $F_x$ corresponds to the length of the *projection* of the vector $\vec{F}$ onto the $x$-axis.

In the last section, we discussed the projection operation and learned how to compute projections using the dot product or as a matrix-vector product:

$$F_x\,\hat{\imath} \;=\; \frac{\vec{F}\cdot\hat{\imath}}{\|\hat{\imath}\|^2}\hat{\imath} \;=\; \Pi_x(\vec{F}) \;=\; \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}}_{M_{\Pi_x}} \underbrace{\begin{bmatrix} F_x \\ F_y \end{bmatrix}}_{\vec{F}},$$

where $\Pi_x : \mathbb{R}^2 \to \mathbb{R}^2$ is the projection onto the $x$-axis (a linear transformation) and $M_{\Pi_x}$ is the matrix representation of $\Pi_x$ with respect to the basis $\{\hat{\imath}, \hat{\jmath}\}$.

In this section, we'll extend what we know about basic vectors coordinates with respect to the basis $\{\hat{\imath}, \hat{\jmath}\}$, and formally define vector coordinates with respect to any basis.

## Concepts

We can define three different types of bases for an $n$-dimensional vector space $V$:

**E4.17** Consider the matrix

$$A = \begin{bmatrix} 1 & 3 & 3 & 3 \\ 2 & 6 & 7 & 6 \\ 3 & 9 & 9 & 10 \end{bmatrix}.$$

Find the RREF of $A$, and bases for $\mathcal{R}(A)$, $\mathcal{C}(A)$, and $\mathcal{N}(A)$.

## 4.6 Geometric problems

So far, we've defined all the important linear algebra concepts like vectors and matrices, and we've learned some useful computational techniques like the Gauss–Jordan elimination procedure. It's now time to apply what you've learned to solve geometric problems.

Points, lines, and planes can be difficult to understand and conceptualize. But now that you're armed with the tools of vectors, projections, and geometric intuition, you can solve all kinds of complicated geometric analysis problems—such as those waiting for you at the end of this paragraph. Remember to always sketch a diagram before you begin to write equations. Diagrams are great for visualizing and determining the steps you'll need to solve each problem.

**P4.1** Find the intersections of the these pairs of lines: **a)** $\ell_1$: $2x + y = 4$ and $\ell_2$: $3x - 2y = -1$, **b)** $\ell_1$: $y + x = 2$ and $\ell_2$: $2x + 2y = 4$, **c)** $\ell_1$: $y + x = 2$ and $\ell_2$: $y - x = 0$.

**P4.2** Find the lines of intersection between these pairs of planes: **a)** $P_1$: $3x - 2y - z = 2$ and $P_2$: $x + 2y + z = 0$, **b)** $P_3$: $2x + y - z = 0$ and $P_4$: $x + 2y + z = 3$.

**P4.3** Find whether the planes are parallel, perpendicular, or neither: **a)** $P_1$: $x - y - z = 0$ and $P_2$: $2x - 2y - 2z = 4$, **b)** $P_3$: $3x + 2y = 1$ and $P_4$: $y - z = 0$, **c)** $P_5$: $x - 2y + z = 5$ and $P_6$: $x + y + z = 3$.

**P4.4** Find the distance from the point $r = (2, 3, 5)$ to the plane $P$ defined by the equation $2x + y - 2z = 0$.

**P4.5** Find the closest distance between $p = (5, 3, 5)$ and $Q : 2x + y - 2z = 1$.

Hint: Consider an arbitrary point in the plane $Q$, such as $q = (0, 1, 0)$.

**P4.6** Find the distance between the points **a)** $p = (4, 7, 3)$ and $q = (1, 1, 1)$, **b)** $m = (4, -2, 0)$ and $n = (0, 1, 0)$, **c)** $r = (1, 0, 1)$ and $s = (-1, 1, -1)$, **d)** $i = (2, 1, 2)$ and $j = (1, -2, -1)$.

**P4.7** Find the general equation of the plane that passes through the points $q = (1, 3, 0)$, $r = (0, 2, 1)$, and $s = (1, 1, 1)$.

**P4.8** Find the symmetric equation of the line $\ell$ described by the equations

$$x = 2t - 3, \quad y = -4t + 1, \quad z = -t.$$

# Chapter 5

# Linear transformations

Linear transformations are a central idea of linear algebra—they form the cornerstone that connects all the seemingly unrelated concepts we've studied so far. We previously introduced linear transformations, informally describing them as "vector functions." In this chapter, we'll formally define linear transformations, describe their properties, and discuss their applications.

In Section 5.2, we'll learn how matrices can be used to *represent* linear transformations. We'll show the matrix representations of important types of linear transformations like projections, reflections, and rotations. Section 5.3 discusses the relation between bases and matrix representations. We'll learn how the bases chosen for the input and output spaces determine the entries of matrix representations. A single linear transformation can correspond to many different matrix representations, depending on the choice of bases for the input and output spaces.

Section 5.4 discusses and characterizes the class of *invertible linear transformations*. This section serves to connect several topics we covered previously: linear transformations, matrix representations, and the fundamental subspaces of matrices.

## 5.1 Linear transformations

Linear transformations take vectors as inputs and produce vectors as outputs. A transformation $T$ that takes $n$-dimensional vectors as inputs and produces $m$-dimensional vectors as outputs is denoted $T\colon \mathbb{R}^n \to \mathbb{R}^m$.

The class of linear transformations includes most of the useful transformations of analytical geometry: stretchings, projections, reflections, rotations, and combinations of these. Since linear transformations describe and model many real-world phenomena in physics, chemistry, biology, and computer science, learning the theory behind them is worthwhile.

## Concepts

Linear transformations are mappings between *vector inputs* and *vector outputs*. The following concepts describe the input and output spaces:

- $V$: the input vector space of $T$
- $W$: the output vector space of $T$
- $\dim(U)$: the dimension of the vector space $U$
- $T : V \to W$: a linear transformation that takes vectors $\vec{v} \in V$ as inputs and produces vectors $\vec{w} \in W$ as outputs. The notation $T(\vec{v}) = \vec{w}$ describes $T$ acting on $\vec{v}$ to produce the output $\vec{w}$.

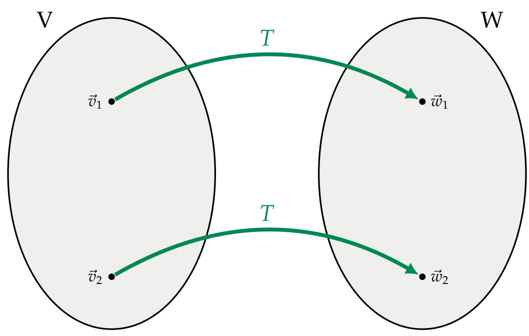



**Figure 5.1:** An illustration of the linear transformation $T : V \to W$.

- $\mathrm{Im}(T)$: the *image space* of the linear transformation $T$ is the set of vectors that $T$ can output for some input $\vec{v} \in V$. The mathematical definition of the image space is

$$\mathrm{Im}(T) \stackrel{\text{def}}{=} \{\vec{w} \in W \mid \vec{w} = T(\vec{v}), \text{ for some } \vec{v} \in V\} \quad \subseteq \quad W.$$

  The image space is the vector equivalent of the *image set* of a single-variable function $\mathrm{Im}(f) \stackrel{\text{def}}{=} \{y \in \mathbb{R} \,|\, y = f(x), \forall x \in \mathbb{R}\}$.

- $\mathrm{Ker}(T)$: the *kernel* of the linear transformation $T$; the set of vectors mapped to the zero vector by $T$. The mathematical definition of the kernel is

$$\mathrm{Ker}(T) \stackrel{\text{def}}{=} \{\vec{v} \in V \mid T(\vec{v}) = \vec{0}\} \quad \subseteq \quad V.$$

The kernel of a linear transformation is the vector equivalent of the roots of a function: $\{x \in \mathbb{R} \mid f(x) = 0\}$.

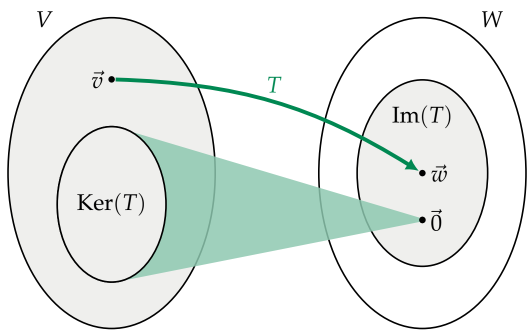



**Figure 5.2:** Two key properties of a linear transformation $T : V \to W$; its kernel $\text{Ker}(T) \subseteq V$, and its image space $\text{Im}(T) \subseteq W$.

**Example** The linear transformation $T : \mathbb{R}^2 \to \mathbb{R}^3$ is defined by the equation $T(x, y) = (x, y, x + y)$. Applying $T$ to the input vector $(1, 0)$ produces the output vector $(1, 0, 1 + 0) = (1, 0, 1)$. Applying $T$ to the input vector $(3, 4)$ produces the output vector $(3, 4, 7)$.

The kernel of $T$ contains only the zero vector $\text{Ker}(T) = \{\vec{0}\}$. The image space of $T$ is a two-dimensional subspace of the output space $\mathbb{R}^3$, namely $\text{Im}(T) = \text{span}((1, 0, 1), (0, 1, 1)) \subseteq \mathbb{R}^3$.

## Matrix representations

Given bases for the input and output spaces of a linear transformation $T$, the transformation's action on vectors can be represented as a matrix-vector product:

- $B_V = \{\vec{e}_1, \vec{e}_2, \ldots, \vec{e}_n\}$: a basis for the input vector space $V$
- $B_W = \{\vec{b}_1, \vec{b}_2, \ldots, \vec{b}_m\}$: a basis for the output vector space $W$
- $M_T \in \mathbb{R}^{m \times n}$: a matrix representation of the linear transformation $T$:
$$\vec{w} = T(\vec{v}) \qquad \Leftrightarrow \qquad \vec{w} = M_T\vec{v}.$$
  To be precise, we denote the matrix representation as ${}_{B_W}[M_T]_{B_V}$ to show it depends on the input and output bases.
- $\mathcal{C}(M_T)$: the *column space* of the matrix $M_T$
- $\mathcal{R}(M_T)$: the *row space* of the matrix $M_T$
- $\mathcal{N}(M_T)$: the *null space* the matrix $M_T$

## Properties of linear transformations

We'll start with the feature of linear transformations that makes them suitable for modelling a wide range of phenomena in science, engineering, business, and computing.

### Linearity

The fundamental property of linear transformations is—you guessed it—their *linearity*. If $\vec{v}_1$ and $\vec{v}_2$ are two input vectors and $\alpha$ and $\beta$ are two constants, then

$$T(\alpha\vec{v}_1 + \beta\vec{v}_2) = \alpha T(\vec{v}_1) + \beta T(\vec{v}_2) = \alpha\vec{w}_1 + \beta\vec{w}_2,$$

where $\vec{w}_1 = T(\vec{v}_1)$ and $\vec{w}_2 = T(\vec{v}_2)$.

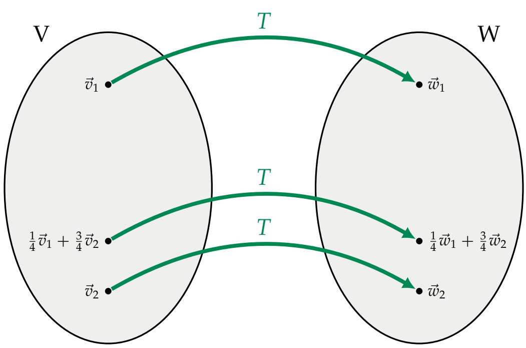



**Figure 5.3:** A linear transformation $T$ maps the linear combination of inputs $\frac{1}{4}\vec{v}_1 + \frac{3}{4}\vec{v}_2$ to the linear combination of outputs $\frac{1}{4}\vec{w}_1 + \frac{3}{4}\vec{w}_2$.

Linear transformations map any linear combination of inputs to the same linear combination of outputs. If you know the outputs of $T$ for the inputs $\vec{v}_1$ and $\vec{v}_2$, you can deduce the output $T$ for any linear combination of the vectors $\vec{v}_1$ and $\vec{v}_2$ by computing the appropriate linear combination of the outputs $T(\vec{v}_1)$ and $T(\vec{v}_2)$. This is perhaps the most important idea in linear algebra: it's the *linear* that we refer to when we talk about *linear algebra*. Linear algebra is not about lines, but about mathematical transformations that map linear combinations of inputs to the same linear combinations of outputs.

In this chapter, we'll study various aspects and properties of linear transformations, the abstract objects that map input vectors to output vectors. The fact that linear transformations map linear combinations of inputs to corresponding linear combinations of outputs will be of central importance in many calculations and proofs. Make a good note and store a mental image of the example shown in Figure 5.3.

## 5.2 Finding matrix representations

Every linear transformation $T : \mathbb{R}^n \to \mathbb{R}^m$ can be represented as a matrix $M_T \in \mathbb{R}^{m\times n}$. Suppose you're given the following description of a linear transformation: "$T$ is the counterclockwise rotation of all points in the $xy$-plane by 30°," and you want to find the matrix $M_T$ that corresponds to this transformation.

Do you know how to find the matrix representation of $T$? This section describes a simple and intuitive probing procedure for finding matrix representations. Don't worry; no alien technology is involved, and we won't be probing any humans—only linear transformations! As you read, try to bridge your understanding between the abstract, mathematical specification of a transformation $T(\vec{v})$ and its concrete implementation as a matrix-vector product $M_T\vec{v}$. We'll use the probing procedure to study various linear transformations and derive their matrix representations.

Once we find the matrix representation of a given transformation, we can efficiently apply that transformation to many vectors. This is exactly how computers carry out linear transformations. For example, a black-and-white image file can be represented as a long list that contains the coordinates of the image's black pixels: $\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_\ell\}$. The image is obtained by starting with a white background and drawing a black pixel in each of the locations $\vec{x}_i$ on the screen.[1] To rotate the image, we can process the list of pixels using the matrix-vector product $\vec{y}_i = M_T\vec{x}_i$, where $M_T$ is the matrix representation of the desired rotation. The transformed list of pixels $\{\vec{y}_1, \vec{y}_2, \ldots, \vec{y}_\ell\}$ corresponds to a rotated version of the image. This is essentially the effect of using the "rotate tool" in an image editing program—the computer multiplies the image by a rotation matrix.

### Concepts

The previous section covered linear transformations and their matrix representations:

- $T : \mathbb{R}^n \to \mathbb{R}^m$: a linear transformation that takes inputs in $\mathbb{R}^n$ and produces outputs in $\mathbb{R}^m$
- $M_T \in \mathbb{R}^{m\times n}$: the matrix representation of $T$

The action of the linear transformation $T$ is equivalent to multiplication by the matrix $M_T$:

$$\vec{w} = T(\vec{v}) \qquad \Leftrightarrow \qquad \vec{w} = M_T\vec{v}.$$

[1] Location on a computer screen is denoted using pixel coordinates $\vec{x}_i = (h_i, v_i)$. The number $h_i$ describes a horizontal distance measured in pixels from the left edge of the image, and $v_i$ measures the vertical distance from the top of the image.

## Theory

To find the matrix representation of the transformation $T\colon \mathbb{R}^n \to \mathbb{R}^m$, it is sufficient to probe $T$ with the $n$ vectors of the standard basis for the input space $\mathbb{R}^n$:

$$\hat{e}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \hat{e}_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \ldots, \quad \hat{e}_n = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}.$$

The matrix $M_T$ that corresponds to the action of $T$ on the standard basis is

$$M_T = \begin{bmatrix} | & | & & | \\ T(\vec{e}_1) & T(\vec{e}_2) & \cdots & T(\vec{e}_n) \\ | & | & & | \end{bmatrix}.$$

This is an $m \times n$ matrix whose columns are the outputs of $T$ for the $n$ probe inputs.

The remainder of this section illustrates the probing procedure for finding matrix representations of linear transformations.

## Projections

We'll start with a class of linear transformations you're already familiar with: *projections*. I hope you still remember what you learned in Section 4.2 (page 219).

### X projection

The projection onto the $x$-axis is denoted $\Pi_x$. The projection $\Pi_x$ acts on any vector or point by leaving the $x$-component unchanged and setting the $y$-component to zero. The action of $\Pi_x$ on two sample points is illustrated in Figure 5.5.

Let's analyze how the projection $\Pi_x$ transforms the two vectors of the standard basis:

$$\Pi_x\!\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \Pi_x\!\left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

The action of $\Pi_x$ on the basis $\hat{\imath}$ is to leave it unchanged. The action of $\Pi_x$ on the basis $\hat{\jmath}$ is to send it to the zero vector.

The matrix representation of $\Pi_x$ is therefore given by:

$$M_{\Pi_x} = \begin{bmatrix} \Pi_x\!\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right) & \Pi_x\!\left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}\right) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}.$$

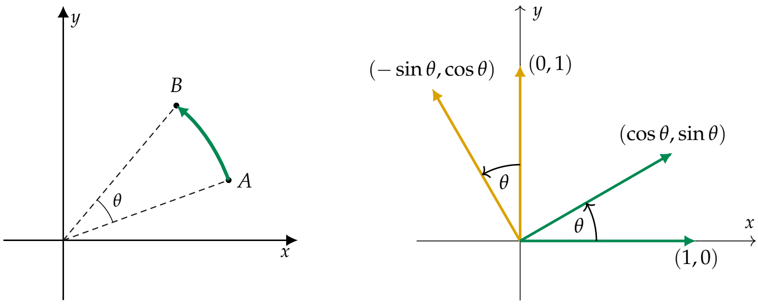



**Figure 5.11:** The linear transformation $R_\theta$ rotates every point in the plane by the angle $\theta$ in the counterclockwise direction. Note the effect of $R_\theta$ on the basis vectors $(1,0)$ and $(0,1)$.

To find the matrix representation of $R_\theta$, probe it with the standard basis as usual:

$$M_{R_\theta} = \left[ R_\theta\!\left(\begin{bmatrix}1\\0\end{bmatrix}\right) \;\; R_\theta\!\left(\begin{bmatrix}0\\1\end{bmatrix}\right) \right].$$

To compute the values in the first column, observe that $R_\theta$ rotates the vector $(1,0)^{\mathsf{T}} = 1\angle 0$ to the vector $1\angle\theta = (\cos\theta, \sin\theta)^{\mathsf{T}}$. The second input $\hat{e}_2 = (0,1)^{\mathsf{T}} = 1\angle\frac{\pi}{2}$ is rotated to $1\angle(\frac{\pi}{2}+\theta) = (-\sin\theta, \cos\theta)^{\mathsf{T}}$. Therefore, the matrix for $R_\theta$ is

$$M_{R_\theta} = \begin{bmatrix} | & | \\ 1\angle\theta & 1\angle(\frac{\pi}{2}+\theta) \\ | & | \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

Finding the matrix representation of a linear transformation is like a colouring-book activity for mathematicians. Filling in the columns is just like colouring inside the lines—nothing too complicated.

## Inverses

Can you determine the inverse matrix of $M_{R_\theta}$? You could use the formula for finding the inverse of a $2 \times 2$ matrix, *or* you could use the $[\,A \mid I\,]$-and-RREF algorithm for finding the inverse; but using either of these approaches would be *waaaaay* too much work. Try to guess the matrix representation of the inverse without doing any calculations. If $R_\theta$ rotates points by $+\theta$, can you tell me what the inverse operation does? I'll leave a blank line here to give you some time to think. . . .

## Change of basis for matrices

Given the matrix representation ${}_{B_W}[M_T]_{B_V}$ of the linear transformation $T : V \to W$, you're asked to find the matrix representation of $T$ with respect to different bases $B'_V$ and $B'_W$. This is the *change-of-basis* task for matrices.

We'll discuss the important special case where the input space and the output space of the linear transformation are the same. Let $T : V \to V$ be a linear transformation, and let $B = \{\hat{e}_1, \hat{e}_2, \ldots, \hat{e}_n\}$ and $B' = \{\hat{e}'_1, \hat{e}'_2, \ldots, \hat{e}'_n\}$ be two bases for the vector space $V$.

Recall the change-of-basis matrix ${}_{B'}[\mathbb{1}]_B$ that converts vectors from $B$ coordinates to $B'$ coordinates, and its inverse ${}_B[\mathbb{1}]_{B'}$, which converts vectors from $B'$ coordinates to $B$ coordinates:

$$[\vec{v}]_{B'} = {}_{B'}[\mathbb{1}]_B[\vec{v}]_B \qquad \text{and} \qquad [\vec{v}]_B = {}_B[\mathbb{1}]_{B'}[\vec{v}]_{B'}\,.$$

A clarification of notation is in order. The change-of-basis matrix ${}_{B'}[\mathbb{1}]_B$ is not equal to the identity matrix $\mathbb{1}_n$. However, the change-of-basis operation is *logically* equivalent to an identity transformation: the vector $\vec{v}$ doesn't change—only its coordinates change. If you don't remember the change-of-basis operation for vectors, now's the time to flip back to Section 4.3 (page 229) and review before continuing.

Given the matrix representation ${}_B[M_T]_B$ of the linear transformation $T$ with respect to $B$, we want to find the matrix ${}_{B'}[M_T]_{B'}$, which is the representation of $T$ with respect to the basis $B'$. The computation is straightforward. Perform the change-of-basis operation on the input and output vectors:

$${}_{B'}[M_T]_{B'} \;=\; {}_{B'}[\mathbb{1}]_B\;{}_B[M_T]_B\;{}_B[\mathbb{1}]_{B'}\,.$$

This group of three matrices is interpreted as follows. Imagine an input vector $[\vec{v}]_{B'}$ multiplying the three matrices ${}_{B'}[\mathbb{1}]_B\;{}_B[M_T]_B\;{}_B[\mathbb{1}]_{B'}$ from the right. In the first step, ${}_B[\mathbb{1}]_{B'}$ converts the vector from the basis $B'$ to the basis $B$ so the matrix ${}_B[M_T]_B$ can be applied. In the last step, the matrix ${}_{B'}[\mathbb{1}]_B$ converts the output of ${}_B[M_T]_B$ to the basis $B'$. The combined effect of multiplying by this specific arrangement of three matrices is the same as applying $T$ to the input vector $\vec{v}$:

$${}_{B'}[\mathbb{1}]_B\;{}_B[M_T]_B\;{}_B[\mathbb{1}]_{B'}[\vec{v}]_{B'} = [T(\vec{v})]_{B'}\,,$$

which means

$${}_{B'}[M_T]_{B'} = {}_{B'}[\mathbb{1}]_B\;{}_B[M_T]_B\;{}_B[\mathbb{1}]_{B'}\,.$$

This formula makes sense intuitively: to obtain a matrix with respect to a different basis, we must surround it by appropriate change-of-basis matrices.

**Invertible matrix theorem.** *For an $n \times n$ matrix $A$, the following statements are equivalent:*

(**1**) *$A$ is invertible*

(**2**) *The equation $A\vec{x} = \vec{b}$ has exactly one solution for each $\vec{b} \in \mathbb{R}^n$*

(**3**) *The null space of $A$ contains only the zero vector $\mathcal{N}(A) = \{\vec{0}\}$*

(**4**) *The equation $A\vec{x} = \vec{0}$ has only the trivial solution $\vec{x} = \vec{0}$*

(**5**) *The columns of $A$ form a basis for $\mathbb{R}^n$:*
- *The columns of $A$ are linearly independent*
- *The columns of $A$ span $\mathbb{R}^n$; $\mathcal{C}(A) = \mathbb{R}^n$*

(**6**) *The rank of the matrix $A$ is $n$*

(**7**) *The RREF of $A$ is the $n \times n$ identity matrix $\mathbb{1}_n$*

(**8**) *The transpose matrix $A^{\mathsf{T}}$ is invertible*

(**9**) *The rows of $A$ form a basis for $\mathbb{R}^n$:*
- *The rows of $A$ are linearly independent*
- *The rows of $A$ span $\mathbb{R}^n$; $\mathcal{R}(A) = \mathbb{R}^n$*

(**10**) *The determinant of $A$ is nonzero* $\det(A) \neq 0$

These 10 statements are either all true or all false for any given matrix $A$. We can split the set of $n \times n$ matrices into two disjoint subsets: invertible matrices, for which all 10 statements are true, and non-invertible matrices, for which all statements are false.

## Proving the invertible matrix theorem

It's essential that you understand the details of this theorem, including its proof. The reasoning that connects these 10 statements unites all the chunks of linear algebra we've discussed. Not being a "proof person" is not a valid excuse! Be sure to read the proof, as it will help to solidify your understanding of the material covered thus far.

### Proofs by contradiction

Since our arrival at the invertible matrix theorem marks an important step, we'll first quickly review some handy proof techniques, just to make sure everyone's ready. A *proof by contradiction* starts by assuming the opposite of the fact we want to prove, and after several

## Discussion

In this chapter, we learned about linear transformations and their matrix representations. The equivalence $T(\vec{x}) = M_T\vec{x}$ is important because it forms a bridge between the abstract notion of a linear transformation and its concrete implementation as a matrix-vector product. Everything you know about matrices can be applied to linear transformations, and everything you know about linear transformations can be applied to matrices. Which is mind-blowing, if you think about it.

We say $T$ is *represented* by the matrix ${}_{B_W}[M_T]_{B_V}$ with respect to the basis $B_V$ for the input space, and the basis $B_W$ for the output space. In Section 5.2, we learned about the probing procedure for finding matrix representations with respect to the standard basis, while Section 5.3 discussed the notion of change of basis for matrices. Hold tight, because in the next chapter we'll learn about the eigenvalues and eigenvectors of matrices and discuss the *eigendecomposition* of matrices, which is a type of change of basis.

Section 5.4 gave us the invertible matrix theorem along with a taste of what it takes to prove formal math statements. It's extra important that you attempt some of the proofs in the exercise section on page 293. Although proofs can be complicated, they're so worth your time because they force you to clarify the definitions and properties of all the math concepts you've encountered thus far. Attempt the proofs in the problems section to find out if you're a linear algebra amateur, or a linear algebra expert.

# 5.5 Linear transformations problems

Understanding linear transformations is extremely important for your overall understanding of linear algebra. This is why it's crucial for you to solve all the problems in this section. By working on these problems, you'll discover whether you really understand all the new material covered in this chapter. Remember in the book's introduction, when I mentioned that linear algebra is all about vectors and linear transformations? Well, if you can solve all the problems in this section, you're 80% of the way to understanding all of linear algebra.

**P5.1** Determine whether each of the following transformations are linear.

**a)** $T_1(x,y) = (y, x+y)$ **b)** $T_2(x,y) = (x+3, y-3)$
**c)** $T_3(x,y) = (|x|, |y|)$ **d)** $T_4(x,y,z) = (3x-2y+z, 2x+y-4z)$
**e)** $T_5(x) = (x, 2x, 3x)$ **f)** $T_6(x,y,z,w) = (5x, 4y, 3z, 2w, 1)$

If the transformation is linear, find its matrix representation. If the transformation is nonlinear, show a calculation where the linear property fails.

# Chapter 6

# Theoretical linear algebra

Let's take a trip down memory lane: 170 pages ago, we embarked on a mind-expanding journey through the land of linear algebra. We encountered vector and matrix operations. We studied systems of linear equations, solving them with row operations. We covered miles of linear transformations and their matrix representations. With the skills you've acquired to reach this point, you're ready to delve into the abstract, theoretical aspects of linear algebra—that is, since you know all the useful stuff, you can officially move on to the cool stuff.

In math, we often use abstraction to find the commonalities between different mathematical objects. These parallels give us a deeper understanding of the mathematical structures we compare. This chapter extends what we know about the vector space $\mathbb{R}^n$ to the realm of abstract vector spaces of vector-like mathematical objects (Section 6.3). We'll discuss linear independence, find bases, and count dimensions for these abstract vector spaces. We'll define abstract inner product operations and use them to generalize the concept of orthogonality for abstract vectors (Section 6.4). We'll explore the Gram–Schmidt orthogonalization procedure for distilling orthonormal bases from non-orthonormal bases (Section 6.5). Finally, we'll introduce vectors and matrices with complex entries (Section 6.7). Section 6.7 also reviews everything we've learned in this book, so be sure to read it even if complex numbers are not required for your course. Along the way, we'll develop a taxonomy for the different types of matrices according to their properties and applications (Section 6.2). We'll also investigate matrix decompositions—techniques for splitting matrices into products of simpler matrices (Section 6.6). The chapter begins by discussing the most important decomposition technique of them all: the *eigendecomposition*, which is a way to uncover the "natural basis" for any matrix.

## 6.1 Eigenvalues and eigenvectors

The set of eigenvectors of a matrix is a special set of input vectors for which the action of the matrix is described as a simple *scaling*. In Section 5.2, we observed how linear transformations act differently on different input spaces. We also observed the special case of the "zero eigenspace," called the *null space* of a matrix. The action of a matrix on the vectors in its null space is equivalent to a multiplication by zero. We'll now put these eigenvalues and eigenvectors under the microscope and see what more there is to see.

Decomposing a matrix in terms of its eigenvalues and its eigenvectors gives valuable insights into the properties of the matrix. Certain matrix calculations, like computing the power of the matrix, become much easier when we use the *eigendecomposition* of the matrix. For example, suppose we're given a square matrix $A$,

$$A = \begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix},$$

and we want to compute $A^7$. In other words, we want to compute

$$A^7 = \begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}\begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}\begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}\begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}\begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}\begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}\begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}.$$

That's an awful lot of matrix multiplications! Now imagine how many times we'd need to multiply the matrix if we wanted to find $A^{17}$ or $A^{77}$. Too many times, that's how many. Let's be smart about this. Every matrix corresponds to some linear operation. This means it's legit to ask, "What does the matrix $A$ do?" Once we figure that out, we can compute $A^{77}$ by simply doing what $A$ does 77 times.

The best way to see what a matrix does is to look inside it and see what it's made of (you may need to gradually gain the matrix's trust before it lets you do this). To understand the matrix $A$, you must find its *eigenvectors* and its *eigenvalues*. The word *eigen* is the German word for "self." The eigenvectors of a matrix are its "self vectors," and correspond to a natural choice of basis for describing the action of the matrix. The *eigendecomposition* is a change-of-basis operation that expresses the matrix $A$ with respect to its *eigenbasis* ("self-basis"). The eigendecomposition of the matrix $A$ is a product of three matrices:

$$A = \begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}}_{Q}\underbrace{\begin{bmatrix} 5 & 0 \\ 0 & 10 \end{bmatrix}}_{\Lambda}\underbrace{\begin{bmatrix} \frac{1}{5} & \frac{2}{5} \\ \frac{2}{5} & -\frac{1}{5} \end{bmatrix}}_{Q^{-1}} = Q\Lambda Q^{-1}.$$

You can multiply the three matrices $Q\Lambda Q^{-1}$ to obtain $A$. Note that the middle matrix $\Lambda$ (the capital Greek letter *lambda*) has entries only on the diagonal. The diagonal matrix $\Lambda$ is sandwiched between the matrix $Q$ on the left and $Q^{-1}$ (the inverse of $Q$) on the right.

The eigendecomposition of $A$ allows us to compute $A^7$ in a civilized manner:

$$\begin{aligned} A^7 &= AAAAAAA \\ &= Q\Lambda \underbrace{Q^{-1}Q}_{\mathbb{1}} \Lambda \underbrace{Q^{-1}Q}_{\mathbb{1}} \Lambda \underbrace{Q^{-1}Q}_{\mathbb{1}} \Lambda \underbrace{Q^{-1}Q}_{\mathbb{1}} \Lambda \underbrace{Q^{-1}Q}_{\mathbb{1}} \Lambda \underbrace{Q^{-1}Q}_{\mathbb{1}} \Lambda Q^{-1} \\ &= Q\Lambda\mathbb{1}\Lambda\mathbb{1}\Lambda\mathbb{1}\Lambda\mathbb{1}\Lambda\mathbb{1}\Lambda\mathbb{1}\Lambda Q^{-1} \\ &= Q\Lambda\Lambda\Lambda\Lambda\Lambda\Lambda\Lambda Q^{-1} \\ &= Q\Lambda^7 Q^{-1}. \end{aligned}$$

All the inner $Q^{-1}$s cancel with the adjacent $Q$s. How convenient! Since the matrix $\Lambda$ is diagonal, it's easy to compute its seventh power:

$$\Lambda^7 = \begin{bmatrix} 5 & 0 \\ 0 & 10 \end{bmatrix}^7 = \begin{bmatrix} 5^7 & 0 \\ 0 & 10^7 \end{bmatrix} = \begin{bmatrix} 78125 & 0 \\ 0 & 10\,000\,000 \end{bmatrix}.$$

Thus we can express our calculation of $A^7$ as

$$A^7 = \begin{bmatrix} 9 & -2 \\ -2 & 6 \end{bmatrix}^7 = \underbrace{\begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}}_{Q} \begin{bmatrix} 78125 & 0 \\ 0 & 10\,000\,000 \end{bmatrix} \underbrace{\begin{bmatrix} \frac{1}{5} & \frac{2}{5} \\ \frac{2}{5} & -\frac{1}{5} \end{bmatrix}}_{Q^{-1}}.$$

We still need to multiply these three matrices together, but we've cut the work from six matrix multiplications to two. The answer is

$$A^7 = Q\Lambda^7 Q^{-1} = \begin{bmatrix} 8015625 & -3968750 \\ -3968750 & 2062500 \end{bmatrix}.$$

With this technique, we can compute $A^{17}$ just as easily:

$$A^{17} = Q\Lambda^{17} Q^{-1} = \begin{bmatrix} 80000152587890625 & -39999694824218750 \\ -39999694824218750 & 20000610351562500 \end{bmatrix}.$$

We could even compute $A^{777} = Q\Lambda^{777}Q^{-1}$ if we wanted to. I hope by now you get the point: once you express $A$ in its *eigenbasis*, computing the powers of $A$ requires computing the powers of its *eigenvalues*, which is much simpler than carrying out matrix multiplications.

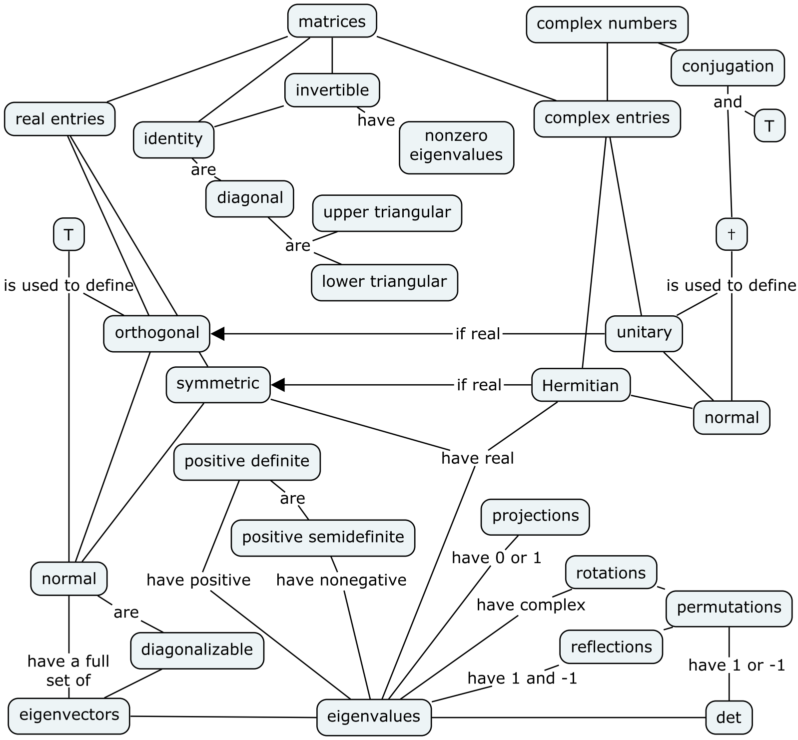



**Figure 6.1:** This concept map illustrates the connections and relations between special types of matrices. We can understand matrices through the constraint imposed on their eigenvalues or their determinants. This diagram shows only a subset of the many connections between the different types of matrices. We'll discuss matrices with complex entries in Section 6.7.

## Discussion

We've defined several special categories of matrices and described their properties. You're now equipped with some very precise terminology for describing different types of matrices. Each of these special matrices plays a role in certain applications.

This section also highlighted the importance of the eigenvalue description of matrices. Indeed, we can understand all special matrices in terms of the constraints imposed on their eigenvalues. The concept map in Figure 6.1 summarizes the relationships between the different special types of matrices. The map also refers to *unitary* and *Hermitian* matrices, which extend the concepts of *orthogonal* and *symmetric* matrices to describe matrices with complex entries.

## Exercises

**E6.8** Find the determinants and inverses of these triangular matrices:

$$A = \begin{bmatrix} 1 & 4 & 15 \\ 0 & 5 & 5 \\ 0 & 0 & 3 \end{bmatrix}, \qquad B = \begin{bmatrix} x & 0 \\ y & z \end{bmatrix}, \qquad C = \begin{bmatrix} \frac{1}{5} & 0 \\ 0 & 5 \end{bmatrix}.$$

**E6.9** Is the matrix $\left[\begin{smallmatrix} 1 & 2 \\ 3 & 4 \end{smallmatrix}\right]$ symmetric?

**E6.10** In this section we learned about different types of matrices: diagonal, triangular, positive (semi)definite, symmetric, and orthogonal matrices. What types of matrices are these?

$$\textbf{a)}\quad A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 1 \\ 0 & 0 & 7 \end{bmatrix} \qquad \textbf{b)}\quad B = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{3}} \\ 0 & \frac{2}{\sqrt{6}} & -\frac{1}{\sqrt{3}} \end{bmatrix} \qquad \textbf{c)}\quad C = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Hint: Compute the matrices' eigenvalues to analyse their properties.

# 6.3 Abstract vector spaces

You can apply your knowledge of vectors more generally to other vector-like mathematical objects. For example, polynomials behave similarly to vectors. To add two polynomials $P(x)$ and $Q(x)$, we add together the coefficients of each power of $x$—the same way vectors are added component by component.

In this section, we'll learn how to use the terminology and concepts associated with vectors to study other mathematical objects. In particular, we'll see that notions such as *linear independence*, *basis*, and *dimension* can be applied to mathematical objects like matrices, polynomials, and functions. We'll use the notation $\mathbf{v}$ for describing abstract vectors as opposed to the usual $\vec{v}$ used for ordinary vectors.

## Definitions

An abstract vector space $(V, F, +, \cdot)$ consists of four things:

- A set of vector-like objects $V = \{\mathbf{u}, \mathbf{v}, \ldots\}$
- A field $F$ of scalar numbers, usually $F = \mathbb{R}$
- An addition operation "$+$" for elements of $V$ that dictates how to add vectors: $\mathbf{u} + \mathbf{v}$
- A scalar multiplication operation "$\cdot$" for scaling a vector by an element of the field. Scalar multiplication is usually denoted implicitly $\alpha\mathbf{u}$ (without the dot).

A vector space satisfies the following eight axioms, for all scalars $\alpha, \beta \in F$ and all vectors $\mathbf{u}, \mathbf{v}, \mathbf{w} \in V$:

1. $\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$ (associativity of addition)
2. $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$ (commutativity of addition)
3. There exists a zero vector $\mathbf{0} \in V$, such that $\mathbf{u} + \mathbf{0} = \mathbf{0} + \mathbf{u} = \mathbf{u}$ for all $\mathbf{u} \in V$.
4. For every $\mathbf{u} \in V$, there exists an inverse element $-\mathbf{u}$ such that $\mathbf{u} + (-\mathbf{u}) = \mathbf{u} - \mathbf{u} = \mathbf{0}$.
5. $\alpha(\mathbf{u} + \mathbf{v}) = \alpha\mathbf{u} + \alpha\mathbf{v}$ (distributivity I)
6. $(\alpha + \beta)\mathbf{u} = \alpha\mathbf{u} + \beta\mathbf{u}$ (distributivity II)
7. $\alpha(\beta\mathbf{u}) = (\alpha\beta)\mathbf{u}$ (associativity of scalar multiplication)
8. There exists a unit scalar 1 such that $1\mathbf{u} = \mathbf{u}$.

If you know anything about vectors, the above properties should be familiar. Indeed, these are the standard properties for the vector space $\mathbb{R}^n$, where the field $F$ is $\mathbb{R}$, and for which standard vector addition and scalar multiplication operations apply.

## Theory

Believe it or not, we're actually done with all the theory for this section. Move along folks, there's nothing more to see here aside from the definitions above—which are restatements of the properties of vector addition and vector scaling that you've already seen before.

The only thing left to do is illustrate these concepts through some examples.

## Examples

Matrices, polynomials, and functions are vector-like math objects. The following examples demonstrate how we can treat these math objects as abstract vector spaces $(V, F, +, \cdot)$.

### Matrices

Consider the vector space of $m \times n$ matrices over the real numbers $\mathbb{R}^{m \times n}$. The addition operation for two matrices $A, B \in \mathbb{R}^{m \times n}$ is the usual rule of matrix addition: $(A + B)_{ij} = a_{ij} + b_{ij}$.

This vector space is $mn$-dimensional, which can be seen by constructing a basis for the space. The standard basis for $\mathbb{R}^{m \times n}$ consists basis for looking at a matrix. A diagonalizable matrix $A$ can be written as

$$A = Q\Lambda Q^{-1},$$

where $Q$ is a matrix whose columns are eigenvectors of $A$, and $\Lambda$ is a diagonal matrix containing the eigenvalues of $A$.

The eigendecomposition of a matrix is a similarity transformation (a change of basis) where the new basis matrix consists of eigenvectors of the matrix.

If $A$ is positive semidefinite then its eigenvalues are nonnegative. If the matrix $A$ is symmetric then its eigenvalues are real numbers.

When the matrix $A$ is *normal*, meaning it satisfies $AA^{\mathsf{T}} = A^{\mathsf{T}}A$, we can choose $Q$ to be an orthogonal matrix $O$ that satisfies $O^{\mathsf{T}}O = \mathbb{1}$. Calculating the inverse of an orthogonal matrix is easy: $O^{-1} = O^{\mathsf{T}}$. The eigendecomposition for normal matrices is $A = O\Lambda O^{\mathsf{T}}$.

## Singular value decomposition

We can generalize the concepts of eigenvalues and eigenvectors to non-square matrices. Consider a matrix $A \in \mathbb{R}^{m\times n}$. Since the matrix $A$ is not a square matrix, we can't use the standard eigendecomposition. However, there is a trick for turning a non-square matrix into a square matrix while preserving some of its properties: multiply the matrix by its transpose. The matrix $AA^{\mathsf{T}} \in \mathbb{R}^{n\times n}$ has the same column space as the matrix $A$. Similarly, $A^{\mathsf{T}}A \in \mathbb{R}^{m\times m}$ has the same row space as the matrix $A$.

The *singular value decomposition* breaks a matrix into the product of three matrices: an $m \times m$ orthogonal matrix $U$ which consists of *left singular vectors*, an $m \times n$ matrix $\Sigma$ with the *singular values* $\sigma_i$ on the diagonal, and an $n \times n$ orthogonal matrix $V^{\mathsf{T}}$ of *right singular vectors*:

$$A = \underbrace{\begin{bmatrix} | & & | \\ \hat{u}_1 & \cdots & \hat{u}_m \\ | & & | \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} \sigma_1 & 0 & \cdots \\ 0 & \sigma_2 & \cdots \\ 0 & 0 & \cdots \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} — & \hat{v}_1 & — \\ & \vdots & \\ — & \hat{v}_n & — \end{bmatrix}}_{V^{\mathsf{T}}} = U\Sigma V^{\mathsf{T}}.$$

To find the matrices $U$, $\Sigma$, and $V$, perform eigendecomposition on the matrix products $AA^{\mathsf{T}}$ and $A^{\mathsf{T}}A$.

Consider first the matrix $AA^{\mathsf{T}}$. Since $AA^{\mathsf{T}}$ is a square matrix, we can compute its eigendecomposition $AA^{\mathsf{T}} = U\Lambda_\ell U^{\mathsf{T}}$. The eigenvectors of $AA^{\mathsf{T}}$ span the same space as the column space of the matrix $A$. We call these vectors the *left singular vectors* of $A$.

The left singular vectors of $A$ (the columns of $U$) are the eigenvectors of the matrix $AA^{\mathsf{T}}$:

$$U = \begin{bmatrix} | & & | \\ \hat{u}_1 & \cdots & \hat{u}_m \\ | & & | \end{bmatrix}, \quad \text{where } \{(\lambda_i, \hat{u}_i)\} = \text{eigenvects}(AA^{\mathsf{T}}).$$

To find the right singular vectors of $A$ (the rows of $V^{\mathsf{T}}$), perform the eigendecomposition on the matrix $A^{\mathsf{T}}A$, denoted $A^{\mathsf{T}}A = V\Lambda_r V^{\mathsf{T}}$. Build the orthogonal matrix $V^{\mathsf{T}}$ by stacking the eigenvectors of $A^{\mathsf{T}}A$ as rows:

$$V^{\mathsf{T}} = \begin{bmatrix} — & \hat{v}_1 & — \\ & \vdots & \\ — & \hat{v}_n & — \end{bmatrix}, \quad \text{where } \{(\lambda_i, \hat{v}_i)\} = \text{eigenvects}(A^{\mathsf{T}}A).$$

The eigenvalues of the matrix $A^{\mathsf{T}}A$ are the same as the eigenvalues of the matrix $AA^{\mathsf{T}}$. In both cases, the eigenvalues $\lambda_i$ correspond to the squares of the singular values of the matrix $A$.

On its diagonal, the matrix of singular values $\Sigma \in \mathbb{R}^{m \times n}$ contains the singular values $\sigma_i$, which are the positive square roots of the eigenvalues $\lambda_i$ of the matrix $AA^{\mathsf{T}}$ (or the matrix $A^{\mathsf{T}}A$):

$$\sigma_i = \sqrt{\lambda_i}\,, \quad \text{where } \{\lambda_i\} = \text{eigenvals}(AA^{\mathsf{T}}) = \text{eigenvals}(A^{\mathsf{T}}A).$$

The singular value decomposition shows the inner structure of the matrix $A$. We can interpret the operation $\vec{y} = A\vec{x} = U\Sigma V^{\mathsf{T}}\vec{x}$ as a three-step process:

1. Convert the input $\vec{x}$ to the basis of right singular vectors $\{\vec{v}_i\}$.
2. Scale each component by the corresponding singular value $\sigma_i$.
3. Convert the output from the $\{\vec{u}_i\}$ basis to the standard basis.

This three-step procedure is analogous to the three-step procedure we used to understand the eigendecomposition of square matrices in Section 6.1 (see page 305).

* * *

The singular value decomposition (SVD) has numerous applications in statistics, machine learning, and computer science. Applying the SVD to a matrix is like looking inside it with X-ray vision, since you can see its $\sigma_i$s. The action of $A = U\Sigma V^{\mathsf{T}}$ occurs in $n$ parallel streams: the $i^{\text{th}}$ stream consists of multiplying the input vector

## 6.7 Linear algebra with complex numbers

So far we've discussed the math of vectors and matrices with real components. In fact, the linear algebra techniques you've learned apply to any *field* $F$. The term *field* applies to any mathematical object for which the operations of addition, subtraction, multiplication, and division are defined.

Since the complex numbers $\mathbb{C}$ are a field, we can perform linear algebra over the field of complex numbers. In this section, we'll define vectors and matrices with complex components, and discover that they behave similarly to their real counterparts. You'll see that complex linear algebra is no more complex than real linear algebra. It's the same, in fact, except for one small difference: instead of matrix transpose $A^{\mathsf{T}}$, we use the Hermitian transpose $A^{\dagger}$, which is the combination of the transpose and an entry-wise complex conjugate operation.

Complex vectors are not just an esoteric mathematical concept intended for specialists. Complex vectors can arise as answers for problems involving ordinary real matrices. For example, the rotation matrix

$$R_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

has complex eigenvalues $\lambda_1 = e^{i\theta}$ and $\lambda_2 = e^{-i\theta}$ and its eigenvectors have complex components. If you want to know how to calculate the eigenvalues and eigenvectors of rotation matrices, you need to understand how to do linear algebra calculations with complex numbers.

This section serves as a review of all the important linear algebra concepts we've learned in this book. I recommend you read this section, even if you're not required to know about complex matrices for your course. As your guide through the land of linear algebra, it's my duty to make sure you understand linear algebra in the complex field. It's good stuff; I guarantee there's *knowledge buzz* to be had in this section.

### Definitions

Recall the basic notions of complex numbers introduced in Section 1.14:

- $i$: the unit imaginary number; $i = \sqrt{-1}$ and $i^2 = -1$
- $z = a + bi$: a complex number $z$ whose real part is $a$ and whose imaginary part is $b$
- $\mathbb{C}$: the set of complex numbers $\mathbb{C} = \{a + bi \mid a, b \in \mathbb{R}\}$

## 6.8 Theory problems

It's now time to test your understanding of the theoretical concepts we discussed in this chapter. The eigenvector equation $A\vec{e}_\lambda = \lambda\vec{e}_\lambda$ is one of the deepest ideas in linear algebra. I've prepared several problems so you can challenge yourself and test your understanding of eigenvalues and eigenvectors. The problems will test your theoretical understanding as well as your stamina, because computing eigenvectors requires many steps of arithmetic and takes a long time. The first eigenvector problem you'll solve might take you up to an hour. Don't be alarmed by this—that's totally normal. After solving a few eigenvector problems, your problem-solving time will drop to 30 minutes; and quickly after that you'll able to solve eigenvalue problems easily in 15 minutes.

It's up to you how fluent you want to become. Certainly if you have a linear algebra exam coming up, it would be good to solve all the problems and maybe even solve problems in other books, too. If you're just reading about linear algebra for fun, you probably don't need to suffer through the steps of finding eigenvalues using only pen and paper. Solve the problems using `SymPy` instead—you can't say no to that!

In this chapter we also learned about abstract vector spaces, another important theoretical idea in linear algebra. All the techniques you've learned about vectors can be applied to polynomials, matrices, functions, and other vector-like objects. That's all nice in theory, but we're going to move beyond passive appreciation and get into the nitty gritty by solving problems that involve bases, linear independence, dimensions, and orthogonality in abstract vector spaces. It might seem like crazy stuff, but if you trust the idea of equivalent representations and the abstract notion of a linear transformation, you'll see it's all good and that you can work with abstract vectors.

Finally, the problems that involve linear algebra over the complex field will serve as the final review of what you've learned in this book. This is the final boss. You'll be asked to review and combine your computational, geometric, and theoretical linear algebra skills, applying them to vectors and matrices with complex coefficients. Are you ready for this?

I'm not going to lie to you and say the problems are easy, but this is the final push, so hang in there and you'll be done with all the linear algebra theory in just a few hours. After finishing the problems in this chapter, the rest of the book winds down with three chapters of cool applications, which are much lighter reading. So grab a pen, pull out some paper and kick some problem ass!

**P6.1** Yuna wants to cheat on her exam and she needs your help. Please help her compute the eigenvalues of the following matrices, and slip her the piece of paper carefully so the teacher doesn't notice. Yuna will give you a chocolate bar to thank you.

**a)** $\begin{bmatrix} 3 & 1 \\ 12 & 2 \end{bmatrix}$ **b)** $\begin{bmatrix} 0 & 1 & 0 \\ 2 & 0 & 2 \\ 0 & 1 & 0 \end{bmatrix}$

**P6.2** Find the eigenvalues of the following matrices.

**a)** $\begin{bmatrix} 4 & 2 \\ 0 & 5 \end{bmatrix}$ **b)** $\begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$ **c)** $\begin{bmatrix} 2 & 0 & 1 \\ 1 & 2 & 0 \\ 0 & 4 & -1 \end{bmatrix}$ **d)** $\begin{bmatrix} -3 & 0 & 0 \\ 4 & 1 & 0 \\ 2 & 1 & -1 \end{bmatrix}$

**P6.3** Compute the eigenvalues of the matrix $A = \left[\begin{smallmatrix} 1 & 1 \\ 1 & 0 \end{smallmatrix}\right]$.

**P6.4** Show that the vectors $\vec{e}_1 = (1, \frac{1}{\varphi})^{\mathsf{T}}$ and $\vec{e}_2 = (1, -\varphi)^{\mathsf{T}}$ are eigenvectors of the matrix $A = \left[\begin{smallmatrix} 1 & 1 \\ 1 & 0 \end{smallmatrix}\right]$. What are the eigenvalues associated with these eigenvectors?

Hint: Compute $A\vec{e}_1$ and $A\vec{e}_2$ to see what happens. Use the fact that $\varphi$ satisfies the equation $\varphi^2 - \varphi - 1 = 0$ to simplify expressions.

**P6.5** We can write the matrix $A = \left[\begin{smallmatrix} 1 & 1 \\ 1 & 0 \end{smallmatrix}\right]$ as the product of three matrices $Q\Lambda X$, where $Q$ contains the eigenvectors of $A$, and $\Lambda$ contains its eigenvalues:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 \\ \frac{1}{\varphi} & -\varphi \end{bmatrix}}_{Q} \underbrace{\begin{bmatrix} \varphi & 0 \\ 0 & \frac{-1}{\varphi} \end{bmatrix}}_{\Lambda} \underbrace{\begin{bmatrix} ? & ? \\ ? & ? \end{bmatrix}}_{X}.$$

Find the matrix $X$.

**P6.6** Compute the eigenvalues and eigenvectors of these matrices:

**a)** $A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ **b)** $B = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -1 & 4 \end{bmatrix}$

**P6.7** Given $A = \begin{bmatrix} 2 & 2 \\ 5 & -1 \end{bmatrix}$, find $A^{10}$.

**P6.8** Consider the sequence of triples $\{(x_n, y_n, z_n)\}_{n=0,1,2,\ldots}$ produced according to the formula:

$$\underbrace{\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}}_{M} \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} = \begin{bmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{bmatrix}.$$

Give a formula for $(x_\infty, y_\infty, z_\infty)$ in terms of $(x_0, y_0, z_0)$. This recurrence relation is related to "surface smoothing" algorithms used in 3D graphics; see `https://youtu.be/mX0NB9IyYpU` for more explanations.

# Chapter 7

# Applications

In this chapter, we'll learn about applications of linear algebra. We'll cover a wide range of topics from different areas of science, business, and technology to give you an idea of the spectrum of possible calculations based on vector and matrix algebra. Don't worry if you're not able to follow all the details in each section—we're taking a broad approach here, covering many different topics in the hope that some will interest you. Note that most of the material covered in this chapter is not likely to show up on your linear algebra final, so no pressure—this is just for fun.

Before we start, I want to say a few words about scientific ethics. Linear algebra is a powerful tool for solving problems and modelling the real world. But with great power comes great responsibility. I hope you'll make an effort to think about the ethical implications when you use linear algebra to solve problems. Certain applications of linear algebra, like building weapons, interfering with crops, and building mathematically-complicated financial scams are clearly evil, so you should avoid them. Other areas where linear algebra can be applied are not so clear-cut: perhaps you're building a satellite localization service to find missing people in emergency situations, but the same technology could be used by governments to spy on and persecute your fellow citizens. Do you want to be the person responsible for bringing about an Orwellian state? All I ask of you is to run a quick "System check" before you set to work on a project: ask yourself "Am I working for the System?" Don't just say "It's my job" and proceed without caution. If you find what you're doing for your employer is unethical, then maybe you should find a different job. There are a lot of jobs out there for people who know math, and if the bad guys can't hire qualified people like you, their power will decrease—and that's a good thing.

Our System check is complete. On to the applications!

## 7.1 Balancing chemical equations

Suppose you're given the chemical equation $H_2 + O_2 \to H_2O$, which indicates that hydrogen molecules ($H_2$) and oxygen molecules ($O_2$) can combine to produce water molecules ($H_2O$). Chemical equations describe how a set of *reactants* are transformed into a set of *products*. In this case, the reactants are hydrogen and oxygen molecules and the products are water molecules.

The equation $H_2 + O_2 \to H_2O$ is misleading since it doesn't tell us the correct *stoichiometric ratios*: how much of each type of molecule is consumed and produced. We say the equation is not *balanced*. To *balance* the equation, we must add coefficients in front of each reactant and each product, so that the total number of atoms on both sides of the reaction is the same: $2H_2 + O_2 \to 2H_2O$. Two hydrogen molecules are required for each oxygen molecule, since water molecules contain one oxygen and two hydrogen atoms.

Let's look at another example. The combustion of methane gas is described by the following chemical equation:

$$CH_4 + O_2 \;\to\; CO_2 + H_2O\,.$$

We want to answer the following two questions. How many molecules of oxygen will be consumed during the combustion of 1000 molecules of methane? How many $CO_2$ molecules will be produced as a result?

Before we can answer such questions, we must find the coefficients $a$, $b$, $c$, and $d$ that balance the methane-combustion equation:

$$aCH_4 + bO_2 \;\to\; cCO_2 + dH_2O.$$

For the equation to be balanced, the same number of atoms of each type must appear on each side of the equation. For the methane combustion reaction to be balanced, the following equations must be satisfied:

$$\begin{aligned} a &= c && \text{for } C \text{ atoms to be balanced,}\\ 4a &= 2d && \text{for } H \text{ atoms to be balanced,}\\ 2b &= 2c + d && \text{for } O \text{ atoms to be balanced.} \end{aligned}$$

We can move the $c$ and $d$ terms to the left side of each equation and rewrite the system of equations as a matrix equation:

$$\begin{array}{rcl} a \qquad - c \qquad\quad &=& 0\\ 4a \qquad\qquad - 2d &=& 0\\ 2b - 2c - d &=& 0 \end{array} \quad\Rightarrow\quad \underbrace{\begin{bmatrix} 1 & 0 & -1 & 0\\ 4 & 0 & 0 & -2\\ 0 & 2 & -2 & -1 \end{bmatrix}}_{A} \begin{bmatrix} a\\ b\\ c\\ d \end{bmatrix} = \begin{bmatrix} 0\\ 0\\ 0 \end{bmatrix}.$$

## 7.2 Input–output models in economics

Suppose you're the top economic official of a small country and you want to make a production plan for the coming year. For the sake of simplicity, let's assume your country produces only three commodities: electric power, wood, and aluminum. Your job is to choose the production rates of these commodities: $x_e$, $x_w$, and $x_a$. Your country must produce enough to satisfy both the internal demand and the external demand for these commodities. The problem is complicated because the production rates in one industry may affect the production rates of other industries. For instance, it takes some electric power to produce each unit of aluminum, so your production plan must account for both external demand for electric power, as well as *internal demand* for electric power for aluminum production. When complex interdependences exist between the different internal industries, as is often the case, it makes the process of picking the right production rates more complex.

In reality, most high-ranking government officials make their decisions about which industry to sponsor based on the dollar amounts of the kickbacks and bribes they received during the previous year. Let's ignore reality for a moment and assume you're an honest economist interested in using math to do what is right for the country instead of abusing your position of power like a blood-thirsty leech.

Let's assume the electric production $x_e$ must satisfy an external demand of 25 units, plus an additional 0.05 units for each unit of wood produced (electricity needed for saw mill operations) and an additional 0.3 units for each unit of aluminum produced. The wood production must be 10 units plus additional small amounts that depend on $x_e$ and $x_a$ (wood for construction). The production of aluminum must match 14 units of external demand plus an additional 0.1 units for each unit of electric power (for repairs of electric cables). We can model the interdependence between the industries using the following system of equations:

$$\begin{aligned}
x_e &= 25 + 0.05x_w + 0.3x_a \\
x_w &= 10 + 0.01x_e + 0.01x_a \\
x_a &= \underbrace{14}_{\text{external demand}} + \underbrace{0.1x_e}_{\text{internal demand}}.
\end{aligned}$$

You can use linear algebra to solve this complicated industry interdependence problem and choose appropriate production rates. Exwire junction must equal the total current flowing out of the junction. You can think of this as a manifestation of the *conservation of charge* principle: the total charge coming into a junction equals the total charge flowing out of the junction, because charges cannot be created or destroyed.

* * *

Recall that to solve a circuit is to find the currents that flow in each wire and the voltage across each resistor. We'll now illustrate how to solve a complicated circuit by introducing current variables, applying Kirchhoff's laws to obtain systems of linear equations, and solving these equations using linear algebra.

## Using linear algebra to solve circuits

The first step is to define variables for each of the quantities of interest in the circuit as shown in Figure 7.2. We'll call $I_1$ the current that flows down through the middle wire of the circuit, which then splits into the current $I_2$ in the left branch and the current $I_3$ going to the right. Next we follow the currents in the circuit and label the terminals of each resistor "+" and "−" to indicate its *polarity*—the direction of the voltage drop across it. The rule to follow is simple: the label "+" goes on the side where the current enters the resistor, and the label "−" goes on the side where the current leaves the resistor. We write the "+" on the side where the current enters because electric potential always *drops* when passing through a resistor. It's important to keep track of the polarity of resistors when writing the KVL equations for the circuit.

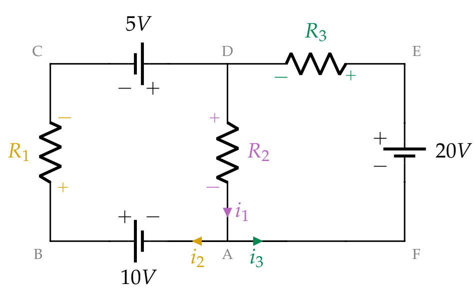



**Figure 7.2:** The circuit with branch currents labelled. Each resistor is assigned a *polarity* relative to the current flowing through it.

We're now in a position to apply Kirchhoff's voltage and current laws to this circuit and obtain a set of equations that relate the unknown currents. Let's first apply Kirchhoff's voltage law to the loop

# 7.4 Graphs

A *graph* is an abstract mathematical model that describes connections between a set of nodes. We call the nodes *vertices* and the connections *edges*. The graph is defined as a pair of sets $G = (V, E)$, where $V$ is the set of vertices and $E$ is the set of edges in the graph. We can also describe the edges by specifying the *adjacency matrix* of the graph.

Rather than define graphs formally and in detail, we'll look at a simple graph example to give you an idea of the main concepts and introduce graph notation. Figure 7.3 shows a small graph with five vertices and seven edges. This abstract link structure could represent many real-world scenarios: five websites and the hyperlinks between them, five Twitter accounts and their "following" relationships, or seven financial transactions between five businesses.

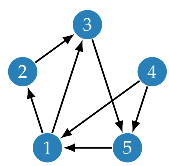



**Figure 7.3:** A simple graph with five vertices and seven edges.

The graph in Figure 7.3 is represented mathematically as $G = (V, E)$, where $V = \{1, 2, 3, 4, 5\}$ is the set of vertices, and $E = \{(1, 2), (1, 3),$ $(2, 3), (3, 5), (4, 1), (4, 5), (5, 1)\}$ is the set of edges. Note the edge from vertex $i$ to vertex $j$ is represented as the pair $(i, j)$.

## Adjacency matrix

The *adjacency matrix* representation of the graph in Figure 7.3 is a $5 \times 5$ matrix $A$ that contains information about the edges in the graph. Specifically, $A_{ij} = 1$ if the edge $(i, j)$ exists, otherwise $A_{ij} = 0$ if the edge doesn't exist:

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Each row contains ones in the positions where edges exist. The adjacency matrix representation works in tandem with the integer labels of the vertices—Vertex 1 corresponds to the first row of $A$, Vertex 2 to the second row, and so on for the other rows. We don't need labels for the vertices since the labels can be deduced from their position in the matrix $A$.

## 7.6 Linear programming

In the early days of computing, computers were primarily used to solve optimization problems, so the term "programming" is often used to describe optimization problems. *Linear programming* is the study of linear optimization problems that involve linear constraints. These types of optimization problems play an important role in business: the whole point of corporations is to constantly optimize profits subject to time, energy, and legal constraints.

Many optimization problems can be expressed as *linear programs*. To solve an optimization problem is to find the optimal value, which is either the maximum or the minimum of some function, called the *objective function*. In a linear program, the objective function is a multivariable linear function $g(x_1, x_2, \ldots, x_n)$ and the constraints on the variables of the problem are also linear. A linear program with $n$ variables and $m$ constraints is expressed as a maximization problem,

$$\max_{x_1,x_2,\ldots,x_n} g(x_1, x_2, \ldots, x_n) = c_1x_1 + c_2x_2 + \cdots + c_nx_n,$$

subject to $m$ linear constraints,

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &\leqslant b_1, \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &\leqslant b_2, \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &\leqslant b_m. \end{aligned}$$

The solution to this linear program is the vector $(x_1, x_2, \ldots, x_n)$ that makes $g(x_1, x_2, \ldots, x_n)$ as large as possible, and also satisfies all the constraints. For example, the variables $x_1, x_2, \ldots, x_n$ could represent the production rates of $n$ different products made by a company. If the coefficients $c_1, c_2, \ldots, c_n$ represent the selling price for each of the $n$ products, then $c_ix_i$ represents the revenue generated by selling $x_i$ units of product $i$, and $g(x_1, x_2, \ldots, x_n) = c_1x_1 + c_2x_2 + \cdots + c_nx_n$ represents the total revenue for a given choice of production rates. The $m$ inequalities could represent various limitations of human resources, production capacity, or logistics constraints. To solve the linear program is to find the production rates $x_1, x_2, \ldots, x_n$ that maximize revenue, subject to the constraints.

The *simplex algorithm* is a systematic procedure for finding solutions to linear programming problems. The simplex algorithm is somewhat similar to the Gauss–Jordan elimination procedure since it uses row operations on a matrix-like structure called a *tableau*. For this reason, linear programming and the simplex algorithm are often forced upon students taking a linear algebra course, especially business students. I'm not going to lie to you and tell you the simplex algorithm is very exciting, but it is very powerful. For this reason, you should know it exists, and develop a general intuition about how it works. And, as with all things corporate-related, it's worth learning about so you'll know the techniques of the enemy.

Since the details of the simplex algorithm might not be of interest to all readers of the book, I split the topic of linear programming into a separate tutorial, which you can read online at the link below.

[ Linear programming tutorial ]
`https://minireference.github.io/linear_programming/tutorial.pdf`

## 7.7 Least squares approximate solutions

An equation of the form $A\vec{x} = \vec{b}$ could have exactly one solution (if $A$ is invertible), infinitely many solutions (if $A$ has a null space), or no solution at all (if $\vec{b}$ is not in the column space of $A$). In this section, we'll discuss the case with no solution, and describe an approach for computing an *approximate* solution $\vec{x}^*$ such that the vector $A\vec{x}^*$ is as close as possible to $\vec{b}$.

We could jump right away to the formula for the least squares approximate solution ($\vec{x}^* = (A^{\mathsf{T}}A)^{-1}A^{\mathsf{T}}\vec{b}$), but this would hardly be enlightening or useful for your understanding. Instead, let's learn about the least squares approximate solution in the context of a machine learning problem in which we'll try to predict some unknown quantities based on a linear model learned from past observations. This is called *linear regression* and it's one of the most useful applications of linear algebra.

Your company's database of current clients contains all the information about the frequency of purchases $f$, value of purchases $V$, promptness of payment $P$, and other useful information. You know what is *really* useful information, though? Knowing the customer lifetime value (CLV)—the total revenue this customer will generate during their entire relationship with your company. You have data on the CLVs of existing customers and you want to leverage this data to predict the CLVs of new customers.

Suppose you have data for $N$ existing customers in the form of vectors, expressed as $\vec{a}_i = (f_i, V_i, P_i, \ldots)$, and you calculate a customer lifetime value (CLV) for each existing customer, $b_i = CLV_i$.

The resulting dataset consists of observations $\vec{a}_i$ and outcomes $b_i$:

$$D = \left\{ \begin{bmatrix} - & \vec{a}_1 & - \\ - & \vec{a}_2 & - \\ - & \vec{a}_3 & - \\ & \vdots & \\ - & \vec{a}_N & - \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_N \end{bmatrix} \right\} = \{A, \vec{b}\}.$$

The clients' observational data is stored in an $N \times n$ matrix $A$, and the corresponding CLVs are stored as an $N \times 1$ column vector $\vec{b}$.

**Statement of the problem** Given the $\vec{a}_k$ of a new customer, predict the customer's $b_k$, based on the information in the dataset $D$.

## Linear model

A simple way to model the dependence of the label $b_i$ on the observational data $\vec{a}_i = (a_{i1}, a_{i2}, \ldots, a_{in})$ is to use a linear model with $n$ parameters $m_1, m_2, \ldots, m_n$:

$$y_{\vec{m}}(x_1, x_2, \ldots, x_n) = m_1 x_1 + m_2 x_2 + \cdots + m_n x_n = \vec{m} \cdot \vec{x}.$$

If your model is accurate, then $y_{\vec{m}}(\vec{a}_i)$ will closely approximate $b_i$. But how can we measure the accuracy of the approximation?

Let's define *error terms* that measure how the model's predictions $y_{\vec{m}}(\vec{a}_i)$ differ from the observed values $b_i$. The error term for the $i^{\text{th}}$ customer is

$$e_i(\vec{m}) = |y_{\vec{m}}(\vec{a}_i) - b_i|^2.$$

The expression $e_i(\vec{m})$ measures the *squared error* between the model's prediction and the known value. Our goal is to choose a model that makes the sum of all the error terms as small as possible:

$$S(\vec{m}) = \sum_{i=1}^{N} e_i(\vec{m}) = \sum_{i=1}^{N} |y_{\vec{m}}(\vec{a}_i) - b_i|^2 .$$

We'll refer to the function $S(\vec{m})$ as the *objective function*, which is the usual terminology for optimization problems. Our goal is to choose the value of $\vec{m}$ that minimizes $S(\vec{m})$, the *sum of squared errors* of the model's predictions.

Intuitively, it makes sense to minimize the objective function $S(\vec{m})$ because it captures how the predictions of the model $y_{\vec{m}}(\vec{a}_i)$ differ from the values $b_i$. The objective function will be zero ($S(\vec{m}) = 0$) if the model *perfectly* predicts the data. On the other hand, any model prediction $y_{\vec{m}}(\vec{a}_i)$ that overshoots or undershoots the correct modelling power, they're easy to implement, and they lead to computational problems that are easy to solve.

## Links

[ Further discussion about least squares problems on Wikipedia ]
`https://en.wikipedia.org/wiki/Linear_regression`
`https://en.wikipedia.org/wiki/Linear_least_squares_(mathematics)`

[ More about the Moore–Penrose inverse ]
`https://en.wikipedia.org/wiki/Moore-Penrose_inverse`

## Exercises

**E7.10** You want to determine whether a coin is fair, so you toss it repeatedly and record the number of times it lands heads. On the first trial, you flip the coin eight times and obtain four heads, which is a heads-to-flips ratio of $\frac{4}{8}$. On subsequent trials you obtain heads-to-flips ratios of $\frac{9}{16}$, $\frac{13}{24}$, $\frac{17}{32}$, and $\frac{20}{40}$. Find the best-fitting linear model $h(x) = mx$ to describe the number of heads in a trial with $x$ flips.

**E7.11** Find the best-fitting affine model $y = b + mx$ to the $(x, y)$ data points $(0, 3.9)$, $(1, 3.2)$, and $(2, 1.9)$. Perform all the calculations by hand.

Hint: Find the Moore–Penrose inverse.

**E7.12** Calculate the total squared error $S(m^*) = \|Am^* - \vec{b}\|^2$ of the best-fit linear model obtained in Example 1 (page 383). Use the `SymPy` calculation at `bit.ly/leastsq_ex1` as your starting point.

Hint: The `Matrix` method `.norm()` might come in handy.

**E7.13** Revisit Example 2 (page 385) and find the total squared error of the best-fit affine model $S(\vec{m}'^*) = \|A\vec{m}'^* - \vec{b}\|^2$. You can start from the calculation provided here `bit.ly/leastsq_ex2` and extend it.

# 7.8 Computer graphics

Linear algebra is the mathematical language of computer graphics. Whether you're building a simple two-dimensional game with stick figures, or a fancy three-dimensional visualization, knowing linear algebra will help you understand the graphics operations that draw pixels on the screen.

In this section, we'll discuss some basic computer graphics concepts. In particular, we'll introduce *homogeneous coordinates*, a

The reason we need so much processing power is because 3D models are made of thousands of little polygons. Drawing a 3D scene (also known as *rendering*) involves performing linear algebra manipulations on all these polygons. This is where the GPU comes in. The job of the GPU is to translate, rotate, and scale the polygons of the 3D models by placing them into the scene, and then computing what the scene looks like when projected to the two-dimensional window (the screen) through which you're observing the virtual world. This transformation—from the model coordinates to world coordinates, and then to screen coordinates (pixels)—is carried out in a *graphics processing pipeline*.

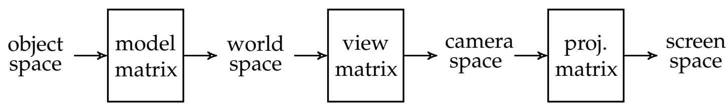



**Figure 7.12:** A graphics processing pipeline for drawing 3D objects on the screen. A 3D model is composed of polygons expressed with respect to a coordinate system centred on the object. The model matrix positions the object in the scene, the view matrix positions the camera in the scene, and finally the projection matrix computes what should appear on the screen.

We can understand the graphics processing pipeline as a sequence of matrix transformations: the model matrix $M$, the view matrix $V$, and the projection matrix $\Pi_s$. The GPU applies this sequence of operations to each of the object's vertices, $(x, y, z, 1)_o$, to obtain the pixel coordinates, $(x', y')_s$, of the vertices on the screen:

$$\begin{bmatrix} x' \\ y' \end{bmatrix}_s = \Pi_s V M \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_m \qquad \Rightarrow \qquad (x, y, z, 1)_m M^{\mathsf{T}} V^{\mathsf{T}} \Pi_s^{\mathsf{T}} = (x', y')_s.$$

In the context of computer graphics, it is customary to represent the graphics processing pipeline in the "transpose picture," so that vertex data flows from left to right as in Figure 7.12. Instead of representing vertices as column vectors multiplied by matrices $M$, $V$, and $\Pi_s$ from the left, we represent vertices as row vectors multiplied by matrices $M^{\mathsf{T}}$, $V^{\mathsf{T}}$, and $\Pi_s^{\mathsf{T}}$ from the right. All the reasoning remains the same, and all the transformation matrices described above still work; you just might need to transpose them if you're using them in a program.

Finally, a comment on efficiency. It is not necessary to compute the matrix-vector products with $M^{\mathsf{T}}$, $V^{\mathsf{T}}$, and $\Pi_s^{\mathsf{T}}$ for each vertex. It's much more efficient to pre-compute a combined transformation matrix, $C^{\mathsf{T}} = M^{\mathsf{T}} V^{\mathsf{T}} \Pi_s^{\mathsf{T}}$, and apply $C^{\mathsf{T}}$ to each of the coordinates of the links below if you want to learn more about homogeneous coordinates, projective spaces, or computer graphics.

### Links

[ A detailed tutorial series on WebGL ]
`https://github.com/greggman/webgl-fundamentals/`

[ Visualization of polygon drawing in a 3D scene ]
`http://orbides.org/apps/superslow.html`

## 7.9 Cryptography

Cryptography is the study of secure communication. The two main tasks that cryptographers aim to achieve are private communication (no eavesdroppers) and authenticated communication (no impersonators). Using algebraic operations over finite fields $\mathbb{F}_q$, it's possible to achieve both of these goals. Math is the weapon for privacy!

The need for private communication between people has existed long before the development of modern mathematics. Thanks to modern mathematical techniques, we can now perform cryptographic operations with greater ease, and build cryptosystems with security guaranteed by mathematical proofs. In this section, we'll discuss the famous *one-time pad* encryption technique invented by Gilbert Vernam. One-time pad encryption is important because Claude Shannon proved it is *absolutely secure*. In order to understand what that means precisely, we'll first need some context about the concepts studied in the field of cryptography.

### Context

The secure communication scenarios we'll discuss in this section involve three parties:

- Alice is the message sender
- Bob is the message receiver
- Eve is the eavesdropper

Alice wants to send a private message to Bob, but Eve has the ability to see all communication between Alice and Bob. You can think of Eve as a Facebook administrator, or an employee of the Orwellian, privacy-invading web application *du jour*. To defend against Eve, Alice will *encrypt* her messages before sending them to Bob, using a secret key only Alice and Bob have access to. Eve will be able to capture the encrypted messages (called *ciphertexts*) but they will be

## Discussion

We'll conclude with some basic advice for programmers who want to use cryptography in their software. Rather than trying to roll your own crypto functions, keep in mind it's best to use established libraries. Libraries are good for you—there are many ways cryptosystems can fail, and experts have thought about defending against many potential attacks. Don't be a cowboy programmer—use the libraries.

## Links

[ Signal is a secure messaging app for mobile ]
`https://whispersystems.org/`

[ Public-key cryptography general concepts ]
`https://en.wikipedia.org/wiki/Public-key_cryptography`

## Exercises

**E7.14** Alice wants to send the message $\vec{m} = 0110\ 1000\ 0110\ 1001$ to Bob. They have pre-shared the secret key $\vec{k} = 1010\ 0111\ 0010\ 0111$. Compute the ciphertext $\vec{c} = \text{Enc}(\vec{m}, \vec{k}) = \vec{m} \oplus \vec{k}$ that Alice will send to Bob. Verify that Bob will obtain the correct message after decrypting.

# 7.10 Error-correcting codes

The raw information-carrying capacity of a DVD is roughly 5.64GB; which is about 20% more than the 4.7GB of data that your computer will let you write to it. Why this overhead? Are DVD manufacturers trying to cheat you? Actually, they're looking out for you; the extra space is required for the *error-correcting code* that is applied to your data before writing it to the disk. Without the error-correcting code, even the tiniest scratch on the surface of the disk would make the disk unreadable, destroying your precious data. In this section, we'll learn how error-correcting codes work.

Error-correcting codes play an essential part in the storage, the transmission, and the processing of digital information. Even the slightest change to a computer program will make it crash—computer programs simply don't like it when you fiddle with their bits. Crashing was the norm back in the 1940s as illustrated by this quote:

> "Two weekends in a row I came in and found that all my stuff had been dumped and nothing was done. I was really annoyed because I wanted

codewords $\vec{c}_1 = 0010$ and $\vec{c}_2 = 0101$ is $d(\vec{c}_1, \vec{c}_2) = 3$, because it takes three substitutions (also called *bit flips*) to convert $\vec{c}_1$ to $\vec{c}_2$ or vice versa.

An $(n, k, d)$ code is defined by a function $\text{Enc} : \{0,1\}^k \to \{0,1\}^n$ that encodes messages $\vec{x}_i \in \{0,1\}^k$ into codewords $\vec{c}_i \in \{0,1\}^n$. Usually the encoding procedure Enc is paired with a decoding procedure, $\text{Dec} : \{0,1\}^n \to \{0,1\}^k$, which recovers messages from (possibly corrupted) codewords.

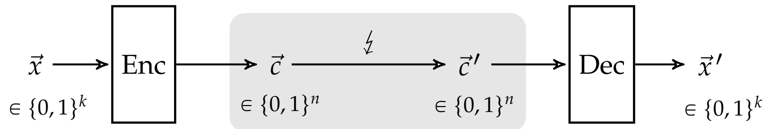



**Figure 7.14:** An error-correcting scheme using the encoding function Enc and the decoding function Dec to protect against the effect of noise (denoted ↯). Each message $\vec{x}$ is encoded into a codeword $\vec{c}$. The codeword $\vec{c}$ is transmitted through a *noisy channel* that can corrupt the codeword by transforming it into another bitstring $\vec{c}\,'$. The decoding function Dec looks for a valid codeword $\vec{c}$ that is close in Hamming distance to $\vec{c}\,'$. If the protocol is successful, the decoded message will match the transmitted message $\vec{x}\,' = \vec{x}$, despite the noise (↯).

## Linear codes

A code is *linear* if its encoding function Enc is a linear transformation:

$$\text{Enc}(\vec{x}_i + \vec{x}_j) = \text{Enc}(\vec{x}_i) + \text{Enc}(\vec{x}_j), \text{ for all messages } \vec{x}_i, \vec{x}_j.$$

An $(n, k, d)$ linear code encodes $k$-bit messages into $n$-bit codewords with minimum inter-codeword distance $d$. Linear codes are interesting because their encoding function Enc can be implemented as a matrix multiplication. We use the following terms when defining linear codes as matrices:

- $G \in \mathbb{F}_2^{k \times n}$: the *generating matrix* of the code. Each codeword $\vec{c}_i$ is produced by multiplying the message $\vec{x}_i$ by $G$ from the right:

$$\text{Enc}(\vec{x}_i) = \vec{c}_i = \vec{x}_i G.$$

- $\mathcal{R}(G)$: the row space of the generator matrix is called the *code space*. We say a codeword $\vec{c}$ is valid if $\vec{c} \in \mathcal{R}(G)$, which means there exists some message $\vec{x} \in \{0,1\}^k$ such that $\vec{x}G = \vec{c}$.
- $H \in \mathbb{F}_2^{(n-k) \times n}$: the *parity check matrix* of the code. The *syndrome* vector $\vec{s}$ of any bitstring $\vec{c}\,'$ is obtained by multiplying $\vec{c}\,'^{\mathsf{T}}$ by $H$

## 7.11 Fourier analysis

Way back in the 17th century, Isaac Newton carried out a famous experiment using light beams and glass prisms. He showed that when a beam of white light passes through a prism, it splits into a rainbow of colours: the rainbow is red at one end, followed by orange, yellow, green, blue, and finally violet at the other end. This experiment showed that white light is made of *components* with different colours. Using the language of linear algebra, we can say that white light is a "linear combination" of different colours.

Today we know that different colours of light correspond to electromagnetic waves with different frequencies: red light has a frequency around 450 THz, while violet light has a frequency around 730 THz. We can therefore say that white light is made of components with different frequencies. The notion of describing complex phenomena in terms of components with different frequencies is the main idea behind *Fourier analysis*.

Fourier analysis is used to describe sounds, vibrations, electric signals, radio signals, light signals, and many other phenomena. The Fourier transform allows us to represent all these "signals" in terms of components with different frequencies. Indeed, the Fourier transform can be understood as a change-of-basis operation that converts a signal from a time basis to a frequency basis:

$$[\mathbf{v}]_t \qquad \Leftrightarrow \qquad [\mathbf{v}]_f.$$

For example, if $\mathbf{v}$ represents a musical vibration, then $[\mathbf{v}]_t$ corresponds to the vibration as a function of time, while $[\mathbf{v}]_f$ corresponds to the frequency content of the vibration. Depending on the properties of the signal in the time domain and the choice of basis for the frequency domain, different Fourier transformations are possible.

We'll study three different bases for the frequency domain based on orthonormal sets of sinusoidal and complex exponential functions. The *Fourier series* is a representation for continuous periodic functions $f(t) \in \{\mathbb{R} \to \mathbb{R}\}$; that is, functions that satisfy $f(T + t) = f(t)$. The Fourier basis used in the Fourier series is the set of sines and cosines of the form $\sin(\frac{2\pi n}{T}t)$ and $\cos(\frac{2\pi n}{T}t)$, which form an orthogonal set. The *Fourier transform* is the continuous version of the Fourier series. Instead of a countable set of frequency components, the frequency representation of the signal is described by a complex-valued continuous function $f(\omega) \in \{\mathbb{R} \to \mathbb{C}\}$. Instead of a continuous time parameter $t \in \mathbb{R}$, certain signals are described in terms of $N$ *samples* from the time signal: $\{f[t]\}_{t\in[0,1,\ldots,N-1]}$. The *discrete Fourier transform* is a version of the Fourier transform for signals defined at discrete time samples. Table 7.2 shows a summary of these three

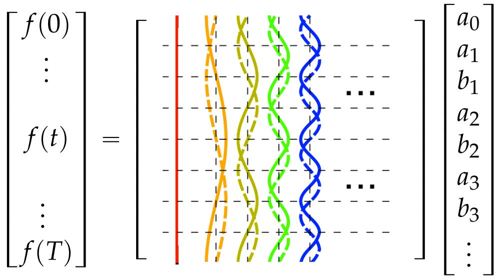



**Figure 7.18:** The Fourier series synthesis equation (FSS) represented as a matrix-vector product. A periodic signal $f(t)$ is represented as a series of Fourier coefficients $(a_0, a_1, b_1, a_2, b_2, a_3, b_3, \ldots)$. The first column of the change-of-basis matrix corresponds to the constant component $1 = \cos(0)$. The remaining columns correspond to cosines and sines with different frequencies.

This is called a *Fourier series approximation* since the frequency representation does not contain the components with frequencies $\frac{N+1}{T}$, $\frac{N+2}{T}$, and higher. Nevertheless, these finite-series approximations of signals are used in many practical scenarios; it's much easier to compute a finite number of Fourier coefficients instead of an infinite number.

**Example** For an example calculation of the Fourier coefficients of the *square wave* signal, see `bit.ly/fourier_series_square_wave` by Joshua Vaughan. Note the square wave analyzed is an *odd* function, so its coefficients $a_n$ are all zero.

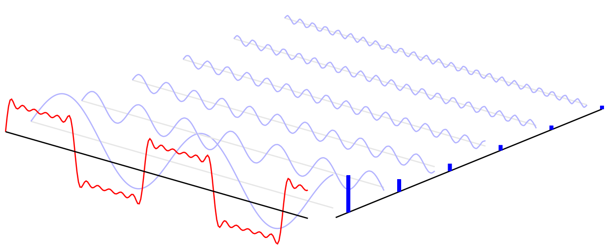


**Figure 7.19:** The square-wave signal can be approximated by a linear combination of sine functions with different frequencies.

In the next section, we'll describe the Fourier transform, which is a continuous-frequency version of the Fourier series.

[ Three compilations of linear algebra applications ]
http://aix1.uottawa.ca/~jkhoury/app.htm
https://medium.com/@jeremyjkun/633383d4153f
http://people.math.harvard.edu/~knill/teaching/math21b2018/handouts/use.pdf

[ A document that describes many applications in detail ]
http://www.geverstine.com/reprints/Everstine_linearalgebra.pdf

[ 33 miniatures: algorithmic applications of linear algebra ]
http://kam.mff.cuni.cz/~matousek/stml-53-matousek-1.pdf

[ A book about linear algebra applications to data science ]
http://amazon.com/Data-Science-from-Scratch/dp/149190142X

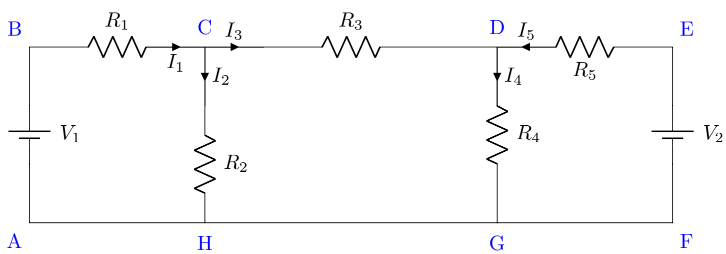
## 7.12 Applications problems

It would be easy to think of all the applications of linear algebra presented in this chapter as a TV program, designed to entertain rather than teach. Certainly you can continue to the next chapter without solving any problems, but do you really want to do that to yourself?

Presented next are a number of practice problems that will test your understanding of the new concepts and give you a great opportunity to practice your linear algebra skills. The linear algebra techniques we learned in previous chapters are key building blocks for applications. So don't sit on your laurels thinking, "Yay, I'm in Chapter 7 and I know linear algebra now, I'm so good." Prove it.

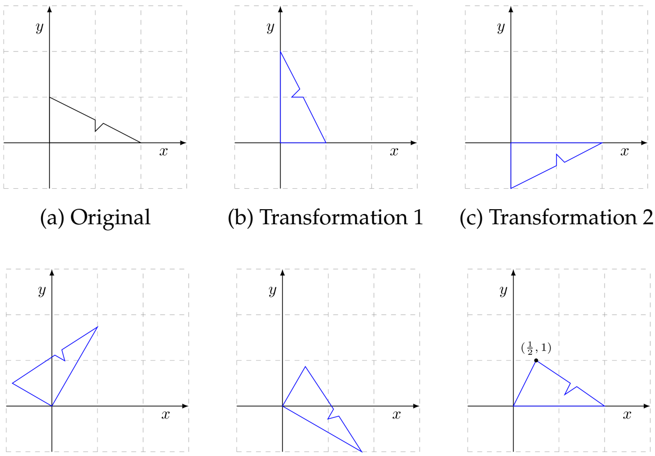
**P7.1** Consider the following chemical equation that describes how your body burns fat molecules: $C_{55}H_{104}O_6 + O_2 \rightarrow CO_2 + H_2O$. Balance this chemical equation.

**P7.2** Check out this circuit containing two batteries and five resistors:




**a)** Label the polarity of each resistor in the circuit.
**b)** Write three KVL equations and two KCL equations.
**c)** Rewrite the equations in the form $R\vec{I} = \vec{V}$, where $R$ is a $5 \times 5$ matrix, $\vec{I} = (I_1, I_2, I_3, I_4, I_5)^{\mathsf{T}}$, and $\vec{V}$ is a vector of constants.
**d)** Find the value of the currents $I_1$ and $I_5$ given $V_1 = 15[\text{V}]$, $V_2 = 10[\text{V}]$, $R_1 = 1[\Omega]$, $R_2 = 1[\Omega]$, $R_3 = 4[\Omega]$, $R_4 = 2[\Omega]$, $R_5 = 2[\Omega]$.

Hint: Your answers should be $3 \times 3$ matrices. Recall that $\sin(\frac{\pi}{6}) = \frac{1}{2}$.




(a) Original (b) Transformation 1 (c) Transformation 2

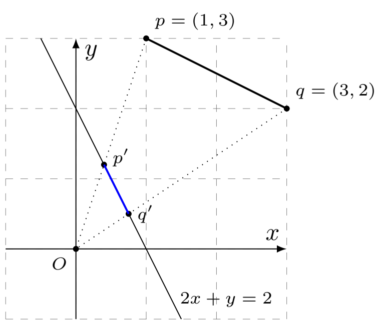
(d) Transformation 3 (e) Transformation 4 (f) Transformation 5

**Figure 7.20:** The effects of various linear transformations on a triangle.

**P7.11** Find the homogeneous coordinates representations of each of the transformations shown in Figure 7.21. The input to each transformation is the triangle with vertices $(0,0)$, $(2,1)$, $(0,1)$ shown in Figure 7.20 (a).

Hint: Use your answers from parts (b) and (d) to answer parts (e) and (f).

**P7.12** Find the homogeneous coordinates representation of the perspective transformation illustrated in the following figure. The observer is located at the origin $(0,0)$ and the projection line is given by the equation $2x + y = 2$.




Use the matrix to find the coordinates of the projected points $p'$ and $q'$.

**P7.13** Alice and Bob share the key $\vec{k}$ = 10010111 01010011 10011110. Bob receives the ciphertext $\vec{c}$ = 11100100 00100110 11101110 sent by Alice. What is the message sent by Alice? Find an ASCII lookup table on the web and use it to convert the binary message into characters.

# Chapter 8

# Probability theory

In this chapter, we'll use linear algebra concepts to explore the world of probability theory. Think of this as bonus material because the topics we'll discuss are not normally part of a linear algebra course. Given the general usefulness of probabilistic reasoning and the fact that you have already covered all the prerequisites, it would be a shame *not* to learn a bit about probability theory and its applications.

The chapter is structured as follows. In Section 8.1, we'll discuss probability distributions, which are mathematical models for describing random events. Section 8.2 introduces the concept of a *Markov chain*, which can be used to characterize the random transitions between different states of a system. Of the myriad of topics in probability theory, we've chosen to discuss probability distributions and Markov chains because they correspond one-to-one with vectors and matrices. This means you should feel right at home. In Section 8.3, we'll describe Google's PageRank algorithm for ranking webpages, which is an interesting application of Markov chains.

## 8.1 Probability distributions

Many phenomena in the world are inherently unpredictable. When you throw a six-sided die, one of the outcomes $\{1, 2, 3, 4, 5, 6\}$ will result, but you don't know which one. Similarly, when you toss a coin, you know the outcome will be either `heads` or `tails` but you can't predict which outcome will result. Probabilities are used to describe events where uncertainty plays a role. We can assign probabilities to the different outcomes of a dice roll, the outcomes of a coin toss, and also to many real-world systems. For example, we can build a probabilistic model of hard drive failures using past observations.

We can then calculate the probability that your family photo albums will survive the next 10 or 20 years. Backups my friends, backups.

Probabilistic models can help us better understand random events. The fundamental concept in probability theory is that of a *probability distribution*, which describes the likelihood of different outcomes of a random event. For example, the probability distribution for the roll of a fair die is $p_X = (\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6})^{\mathsf{T}}$, and the probability distribution for a coin toss is $p_Y = (\frac{1}{2}, \frac{1}{2})^{\mathsf{T}}$. Each entry of a probability distribution corresponds to the *probability mass* of a given outcome. This terminology borrows from the concept of mass distribution used in physics. The entries of a probability distribution satisfy the following conditions: each entry is a nonnegative number, and the sum of the entries is one. These two conditions are known as the *Kolmogorov axioms* of probability.

Strictly speaking, understanding linear algebra is not required for understanding probability theory. However, vector notation is very effective for describing probability distributions. Your existing knowledge of vectors and the rules for matrix multiplication will allow you to quickly understand many concepts in probability theory. Probabilistic reasoning is highly useful, so it's totally worth taking the time to learn about it.

## Random variables

A random variable $X$ is described by a probability distribution $p_X$. Before we formally define the notion of a probability distribution, we must introduce some formalism. We denote by $\mathcal{X}$ (calligraphic $X$) the *sample space* of the random variable $X$, which is the set of all possible outcomes of the random variable. A *discrete* random variable has a finite sample space. For example, we can describe the outcome of rolling a six-sided die using the random variable $X \in \mathcal{X}$, where the sample space is $\mathcal{X} = \{1, 2, 3, 4, 5, 6\}$. The number of possible outcomes is six: $|\mathcal{X}| = 6$.

We can describe the random outcome of a coin toss as a random variable $Y \in \{\texttt{heads}, \texttt{tails}\}$. The possible outcomes of the coin toss are $\mathcal{Y} = \{\texttt{heads}, \texttt{tails}\}$. The number of possible outcomes is two: $|\mathcal{Y}| = 2$.

In the case of the hard drive failure model, we can define the random variable $L \in \mathbb{N}$ as the years of a hard drive's lifetime before it fails. Using the random variable $L$ we can describe interesting scenarios using probabilistic reasoning. For example, the condition that a hard drive will function correctly for at least eight years can be described as $L \geqslant 8$.

## Probability distributions

The probability distribution of a discrete random variable $X \in \mathcal{X}$ is a vector of $|\mathcal{X}|$ nonnegative numbers that sum to one. Using mathematically precise notation, we write the definition of $p_X$ as follows:

$$p_X \in \mathbb{R}^{|\mathcal{X}|} \quad \text{such that} \quad p_X(x) \geqslant 0, \forall x \in \mathcal{X} \;\; \text{and} \;\; \sum_{x \in \mathcal{X}} p_X(x) = 1.$$

A probability distribution is a vector in $\mathbb{R}^{|\mathcal{X}|}$ that satisfies two special requirements: its entries must be nonnegative and the sum of the entries must be one.

## Events

In addition to the individual outcomes of a random variable, we can define *events* that consist of compositions of individual outcomes. Events correspond to different subsets of the sample space and are usually defined using words like "{event description}," which is the standard curly-bracket notation used to denote sets and subsets.

Recall the random variable $X$ that describes the outcome of rolling a six-sided die. The sample space for this random variable is $\mathcal{X} = \{1,2,3,4,5,6\}$. We can define various events that involve the six-sided die and describe them as subsets of the sample space. For example, the event $\{X \text{ is odd}\}$ corresponds to the subset $\{1,3,5\} \subset \mathcal{X}$.

Events are useful because they allow us to describe specific combinations of outcomes that are of practical interest. We can use the logical operators `OR` and `AND` to define composite events. The logical `OR` operator for events corresponds to the union operation for sets. For example, $X$ resulting in an odd number,

$$\{X \text{ is odd}\} = \{X = 1\} \;\texttt{OR}\; \{X = 3\} \;\texttt{OR}\; \{X = 5\},$$

corresponds to the union of sets $\{1\} \cup \{3\} \cup \{5\} = \{1,3,5\} \subset \mathcal{X}$.

The logical `AND` operator corresponds to the intersection of sets, and is often used to describe composite events that involve multiple random variables.

With these simple principles for describing probabilistic phenomena, we can build rich combinations of random events, expectations, and statistics—and develop methods for predicting the likelihood of future events. In the remainder of this section, we'll explore concepts of probabilistic reasoning through a series of examples.

In the next section, we'll learn about an application of conditional probability distributions that describes the state of a system undergoing random transitions between a number of possible states. This is a special case of a conditional probability distribution for which the random conditioning variable and random outcome variable are of the same dimension. Indeed, the outcome variable and the conditional space both represent states of the same system at different times.

## Links

[ Discussion on the Bayesian way of thinking about probabilities ]
`https://en.wikipedia.org/wiki/Bayesian_probability`

[ Detailed discussion about Bayes rule ]
`http://yudkowsky.net/rational/bayes`
`http://yudkowsky.net/rational/technical`

## Exercises

**E8.1** Do the following vectors represent probability distributions?

**a)** $\left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)^{\mathsf{T}}$ **b)** $\left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)^{\mathsf{T}}$ **c)** $(0.3, 0.3, -0.1, 0.5)^{\mathsf{T}}$

**E8.2** Compute the expected value $\mu_X$ and the variance $\sigma_X^2$ of the random variable $X$ that describes the outcome of rolling a six-sided die. The probability distribution of $X$ is $p_X = (\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6})^{\mathsf{T}}$.

# 8.2 Markov chains

So far, we've talked about random events without any reference to the flow of time. In this section we'll combine the idea of random variables with the notion of time. A *random process* is a model of a system that undergoes transitions between states over time. The state of the system at time $t$ is described by a random variable $X_t$. We obtain the next state of the system $X_{t+1}$ from the following conditional probability distribution:

$$p_{X_{t+1}|X_t X_{t-1}\cdots X_0}(x_{t+1}|x_t x_{t-1}\cdots x_0).$$

The random variable $X_{t+1}$ depends on the previous states of the system: $X_t$, $X_{t-1}$, $X_{t-2}$, $\cdots$, $X_0$. Studying such history-dependent processes is a formidable task due to the myriad of possible influences

$p_{X_{t+1}|X_t}(x_{t+1}|x_t)$ describe how the next state of the ball's possession $x_{t+1}$ depends on the previous state of the ball's possession $x_t$. The transition matrix of the Markov chain in our current example is

$$p_{X_{t+1}|X_t} = \begin{bmatrix} 0.2 & 0.1 & 0.\overline{3} \\ 0.4 & 0.8 & 0.\overline{3} \\ 0.4 & 0.1 & 0.\overline{3} \end{bmatrix} = M.$$

To maintain consistency with the notation for conditional probability distributions, we refer to the entries of $M$ as $p_{X_{t+1}|X_t}(x_{t+1}|x_t)$. The "given" variable $x_t$ selects the column of the matrix $M$, and the different entries in this column represent the transition probabilities for that state. Using the matrix $M$ and some basic linear algebra techniques, we can calculate the probability of finding the ball in any given player's possession after many iterations of the "pass the ball" Markov process.

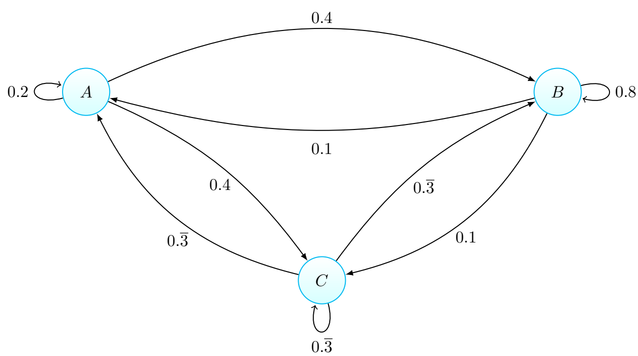



**Figure 8.1:** A representation of the transition probabilities between three states: "Alice has the ball," "Bob has the ball," and "Charlie has the ball."

Let's walk through an example calculation in which we assume the ball starts in Alice's possession. Since we know that Alice has the ball at $t = 0$, we can describe the initial state of the system by the probability distribution $p_{X_0} = (1, 0, 0)^{\mathsf{T}}$, with 100% of the probability on Alice. We obtain the probability of finding the ball in each player's possession after one time step, or unit of time, by multiplying the initial probability vector $p_{X_0}$ by the matrix $M$:

$$\begin{aligned} p_{X_1} &= M p_{X_0} \\ &= \begin{bmatrix} 0.2 & 0.1 & 0.\overline{3} \\ 0.4 & 0.8 & 0.\overline{3} \\ 0.4 & 0.1 & 0.\overline{3} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.2 \\ 0.4 \\ 0.4 \end{bmatrix}. \end{aligned}$$

**E8.4** Go to `https://live.sympy.org` and create a `Matrix` object describing the passing-the-ball Markov chain transition probabilities. Use the `Matrix` methods `.eigenvects()` and `.nullspace()` to confirm the stationary distribution of the Markov chain is $\left(\frac{5}{31}, \frac{20}{31}, \frac{6}{31}\right)$.

Hint: You can create a `Matrix` object using

```
>>> M = Matrix([[ 2/10, 1/10, 1/3 ],
                [ 4/10, 8/10, 1/3 ],
                [ 4/10, 1/10, 1/3 ]])
```

If you start from a matrix with exact rational entries, you'll obtain the exact answer in terms of rational numbers. You can also use the function `S` (short for `sympify`) to create rationals: `S(1/3)` = $\frac{1}{3}$.

Hint: The eigenspace that corresponds to the eigenvalue $\lambda = 1$ is the null space of the matrix $(M - \mathbb{1}_3)$. Use the `SymPy` command `eye(3)` to create a $3 \times 3$ identity matrix $\mathbb{1}_3$, then apply the `nullspace` method.

Hint: Use `vec.norm(1)` to compute the $\ell^1$-norm of the vector `vec`.

**E8.5** Find the stationary distribution of the following Markov chain:

$$C = \begin{bmatrix} 0.8 & 0.3 & 0.2 \\ 0.1 & 0.2 & 0.6 \\ 0.1 & 0.5 & 0.2 \end{bmatrix}.$$

## 8.3 Google's PageRank algorithm

Consider the information contained in the **links between webpages**. Each link from Page A to Page B can be interpreted as a recommendation by Page A's author for the contents of Page B. In web-speak we say links from Page A to Page B are "sending eyeballs" to Page B, presumably because there is something interesting to see on Page B. These observations about "eyeball worthiness" are the inspiration behind Google's PageRank algorithm. We find a good summary of the idea behind PageRank in Google's 2001 patent application:

> A method assigns importance ranks to nodes in [...] the world wide web or any other hypermedia database. The rank assigned to a document is calculated from the ranks of documents citing it. In addition, the rank of a document is calculated from a constant representing the probability that a browser through the database will randomly jump to the document. — Patent `US6285999`

## Links

[ The original PageRank paper ]
`http://ilpubs.stanford.edu/422/1/1999-66.pdf`

[ Further discussion about the PageRank algorithm ]
`https://en.wikipedia.org/wiki/PageRank`

[ The *power iteration* method for finding the PageRank eigenvector ]
`https://en.wikipedia.org/wiki/Power_iteration`

## Exercises

**E8.6** Compute the PageRank vector for the network of webpages shown in Figure 7.3 on page 373.

Hint: Use `https://live.sympy.org` and define the matrix $M_1$ using the syntax `Matrix([[...], ...])`, then mix it with an all-ones matrix `ones(n,n)`. Use the `nullspace` method from E8.4 to obtain the eigenvector in the $\lambda = 1$ eigenspace.

# 8.4 Probability problems

To better understand random variables and probability distributions, you need to practice using these concepts to solve real-world problems. It just so happens there are some practice problems on this very topic in this section—how convenient is that? Don't skip them!

Solving practice problems will help you understand probability theory and Markov chains. If you haven't played with `SymPy` yet, now is a great chance to get to know this powerful computer algebra system because Markov chain calculations are difficult to do by hand.

**P8.1** Given a random variable $X$ with three possible outcomes $\{1, 2, 3\}$ and probability distribution $p_X = (p_1, p_2, p_3)$, prove that $p_1 \leqslant 1$.

Hint: Use the Kolmogorov's axioms and build a proof by contradiction.

**P8.2** The probability of `heads` for a fair coin is $p = \frac{1}{2}$. The probability of getting `heads` $n$ times in a row is given by the expression $p^n$. What is the probability of getting `heads` four times in a row?

**P8.3** You have a biased coin that lands on `heads` with probability $p$, and consequently lands on `tails` with probability $(1 - p)$. Suppose you want to flip the coin until you get `heads`. Define the random variable $N$ as the number of tosses required until the first `heads` outcome. What is the probability mass function $P_N(n)$ for success on the $n^{\text{th}}$ toss? Confirm that the formula is a valid probability distribution by showing $\sum_{n=1}^{\infty} P_N(n) = 1$.

Hint: Find the probabilities for cases $n = 1, 2, 3, \ldots$ and look for a pattern.

# Chapter 9

# Quantum mechanics

By the end of the 19$^{\text{th}}$ century, physicists thought they had figured out most of what there is to know about the laws of nature. Newton's laws of mechanics described the motion of objects in space, and Maxwell's equations described electricity and magnetism. Wave phenomena—including the propagation of sound, light, and waves on the surface of liquids—were also well understood. Only a few small inconsistencies between theory and experiments with atoms and radiation remained unsolved.

> "[...] it seems probable that most of the grand underlying principles have now been firmly established and that further advances are to be sought chiefly in the rigorous application of these principles to all the phenomena which come under our notice."
>
> —Albert A. Michelson in 1894

Physicists like Michelson were worried about the future of physics research. It was as if they were wondering, "What are we going to do now that we've figured everything out?" Little did they know about the quantum storm that was about to hit physics, and with it, the complete rewrite of our understanding of nature at the smallest scale.

Understanding the structure of atoms—the smallest constituents of matter known at the time—was no trivial task. Describing the absorption of electromagnetic radiation by metals also turned out to be quite complicated. In both cases, the physical theories of the time predicted that the energy of physical systems could take on any value; yet experimental observations showed *discrete* energy levels. Imagine you throw a (very-very tiny) ball, and the laws of physics force you to choose an initial velocity for the ball from a list of "allowed" values: 0 m/s, 1 m/s, 2 m/s, 3 m/s, and so forth. That would be weird, no? Weird indeed, and this is the situation physicists were facing in the beginning of the $20^{\text{th}}$ century: their theories described the energy levels of atoms as real numbers $E \in \mathbb{R}$, but experiments showed that only a discrete set of energy levels exist. For example, the energy levels that the electrons of the hydrogen atom can take on are:

$$E \in \{\, 21.8\times10^{-19}\,\text{J},\ 5.44\times10^{-19}\,\text{J},\ 2.42\times10^{-19}\,\text{J},\\ 13.6\times10^{-20}\,\text{J},\ 8.71\times10^{-20}\,\text{J},\ 6.05\times10^{-20}\,\text{J},\ \ldots\,\}.$$

Other experimental observations suggested that electromagnetic radiation is not a continuous wave, but comes in discrete "wave packets," which we call *photons* today. The theory of *quantum mechanics* was born out of a need to explain these observations. The term *quantum*, from the Latin *quantus* for quantity, was coined to describe the discrete nature of the phenomena that physicists were trying to explain.

During the first half of the $20^{\text{th}}$ century, in experiment after experiment, quantum principles were used to correctly predict many previously-unexplained observations. During the second half of the $20^{\text{th}}$ century, biologists, chemists, engineers, and physicists applied quantum principles to all areas of science. This process of "upgrading" classical models to quantum models led to a better understanding of the laws of nature, and the discovery of useful things like transistors and lasers.

The fundamental principles of quantum mechanics can be explained in the space on the back of an envelope. Understanding quantum mechanics is a matter of combining a little knowledge of linear algebra (vectors, inner products, projections) with some probability theory (Chapter 8). In this chapter, we'll take a little excursion to the land of physics to learn about the ideas of great scientists like Bohr, Planck, Dirac, Heisenberg, and Pauli. Your linear algebra skills will allow you to learn about some fascinating $20^{\text{th}}$-century discoveries. This chapter is totally optional reading, reserved for readers who *insist* on learning about the quantum world. If you're not interested in quantum mechanics, it's okay to skip this chapter, but I recommend you check out Section 9.3 on *Dirac notation* for vectors and matrices. Learning Dirac notation serves as an excellent review of the core concepts of linear algebra.

## 9.1 Introduction

The principles of quantum mechanics have far-reaching implications for many areas of science: physics, chemistry, biology, engineering, philosophy, and many other fields of study. Each field of study has its own view on quantum mechanics, and has developed a specialized language for describing quantum concepts. We'll formally introduce the postulates of quantum mechanics in Section 9.5, but before we get there, let's look at some of the disciplines where quantum principles are used.

**Physics** Physicists use the laws of quantum mechanics as a toolbox to understand and predict the outcomes of atomic-scale physics experiments. By "upgrading" classical physics models to reflect the ideas of quantum mechanics, physicists (and chemists) obtain more accurate models that lead to better predictions.

For example, in a *classical* physics model, the motion of a particle is described by its position $x(t)$ and velocity $v(t)$ as functions of time:

$$\text{classical state} = (x(t), v(t)), \text{ for all times } t.$$

At any given time $t$, the particle is at position $x(t)$ and moving with velocity $v(t)$. Using Newton's laws of motion and calculus, we can predict the position and the velocity of a particle at all times.

In a quantum description of the motion of a particle in one dimension, the state of a particle is represented by a *wave function* $|\psi(x,t)\rangle$, which is a complex-valued function of position $x$ and time $t$:

$$\text{quantum state} = |\psi(x,t)\rangle, \text{ for all times } t.$$

At any given time $t$, the state of the particle corresponds to a complex-valued function of a real variable $|\psi(x)\rangle \in \{\mathbb{R} \to \mathbb{C}\}$. The wave function $|\psi(x)\rangle$ is also called the *probability-amplitude* function. The probability of finding the particle at position $x_a$ is proportional to the value of the squared norm of the wave function:

$$\Pr(\{\text{particle position} = x_a\}) \quad \propto \quad \big||\psi(x_a)\rangle\big|^2.$$

Instead of having a definite position $x(t)$ as in the classical model, the position of the particle in a quantum model is described by a probability distribution calculated from its wave function $|\psi(x)\rangle$. Instead of having a definite momentum $p(t)$, the momentum of a quantum particle is another function calculated based on its wave function $|\psi(x)\rangle$.

Classical models provide accurate predictions for physics problems involving macroscopic objects, but fail to predict the physics of atomic-scale phenomena. Much of 20$^{\text{th}}$-century physics research efforts were dedicated to the study of quantum concepts like ground states, measurements, spin angular momentum, polarization, uncertainty, entanglement, and non-locality.

**Computer science** Computer scientists understand quantum mechanics using principles of information. Quantum principles impose a fundamental change to the "data types" used to represent information. Classical information is represented as *bits*, elements of the finite field of size two $\mathbb{Z}_2$:

$$\text{bit: } x = 0 \text{ or } x = 1.$$

In the quantum world, the fundamental unit of information is the *qubit*, which is a two-dimensional unit vector in a complex inner product space:

$$\text{qubit: } |x\rangle = \alpha|0\rangle + \beta|1\rangle.$$

This change to the underlying information model requires reconsidering fundamental information processing tasks like computation, data compression, encryption, and communication.

**Philosophy** Philosophers have also updated their conceptions of the world to incorporate the laws of quantum mechanics. Observations of physics experiments forced them to reconsider the fundamental question, "What are things made of?" Another interesting question philosophers have considered is whether the quantum state $|\psi\rangle$ of a physical system really exists, or if $|\psi\rangle$ is a representation of our knowledge about the system.

A third central philosophy concept that quantum mechanics calls into question is *determinism*—the clockwork-model of the universe, where each effect has a cause we can trace, like the connections between gears in a mechanical clock. The laws of physics tell us that the next state of the universe is determined by the current state of the universe, and the state changes according to the equations of physics. However, representing the universe as a quantum state has implications for our understanding of how the universe "ticks." Clockwork (deterministic) models of the universe are not wrong—they just require a quantum upgrade.

Many scientists are also interested in the philosophical aspects of quantum mechanics. Physicists call these types of questions *foundations* or *interpretations*. Since different philosophical interpretations of quantum phenomena cannot be tested experimentally, these questions are considered outside the scope of physics research. Nevertheless, these questions are so deep and fascinating that physicists continue to pursue them, and contribute interesting philosophical work.

[ Philosophical issues in quantum theory ]
`http://plato.stanford.edu/entries/qt-issues/`

## Physical models of the world

We'll situate our discussion of quantum mechanics within two conceptual worlds:

- The **real world** is where physical experiments are performed.
- The **mathematical world** is a purely theoretical construct that aims to model certain aspects of the real world.

The better the mathematical model, the more closely its predictions correspond to the behaviour of real-world systems. Sill, no math or physics model can ever predict real-world outcomes with 100% accuracy. When we say that a certain mathematical model is better than another, we mean it can predict the outcomes of controlled experiments with greater accuracy. Physicists are very open to new theories—anyone can be a physicist! You can start by constructing any crazy mathematical model for describing nature, and if your model correctly predicts the outcomes of experiments, other physicists will start using it.

We can make a further distinction among mathematical models, classifying them into two categories depending on the type of math they use:

- **Classical models** describe the world in terms of real variables like positions and velocities.
- **Quantum models** describe systems in terms of vectors in complex vector spaces.

Table 9.1 compares the objects used in the two types of mathematical models of the real world. In physics, classical models describe the motion of particles using trajectories $\vec{r}(t)$, whereas quantum models use wave functions $|\psi(\vec{r},t)\rangle$. In computer science, classical information is stored in bits $i \in \{0,1\}$, whereas quantum information is stored in qubits $|x\rangle \in \mathbb{C}^2$.

**Example** Let's analyze the difference between classical and quantum models of the real world using an example. Consider a photon (a particle of light) going through an optical circuit that consists of several lenses, mirrors, and other optical instruments. A photon detector is placed at position $x_f$ at the end of the circuit. The objective of the experiment is to predict if the photon will arrive at the detector and cause it to "click." The two possible outcomes of the experiment are `click` (photon arrives at detector) or `noclick` (photon doesn't arrive at detector).[1]

[1] We're assuming the detector has 100% efficiency (detects every photon that arrives at it) and generates zero noise (no false-positive clicks).

**Real world:**

- The motion of a ball thrown in the air
- The motion of an electron through space
- The paths of light particles moving through optical circuits
- The electric current flowing though a superconducting loop

| **Classical models:** | **Quantum models:** |
|---|---|
| • $x(t) \in \{\mathbb{R} \to \mathbb{R}\}$ | • $\lvert\psi(x,t)\rangle \in \{\mathbb{R} \times \mathbb{R} \to \mathbb{C}\}$ |
| • $\vec{r}(t) \in \{\mathbb{R} \to \mathbb{R}^3\}$ | • $\lvert\psi(\vec{r},t)\rangle \in \{\mathbb{R}^3 \times \mathbb{R} \to \mathbb{C}\}$ |
| • $i \in \mathbb{Z}_2 = \{0,1\}$ (a bit) | • $\lvert x\rangle \in \mathbb{C}^2$ (a qubit) |
| • $j \in \mathbb{Z}_d$ | • $\lvert y\rangle \in \mathbb{C}^d$ |

**Table 9.1:** Examples of the math used in classical and quantum models.

A classical model of the motion of the photon calculates the photon's position at all times $x(t)$ and leads to the prediction $i = 1$ (`click`) if $x_f = x(t)$, for some $t$. On the other hand, if the detector does not lie on the photon's trajectory, then the classical model will predict $i = 0$ (`noclick`).

A quantum model would describe the photon's trajectory through the circuit as a linear combination of two different possible paths:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad \text{where} \quad |\alpha|^2 + |\beta|^2 = 1.$$

Here $|1\rangle$ describes paths that arrive at the detector, and $|0\rangle$ describes paths that don't. The coefficients $\alpha$ and $\beta$ describe the relative "weights" of the different paths. Using the quantum model, we can obtain a probabilistic prediction of whether the detector will click or not:

$$\Pr(\texttt{noclick}) = |\alpha|^2 \quad \text{and} \quad \Pr(\texttt{click}) = |\beta|^2.$$

For this example, both the classical and the quantum models describe the same real-world phenomenon. We can test the validity of both models by comparing the models' predictions with what happens in reality.

Note that the two models make very different assumptions about reality. The classical model assumes the photon follows a single path through the circuit, whereas the quantum model assumes the photon can take multiple paths through the circuit. Despite the difference in the mathematical substrate of the models and their fundamentally different views of reality, we can compare the two models' predictions on the same footing. Note it doesn't make sense to say one model is more real than the other. The only thing that is real is the photon in the optical circuit, and the photon doesn't care whether you use classical or quantum models to describe its path.

## Quantum model peculiarities

We'll now comment on the relative "intuitiveness" of classical and quantum models and introduce the concept of *quantum measurement*, which is of central importance in quantum mechanics.

Classical models have the advantage of being more intuitively understandable than quantum models. The variables in classical models often correspond to measurable aspects of real-world systems. We can identify the position variable in a classical model with the position of a particle in the real world. Velocity and momentum are harder to understand intuitively, but we have some general intuition about motion and collisions from everyday life. In general, we can understand classical models more readily because it's easier for us to think about a mechanistic, clockwork-like universe, in which objects push on each other with clearly defined cause and effect, like a clock that goes click, click, click.

In contrast, we do not enjoy such intuitive interpretation of quantum models, since our senses cannot directly perceive movement and interaction at the quantum level. Because quantum models' states do not directly correspond to observable aspects in the real world, quantum models are often described as mysterious and counterintuitive. Quantum models are harder to understand in part because they use complex vector quantities to represent systems, and complex numbers are more difficult to visualize. For example, visualizing the complex-valued state of a photon $|\psi\rangle$ is difficult, since you must think about both the real part and the imaginary part of $|\psi\rangle$. Even though we can't see what $|\psi\rangle$ looks like, we can describe it using an equation, and do math calculations with it. In particular, we can compare the predictions obtained from calculations based on the quantum state $|\psi\rangle$ to measurements performed in the real world.

The process of *quantum measurement* is how we map the predictions of the quantum model to observable quantities. A quantum measurement acts on a particle's wave function $|\psi\rangle$ to produce a classical outcome. **Performing quantum measurements is like asking questions of particles, and the measurement outcomes are the answers to these questions.**

$$\begin{aligned}
\text{What is your position?} \quad &\Leftrightarrow \quad \text{position}(|\psi\rangle) = x \in \mathbb{R} \\
\text{What is your momentum?} \quad &\Leftrightarrow \quad \text{momentum}(|\psi\rangle) = p \in \mathbb{R} \\
\text{What is your spin momentum?} \quad &\Leftrightarrow \quad \text{spin}_{\uparrow\downarrow}(|\psi\rangle) = s \in \{\uparrow, \downarrow\}
\end{aligned}$$

Since measurement outcomes correspond to real-world quantities that can be measured, we can judge the merits of quantum models the same way we judge the merits of classical models—in terms of the quality of their predictions.

### Chapter overview

In the next section, we'll describe a tabletop experiment involving lasers and polarization lenses, with an outcome that's difficult to explain using classical physics. The remainder of the chapter will introduce the tools needed to explain the outcome of this experiment in terms of quantum mechanics. We'll start by introducing a special notation for vectors that is used to describe quantum phenomena (Section 9.3).

In Section 9.5, we'll formally define the "rules" of quantum mechanics, also known as the *postulates* of quantum mechanics. We'll learn the "rules of the game" using the simplest possible quantum systems (qubits), and define how quantum systems are prepared, how we manipulate them using *quantum operations*, and how we extract information from them using *quantum measurements*. This part of the chapter is based on the notes from the introductory lectures of a graduate-level quantum information course, so don't think you'll be getting some watered-down, hand-wavy version of quantum mechanics. You'll learn the real stuff, because I know you can handle it.

In Section 9.6 we'll apply the quantum formalism to the polarizing lenses experiment, showing that a quantum model leads to the correct qualitative and quantitative prediction for the observed outcome. We'll close the chapter with short explanations of different applications of quantum mechanics with pointers for further exploration about each topic.

Throughout the chapter, we'll focus on *matrix* quantum mechanics and use computer science language to describe quantum phenomena. A computer science approach allows us to discuss the fundamental aspects of quantum theory without introducing all the physics required to understand atoms. Finally, I just might throw in a sample calculation using the wave function of the hydrogen atom, to give you an idea of what that's like.

## 9.2 Polarizing lenses experiment

Let's run through a simple tabletop experiment that illustrates the limitations of classical, deterministic reasoning. The outcome of the tiny conductive bands that interact with the electric component of incoming light particles. Light rays that hit the polarizing lens will either pass through or be reflected depending on their polarization. Light particles with a polarization perpendicular to the conductive bands pass through the lens, while light particles with polarization parallel to the conductive bands are reflected. This is because the surface of the lens has different conductive properties in the parallel and perpendicular directions.

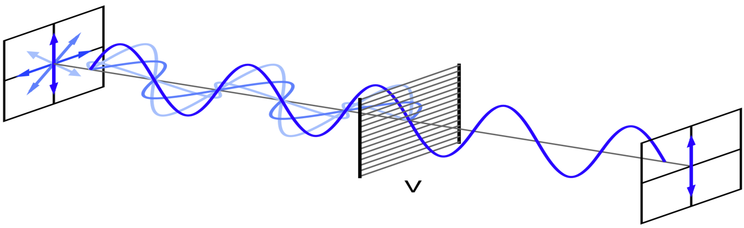



**Figure 9.2:** Incoming photons interact with the horizontal conductive bands of a polarizing lens. The horizontal bands of the lens reflect the horizontal component of the photons's electric field. Vertically-polarized photons pass through the lens because the conductive bands are perpendicular to their electric field. Thus, a vertically polarizing lens denoted $V$ allows only vertically polarized light to pass through.

Consider the illustration in Figure 9.3. The effect of a vertically polarizing lens on a beam of light is to only allow vertically polarized light to pass through.

$$\text{unpolarized light} \rightarrow \boxed{V} \rightarrow \text{vertically polarized light}$$

**Figure 9.3:** A vertically polarizing lens ($V$) allows only vertically polarized light particles to pass through.

In Figure 9.4 we see another aspect of polarizing lenses. If the light is already vertically polarized, adding a second vertically polarizing lens will not affect the beam. All light that passes through the first lens will also pass through the second.

Taking a vertically polarizing lens and rotating it by 90 degrees turns it into a horizontally polarizing lens. See Figure 9.5.
Note that horizontally polarizing lenses and vertically polarizing lenses are complementary: vertically polarized light will not pass

To understand any vector space, it is essential to construct a basis for the space. A natural choice for a basis is the standard basis, which we'll denote $\{|0\rangle, |1\rangle, |2\rangle, \ldots, |d-1\rangle\}$. The basis vectors are defined as:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \ldots, \quad |d-1\rangle = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}.$$

Note the indices are shifted by one so the first basis vector has index 0, not index 1. This zero-based indexing is chosen to make certain links between quantum theory and computer science more apparent.

One benefit of Dirac notation is that it doesn't require writing subscripts. To refer to a vector associated with properties $a$, $b$, and $c$, we can write $|a, b, c\rangle$, instead of the more convoluted expression $\vec{v}_{a,b,c}$.

We'll now focus solely on the two-dimensional complex vector space $\mathbb{C}^2$; however, the results and definitions presented below also apply to vectors of any dimension.

### Vectors

In Dirac notation, a vector in $\mathbb{C}^2$ is denoted as a *ket*:

$$|v\rangle = \alpha|0\rangle + \beta|1\rangle \qquad \Leftrightarrow \qquad \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

where $\alpha \in \mathbb{C}$ and $\beta \in \mathbb{C}$ are the *components* of $|v\rangle$ and $\{|0\rangle, |1\rangle\}$ is the standard basis for $\mathbb{C}^2$:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Why do we call the angle-bracket thing a "ket," you ask? Let me tell you about the *bra* part, and then it will start to make sense.

The Hermitian transpose of the ket-vector $|v\rangle = \alpha|0\rangle + \beta|1\rangle$ is the *bra*-vector $\langle v|$:

$$\langle v| = \overline{\alpha}\langle 0| + \overline{\beta}\langle 1| \qquad \Leftrightarrow \qquad [\overline{\alpha}, \overline{\beta}] = \overline{\alpha}[1, 0] + \overline{\beta}[0, 1].$$

Recall that the Hermitian transpose, also called the complex transpose, is the combination of the regular transpose ($\vec{v} \to \vec{v}^{\mathsf{T}}$) and the complex conjugation of each component of the vector ($v_i \to \overline{v_i}$), and is denoted as the dagger operator "$\dagger$" (see page 340 for a refresher on the dagger operator). Now observe how much simpler the *bra* input some data into a machine, and wait for the machine to process the data and output the answer.

We can think of both digital and quantum technology as black box processes, with internal workings that we can't access directly. In both cases, the intermediate representation of data is in a format that is unintelligible: we can't understand what is encoded in quantum states any more than we can understand what is encoded in digital data. For instance, an `mp3` file contains ones and zeros; but, unless we're unusually gifted, it's impossible to tell which artist plays the song 0101001010101010100111... just by looking at the raw, digital data. To understand information processing in the digital and quantum worlds, we must study the "adaptors"—the processes used to convert the internal data representation into signals we can intelligibly perceive.

To further highlight the parallel structure between digital information processing and quantum information processing, we'll now review the `mp3` compression task as an example of a digital information processing pipeline.

## Digital signal processing

A *sound card* is a computer component that converts between analog signals that we can hear and digital signals that computers understand. The sound card digitizes sound using an analog-to-digital converter (ADC). For music playback, the sound card uses a digital-to-analog converter (DAC), which transforms digital sounds into analog sound vibrations to be played through speakers. The ADC receives signal via the sound card's line-in and microphone jacks; the DAC outputs sound via the sound card's line-out and headphone jacks.

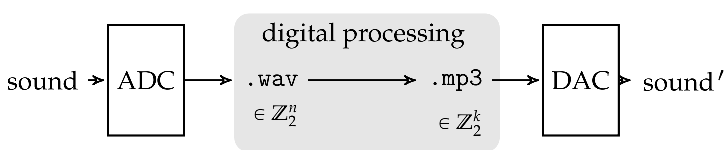



**Figure 9.9:** A digital information processing pipeline for sound recording and playback. Sound vibrations are captured by a microphone and converted to digital form using an analog-to-digital converter (ADC). Next the digital `wav` file is converted to the more compact `mp3` format using digital processing. In the last step, sound is converted back into analog sound vibrations by a digital-to-analog converter (DAC).

Figure 9.9 illustrates a full digital information processing pipeline for sound. We use an *analog-to-digital converter* (ADC) to transform of quantum effects like constructive interference.

The other big difference between classical and quantum computation is that the quantum computer outputs the state $|y\rangle$ only once. You can think of **quantum measurement as asking a question** about the state $|y\rangle$. You're free to perform any measurement, but **you can ask only one question**, since quantum measurements disrupt the state of a system.

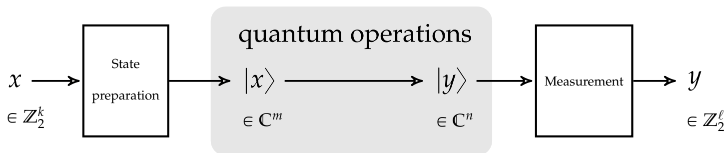



**Figure 9.10:** A quantum information processing pipeline. A classical bitstring $x$ of length $k$ is used as instructions for preparing an $m$-dimensional quantum state $|x\rangle$. Next, quantum operations are performed on the state $|x\rangle$ to convert it to the output state $|y\rangle$. Finally, the state $|y\rangle$ is *measured* to obtain the classical bitstring $y$ as output.

Quantum processing pipelines are analogous to digital information processing pipelines. The process of *state preparation* in quantum processing is analogous to the analog-to-digital conversion step in digital processing. In both cases we convert the input to the format required for the processing step. Similarly, quantum *measurements* correspond to the digital-to-analog conversion step. In both cases we must convert the output to a format we can understand.

In the next section, we'll discuss the components of the quantum information processing pipeline in more detail. We'll introduce the four *postulates* of quantum mechanics, which specify how quantum systems are represented and what we can do with them. The postulates of quantum mechanics roughly correspond to the conversion steps illustrated in Figure 9.10. One postulate defines how quantum states are prepared, another postulate describe the types of operations we can perform on quantum states, and a third postulate formally defines the process of quantum measurement. The next section is the "quantum mechanics explained in the space on the back of an envelope" part alluded to in the introduction of this chapter. We've set the scene, introduced Dirac notation, and now we can finally discuss the details of the quantum formalism.

## 9.5 Postulates of quantum mechanics

The *postulates* of quantum mechanics dictate the rules for working within the "quantum world." The four postulates define:

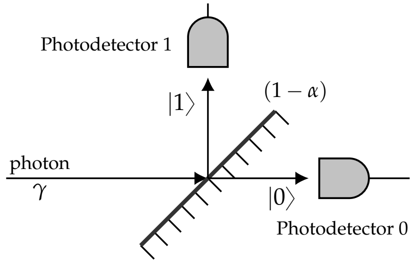



**Figure 9.13:** The state of a photon after encountering a $(1-\alpha)$-silvered mirror is $|\gamma\rangle = \alpha|0\rangle + \beta|1\rangle$. The probability that the horizontal photodetector "clicks" is $|\alpha|^2$, and is obtained by projecting $|\gamma\rangle$ on the subspace $|0\rangle\langle 0|$. The probability that the top photodetector clicks is equal to $|\beta|^2$, and is obtained by projecting $|\gamma\rangle$ on the subspace $|1\rangle\langle 1|$.

## Composite quantum systems

So far we discussed state preparation, quantum operations, and quantum measurements of individual qubits. There's just enough room on the back of our envelope to discuss quantum models for systems made of multiple qubits.

Classically, if we have two bits $b_1 \in \{0,1\}$ and $b_2 \in \{0,1\}$, we can concatenate them to obtain a bit string $b_1b_2 \in \{0,1\}^2$, which can have one of four possible values: 00, 01, 10, and 11. The combined state of two qubits $|\varphi_1\rangle \in \mathbb{C}^2$ and $|\varphi_2\rangle \in \mathbb{C}^2$ is the *tensor product state* $|\varphi_1\rangle \otimes |\varphi_2\rangle$ in the four-dimensional *tensor product space* $\mathbb{C}^2 \otimes \mathbb{C}^2 = \mathbb{C}^4$. A basis for the tensor product space can be obtained by taking all possible combinations of the basis vectors for the individual qubits: $\{|0\rangle \otimes |0\rangle, |0\rangle \otimes |1\rangle, |1\rangle \otimes |0\rangle, |1\rangle \otimes |1\rangle\}$.

**Postulate 4.** The state space of a composite quantum system is equal to the tensor product of the state spaces of the individual systems. If systems $1, 2, \ldots, n$ exist in states $|\varphi_1\rangle, |\varphi_2\rangle, \cdots, |\varphi_n\rangle$, then the state of the composite system is $|\varphi_1\rangle \otimes |\varphi_2\rangle \otimes \cdots \otimes |\varphi_n\rangle$.

Postulate 4 tells us how we can combine the state spaces of different quantum systems to describe a composite system. Many interesting quantum applications involve operations on multiple qubits and are described by vectors in a tensor product space, so let's look more closely at this "$\otimes$"-thing.

**Tensor product space** If you'd never heard of *tensor products* before, don't worry—the only scary part is the tensor product symbol "$\otimes$," which we'll explain in this section. A *tensor product space* consists of all possible combinations of the basis vectors for the two subspaces.

she performs, and her measurement disturbs the state $|\psi\rangle$ whenever she picks a basis different from the one used by Alice. Since it's not possible to measure quantum systems without disturbing them, the eavesdropper Eve reveals her presence by introducing errors in the transmitted data. Some of the *check bits* Alice and Bob compare in Step 3 will disagree, and thus Alice and Bob will know that someone is eavesdropping on them. Though quantum mechanics does not prevent eavesdropping, it does give Alice and Bob the ability to detect when an eavesdropper is present.

The BB84 protocol established the beginning of a new field at the intersection of computer science and physics that studies *quantum key distribution* protocols. The field has developed rapidly from theory to research, and today there are even commercial quantum cryptography systems. It's interesting to compare quantum cryptography with the public key cryptography systems discussed in Section 7.9. The security of the RSA public-key encryption is based on the computational difficulty of factoring large numbers. The security of quantum cryptography is guaranteed by the laws of quantum mechanics.

[ Bennett–Brassard quantum cryptography protocol from 1984 ]
`https://en.wikipedia.org/wiki/BB84`

[ Using quantum phenomena to distribute secret keys ]
`https://en.wikipedia.org/wiki/Quantum_key_distribution`

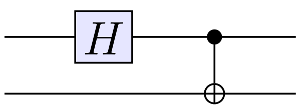
## Quantum computing

The idea of quantum computing has existed since the early days of quantum mechanics. Richard Feynman originally proposed the idea of a *quantum simulator* in 1982, which is a quantum apparatus that can simulate the quantum behaviour of another physical system. Imagine a device that can simulate the behaviour of physical systems that would otherwise be too difficult and expensive to build. The quantum simulator would be much better at simulating quantum phenomena than any simulation of quantum mechanics on a classical computer.

Another possible application of a quantum simulator could be to encode classical mathematical optimization problems as constraints in a quantum system, then let the quantum evolution of the system "search" for good solutions. Using a quantum simulator in this way, it might be possible to find solutions to optimization problems much faster than any classical optimization algorithm could.

Once computer scientists started thinking about quantum computing, they weren't satisfied with studying optimization problems alone, and they set out to qualify and quantify all the computational tasks that are possible with qubits. A quantum computer stores and manipulates information that is encoded as quantum states. It's possible to perform certain computational tasks on a quantum computer much faster than on any classical computer. We'll discuss *Grover's search algorithm* and *Shor's factoring algorithm* below, but first let's introduce the basic notions of quantum computing.

**Quantum circuits** Computer scientists like to think of quantum computing tasks as series of "quantum gates," in analogy with the logic gates used to construct classical computers. Figure 9.20 shows an example of a quantum circuit that takes two qubits as inputs and produces two qubits as outputs.




$$\Leftrightarrow \quad \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 & 0 & -\frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix}$$

**Figure 9.20:** A quantum circuit that applies the Hadamard gate to the first qubit, then applies the controlled-NOT gate from the first qubit to the second qubit.

This circuit in Figure 9.20 is the combination of two quantum gates. The first operation is to apply the Hadamard gate $H$ on the first qubit, leaving the second qubit untouched. This operation is equivalent to multiplying the input state by the matrix $H \otimes \mathbb{1}$. The second operation is called the *controlled-NOT* (or controlled-$X$) gate, which applies the $X$ operator (also known as the NOT gate) to the second qubit whenever the first qubit is $|1\rangle$, and does nothing otherwise:

$$\text{CNOT}(|0\rangle \otimes |\varphi\rangle) = |0\rangle \otimes |\varphi\rangle, \qquad \text{CNOT}(|1\rangle \otimes |\varphi\rangle) = |1\rangle \otimes X|\varphi\rangle.$$

The circuit illustrated in Figure 9.20 can be used to create entangled quantum states. If we input the quantum state $|00\rangle = |0\rangle \otimes |0\rangle$ into the circuit, we obtain the maximally entangled state $|\Phi_+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ as output, as depicted in Figure 9.21.

Quantum circuits can also represent quantum measurements. Figure 9.22 shows how a quantum measurement in the standard basis is represented.

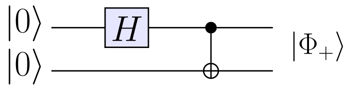



**Figure 9.21:** Inputting $|0\rangle\otimes|0\rangle$ into the circuit produces an EPR state $|\Phi_+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ on the two output wires of the circuit.

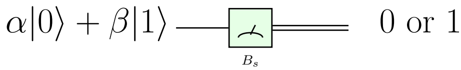



**Figure 9.22:** Measurement in the standard basis $B_s = \{|0\rangle, |1\rangle\}$. The projectors of this measurement are $\Pi_0 = |0\rangle\langle 0|$ and $\Pi_1 = |1\rangle\langle 1|$.

We use double lines to represent the flow of classical information in the circuit.

**Quantum registers** Consider a quantum computer with a single register $|R\rangle$ that consists of three qubits. The quantum state of this quantum register is a vector in $\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2$:

$$|R\rangle = \; (\alpha_1|0\rangle + \beta_1|1\rangle) \otimes (\alpha_2|0\rangle + \beta_2|1\rangle) \otimes (\alpha_3|0\rangle + \beta_3|1\rangle),$$

where the tensor product $\otimes$ is used to combine the quantum states of the individual qubits. We'll call this the "physical representation" of the register and use 0-based indexing for the qubits. Borrowing language from classical computing, we'll call the rightmost qubit the *least significant* qubit, and the leftmost qubit the *most significant* qubit.

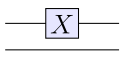
The tensor product of three vectors with dimension two is a vector with dimension eight. The quantum register $|R\rangle$ is thus a vector in an eight-dimensional vector space. The quantum state of a three-qubit register can be written as:

$$|R\rangle = a_0|0\rangle + a_1|1\rangle + a_2|2\rangle + a_3|3\rangle + a_4|4\rangle + a_5|5\rangle + a_6|6\rangle + a_7|7\rangle,$$

where $a_i$ are complex components. We'll call this eight-dimensional vector space the "logical representation" of the quantum register. Part of the excitement about quantum computing is the huge size of the "logical space" where quantum computations take place. The logical space of a 10-qubit quantum register has dimension $2^{10} = 1024$. That's 1024 complex components we're talking about. That's a big state space for just a 10-qubit quantum register. Compare this with a 10-bit classical register, which can store one of $2^{10} = 1024$ discrete values.

We won't discuss quantum computing further here, but I still want to show you some examples of single-qubit quantum operations and their effect on the tensor product space, so you'll have an idea of the craziness that is possible.

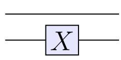
**Quantum gates** Let's say you've managed to construct a quantum register; what can you do with it? Recall the single-qubit quantum operations $Z$, $X$, and $H$ we described earlier. We can apply any of these operations on individual qubits in the quantum register. For example, applying the $X = \left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$ gate to the first (most significant) qubit of the quantum register corresponds to the following quantum operation:



$$\Leftrightarrow \qquad X \otimes \mathbb{1} \otimes \mathbb{1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The operator $X\otimes\mathbb{1}\otimes\mathbb{1}$ "toggles" the first qubit in the register while leaving all other qubits unchanged.

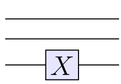
Yes, I know the tensor product operation is a bit crazy, but that's the representation of composite quantum systems and operations so please get used to it. What if we apply the $X$ operator to the middle qubit?



$$\Leftrightarrow \qquad \mathbb{1} \otimes X \otimes \mathbb{1} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

Compare the structure of the operators $X\otimes\mathbb{1}\otimes\mathbb{1}$ and $\mathbb{1}\otimes X\otimes\mathbb{1}$. See how the action of $X$s affects different parts of the tensor product space $\mathbb{C}^8$?

To complete the picture, let's also see the effects of applying the $X$ gate to the third (least significant) qubit in the register:



$$\Leftrightarrow \qquad \mathbb{1} \otimes \mathbb{1} \otimes X = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Crazy stuff, right? Don't worry, in time you'll get used to the space-within-a-space structure concept.

**Okay, so what?**

Quantum computers give us access to a very large state space. The fundamental promise of quantum computing is that a small set of simple quantum operations (quantum gates) can be used to perform interesting computational tasks. Sure it's difficult to interact with and manipulate quantum systems, but the space is so damn big that it's worth checking out what kind of computing you can do in there. It turns out there are already several useful things you can do using a quantum computer. The two flagship applications for quantum computing are Grover's search algorithm and Shor's factoring algorithm.

**Grover's search algorithm** Suppose you're given an unsorted list of $n$ items and you want to find a particular item in that list. This is called an *unstructured search problem*. This is a hard problem to solve for a classical computer since the algorithm must parse through the entire list, which takes roughly $n$ steps. In contrast, the unstructured problem can be solved in roughly $\sqrt{n}$ steps on a quantum computer using *Grover's search algorithm*.

The quantum speed for the unstructured search problem sure is nice, but it's really nothing to get excited about. The real money-maker for the field of quantum computing has been Shor's factoring algorithm for factoring products of prime numbers.

**Shor's factoring algorithm** The security of the RSA cryptosystem we discussed in Section 7.9 is based on the assumption that factoring products of large prime numbers is computationally intractable. Given the product $de$ of two unknown prime numbers $d$ and $e$, it is computationally difficult to find the factors $e$ and $d$. No classical algorithm is known that can factor large numbers; even the letter agencies will have a hard time finding the factors of $de$ when $d$ and $e$ are chosen to be sufficiently large prime numbers. Thus, if an algorithm that could quickly factor large numbers existed, attackers would be able to break many of the current security systems. *Shor's factoring algorithm* fits the bill, theoretically speaking.

Shor's algorithm reduces the factoring problem to the problem of *period finding*, which can be solved efficiently using the quantum Fourier transform. Shor's algorithm can factor large numbers efficiently (in polynomial time). This means RSA encryption would be easily hackable by running Shor's algorithm on a sufficiently large, and sufficiently reliable quantum computer. The letter agencies are excited about this development since they'd love to be able to hack all present-day cryptography. Can you imagine not being able to log in securely to any website because Eve is listening in, hacking your crypto using her quantum computer?

Currently, Shor's algorithm is only a *theoretical* concern. Despite considerable effort, no quantum computers exist today that can manipulate quantum registers with thousands of qubits.

## Discussion

Quantum computing certainly presents interesting possibilities, but it's a little early to imagine a quantum computing revolution in tomorrow's newspaper. As with startup ventures, it's the implementation that counts—not the idea. The current status of quantum computing as a technology is mixed. On one hand, certain quantum algorithms performed in logical space are very powerful; on the other hand, the difficulty of building a quantum computer is not to be underestimated.

It's also important to keep in mind that quantum computers are not better at solving arbitrary computational problems than the computers we already use. The problems that may benefit from a quantum speedup have a particular structure, which can be tackled with a choreographed pattern of constructive and destructive interference in quantum registers. Yet not all computationally hard problems have this structure. Quantum computing technology is at a crossroad: it could become a revolutionary development, or it could turn out that building a large-scale quantum computer is not worth the engineering challenge. So although it's cool we can execute certain tasks faster on a quantum computer, don't throw out your classical computer just yet.

Even if the quest to build a quantum computer doesn't pan out, we're certain to learn many interesting things about fundamental physics along the way. Besides, learning about the fundamental nature of quantum information is more scientifically valuable than trying to hack people's email. In the next section, we'll give an example of a new communication task that was discovered through the study of quantum information science.

**Quantum teleportation** Figure 9.23 illustrates a surprising aspect of quantum information: we can "teleport" a quantum state $|\psi\rangle$ from one lab to another. The quantum state $|\psi\rangle$ starts in the first qubit of the register, which is held by Alice, and ends in the third qubit, which is in Bob's lab, but there is no quantum communication channel between the two labs. This is why the term "quantum teleportation" was coined to describe this communication task, since the state $|\psi\rangle$

can realize the potential of every child born anywhere in the world. Right now it seems like it won't be an easy change, but this is how it seemed when people were trying to figure out quantum mechanics, too. All it takes is a critical mass of people who realize and truly internalize the fact we're all on the same team, and all the divisions we see between us are pure bullshit. If we make sure that two generations of kids grow up without economic strife or bullets flying by their heads, then—I guarantee you—they will be able to figure out the rest. Together, an educated citizenry armed with the knowledge of math, history, science, and technology is more powerful than a dozen Systems combined. All the System has is an outdated hierarchical power structure and capital to pay people and make them do as they're told. We've got six billion people, the internet, and printing presses on our side. Who do you think will win?

## 9.9 Quantum mechanics problems

Let's recap what just happened here. Did we really cover all the topics of an introductory quantum mechanics course? Yes, we did! Thanks to your solid knowledge of linear algebra, learning the postulates of quantum mechanics took only a few dozen pages. Sure we went quickly and skipped the more physics-y topics, but we covered all the core ideas of quantum theory.

But surely it's impossible to learn quantum mechanics in such a short time? Well, you tell me. You're here. The problems are here. Prove to me you've really learned quantum mechanics by tackling the practice problems presented in this section like a boss. It's the end of the book, so don't be saving your energy. Solve these problems and then you're done.

**P9.1** You work in a quantum computing startup and your boss asks you to implement the quantum gate $Q = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right]$. Can you do it?

Hint: Recall the requirements for quantum gates.

**P9.2** The $Y$ gate is defined as $Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$. Compute the effect of the operator $YY$ on the vectors of the standard basis $\{|0\rangle, |1\rangle\}$.

**P9.3** Compute $XHHY(\alpha|0\rangle + \beta|1\rangle)$.

Hint: Use the Hadamard gate's properties to simplify the calculation.

**P9.4** Specifying an arbitrary vector $\alpha|0\rangle + \beta|1\rangle \in \mathbb{C}^2$ requires four parameters: the real and imaginary parts of $\alpha$ and $\beta$. Thus one might think that qubits have four *degrees of freedom*. However, the unit-length requirement and the fact that we can ignore the global phase of a qubit correspond to

# End matter

## Conclusion

By tackling the linear algebra concepts in this book, you've proven you can handle computational complexity, develop geometric intuition, and understand abstract math ideas. These are precisely the types of skills you'll need in order to understand more advanced math concepts, build scientific models, and develop useful applications. Congratulations on taking this important step toward your mathematical development. Throughout this book, we learned about vectors, linear transformations, matrices, abstract vector spaces, and many other math concepts that are useful for building math models.

Mathematical models serve as a highly useful common core for all sciences, and the techniques of linear algebra are some of the most versatile modelling tools that exist. Every time you use an equation to characterize a real-world phenomenon, you're using your math modelling skills. Whether you're applying some well-known scientific model to describe a phenomenon or developing a new model specifically tailored to a particular application, the deeper your math knowledge, the better the math models you'll be able to leverage. Let's review and catalogue some of the math modelling tools we've learned about, and see how linear algebra fits into a wider context.

To learn math modelling, you must first understand basic math concepts such as numbers, equations, and functions $f : \mathbb{R} \to \mathbb{R}$. Once you know about functions, you can use different formulas $f(x)$ to represent, model, and predict the values of real-world quantities. Working with functions is the first modelling superpower conferred on people who become knowledgeable in math. For example, understanding the properties of the function $f(x) = Ae^{-x/B}$ in the abstract enables you to describe the expected number of atoms remaining in a radioactive reaction $N(t) = N_0\, e^{-\gamma t}$, predict the voltage of a discharging capacitor over time $v(t) = V_0\, e^{-\frac{t}{RC}}$, and understand the exponential probability distribution $p_X(x) = \lambda e^{-\lambda x}$.

To further develop your math modelling skills, the next step is to generalize the concepts of inputs $x$, outputs $y$, and functions $f$ to other input-output relationships. In linear algebra, we studied functions of the form $T : \mathbb{R}^n \to \mathbb{R}^m$ that obey the linear property:

$$T(\alpha\vec{x}_1 + \beta\vec{x}_2) = \alpha T(\vec{x}_1) + \beta T(\vec{x}_2).$$

This linear structure enables us to study the properties of many functions, solve equations involving linear transformations, and build useful models for many applications (some of which we discussed in Chapter 7). The mathematical structure of a linear transformation $T : \mathbb{R}^n \to \mathbb{R}^m$ can be *represented* as multiplication by a matrix $M_T \in \mathbb{R}^{m\times n}$. The notion of matrix representations ($T \Leftrightarrow M_T$) was central throughout this book. Even if you forget the computational procedures we learned, the idea of representations should stick with you, and you should be able to recognize representations in many contexts. That's a big deal, because most advanced math topics involve studying the parallels between different abstract notions. Understanding linear transformations and their representations is an important first step toward advanced math topics.

The computational skills you learned in Chapter 3 are also useful; though you probably won't be solving any problems by hand using row operations from this point forward, since computers outclass humans on matrix arithmetic tasks. Good riddance. Until now, you did all the work and used `SymPy` to check your answers. From now on, you can let `SymPy` do all the calculations and your job will be to chill.

If you didn't skip the sections on abstract vector spaces, you know about the parallels between the vector space $\mathbb{R}^4$ and the abstract vector spaces of third-degree polynomials $a_0 + a_1x + a_2x^2 + a_3x^3$ and $2 \times 2$ matrices $\left[\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right]$. This is another step up the ladder of abstraction, as it deepens your understanding of all math objects with vector-like structure.

It was my great pleasure to be your guide through the subject of linear algebra. I hope you walk away from this book with a solid understanding of how the concepts of linear algebra fit together. In the book's introduction, I likened linear algebra to playing with LEGOs. Indeed, if you feel comfortable manipulating vectors and matrices, performing change-of-basis operations, and using the matrix decomposition techniques to see inside matrices, you'll be able to "play" with all kinds of complex systems and problems. For example, consider the linear transformation $T$ that you want to apply to an input vector $\vec{v}$. Suppose the linear transformation $T$ is most easily described in the basis $B'$, but the vector $\vec{v}$ is expressed with respect to the basis $B$. "No problem," you can say, and proceed to build the following chain of matrices that compute the output vector $\vec{w}$:

$$[\vec{w}]_B = {}_B[\mathbb{1}]_{B'}\ {}_{B'}[A_T]_{B'}\ {}_{B'}[\mathbb{1}]_B[\vec{v}]_B.$$

Do you see how matrices and vectors fit together neatly like LEGOs?

I can't tell you what the next step on your journey will be. With your new linear algebra modelling skills, a thousand doors have opened for you; now you must explore and choose. Will you learn how to code and start a software company? Maybe you'll use your analytical skills to go to Wall Street and destroy the System from the inside. Or perhaps you'll apply your modelling skills to revolutionize energy generation, thus making human progress sustainable. Regardless of your choice of career, I hope you'll stay on good terms with math and continue learning whenever you have the chance. Good luck with your studies!

# Social stuff

Be sure to contact me if you have any feedback about this book. It helps to hear which parts of the book readers like, hate, or don't understand. I consider all feedback in updating and improving future editions of this book. This is how the book got good in the first place—lots of useful feedback from readers. You can reach me by email at `ivan@minireference.com`.

Another thing you can do to help us is to review the book on Amazon.com, Goodreads, Google Books, or otherwise spread the word about the **No Bullshit Guide** textbook series. Talk to your friends and let them in on the knowledge buzz.

If you want to know what Minireference Co. has been up to, check out our blog at `minireference.com/blog/`. The blog is a mix of 30% publishing technology talk, 50% startup business talk, and 20% announcements. Checking the blog is the easiest way to follow the progress of our revolution in the textbook industry. For real; we're totally serious about making education accessible through affordable textbooks. You can also connect via Twitter `@minireference` and Facebook `fb.me/noBSguide`.

# Acknowledgements

This book would not have been possible without the support of my parents, and my teachers: Claude Crépeau, Patrick Hayden, Paul Kenton, Benoit Larose, and Ioannis Psaromiligkos. It's by listening to them that I gained the confidence to learn and later teach advanced math topics. Patrick Hayden deserves a particular mention in this regard since his ability to explain concepts clearly inspired me to tackle complicated problems using a steady, methodical approach. Patrick also showed me that it's possible to trick students into learning even the most advanced topics by presenting the material in a logical and structured manner. Many thanks also to David Avis, Arlo Breault, Michael Hilke, Igor Khavkine, Felix Kwok, Juan Pablo Di Lelle, Ivo Panayotov, and Mark M. Wilde for their support with the book.

The errorlessness and consistency of the text would not have been possible without the help of my editor Sandy Gordon, who did a great job at polishing the text until it flowed. Truly no bullshit is allowed into the book when Sandy's on watch. Many thanks to Polina Anis'kina who helped me to create the problem sets for the book.

## General linear algebra links

Below are some useful links to resources where you can learn more about linear algebra. We covered a lot of ground, but linear algebra is endless. Don't sit on your laurels and think you're the boss now that you've completed this book and its problem sets. You have the tools, but you need to practice using them. Try reading about the same topics from some other sources. See if you can do the problem sets in another linear algebra textbook. Try to use linear algebra in the coming year and further solidify your understanding of the material.

[ Video lectures of Gilbert Strang's linear algebra class at MIT ]
`http://ocw.mit.edu/courses/mathematics/18-06-linear-algebra-spring-2010`

[ The essence of linear algebra video playlist by 3Blue1Brown ]
`http://bit.ly/essence_of_LA`

[ A free online textbook with amazing interactive visualizations ]
`http://immersivemath.com/ila/index.html`

[ Lecture notes by Terrence Tao ]
`http://www.math.ucla.edu/~tao/resource/general/115a.3.02f/`

[ Wikipedia overview on matrices ]
`https://en.wikipedia.org/wiki/Matrix_(mathematics)`

[ Linear algebra wikibook (with solved problems) ]
`https://en.wikibooks.org/wiki/Linear_Algebra`

[ Proofs involving linear algebra ]
`http://proofwiki.org/wiki/Category:Linear_Algebra`

[ Linear algebra from first principles using diagrams only ]
`https://graphicallinearalgebra.net/`

# Appendix A

# Answers and solutions

## Chapter 1 solutions

### Answers to exercises

**E1.1 a)** $x = 3$; **b)** $x = 30$; **c)** $x = 2$; **d)** $x = -3$. **E1.2 a)** $\mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$; **b)** $\mathbb{C}$; **c)** $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$; **d)** $\mathbb{Q}, \mathbb{R}, \mathbb{C}$; **e)** $\mathbb{R}, \mathbb{C}$. **E1.3 a)** 21; **b)** 0; **c)** $\frac{2}{27}$. **E1.4 a)** $x = 2$; **b)** $x = 25$; **c)** $x = 100$. **E1.5 a)** $f^{-1}(x) = x^2$, $x = 16$. **b)** $g^{-1}(x) = -\frac{1}{2}\ln(x)$, $x = 0$. **E1.6 a)** $(x-1)(x-7)$; **b)** $(x+2)^2$; **c)** $(x+3)(x-3)$. **E1.7 a)** $x^2 + 2x - 15 = (x+1)^2 - 16 = 0$, which has solutions $x = 3$ and $x = -5$; **b)** $x^2 + 4x + 1 = (x+2)^2 - 3 = 0$, with solutions $x = -2 + \sqrt{3}$ and $x = -2 - \sqrt{3}$. **E1.8** $x_1 = \frac{3}{2}$ and $x_2 = -1$. **E1.9** $x = \pm\sqrt{2}$. **E1.10** Domain: $\mathbb{R}$. Image: $[-2, 2]$. Roots: $\{\ldots, -\frac{\pi}{2}, \frac{\pi}{2}, \frac{3\pi}{2}, \frac{5\pi}{2}, \ldots\}$. **E1.11 a)** $p(x)$ is even and has degree 4. **b)** $q(x)$ is odd and has degree 7. **E1.12 a)** $x = 5$ and $x = -3$; **b)** $x = 1 + \sqrt{3}$ and $x = 1 - \sqrt{3}$. **E1.13** $x = \sqrt{21}$. **E1.14** $V = 33.51$ and $A = 50.26$. **E1.15** Length of track $= 5C = 5\pi d = 11.47$ m. **E1.16** $x = 5\cos(45°) = 3.54$, $y = 5\sin(45°) = 3.54$; $C = 10\pi$. **E1.17 a)** $\frac{\pi}{6}$ rad; **b)** $\frac{\pi}{4}$ rad; **c)** $\frac{\pi}{3}$ rad; **d)** $\frac{3\pi}{2}$ rad. **E1.18 a)** $-1$; **b)** 1; **c)** 0. **E1.19 a)** 0; **b)** 1; **c)** $\frac{1}{2}$; **d)** 1. **E1.20 a)** $(4, 0)$. **b)** $(-2, -3)$. **c)** $(7, 3)$. **E1.21 a)** $\vec{v}_1 = (5\sqrt{3}, 5) = (8.66, 5)$. **b)** $\vec{v}_2 = (0, -12)$. **c)** $\vec{v}_3 = (-2.95, 0.52)$. **E1.22 a)** $\vec{u}_1 = 4\angle 0°$. **b)** $\vec{u}_2 = \sqrt{2}\angle 45°$. **c)** $\vec{u}_3 = \sqrt{10}\angle 108.43°$. **E1.23 a)** $(1, \frac{1}{2})$. **b)** $(1, 2)$. **c)** $(-2, 2)$. **E1.24** $x = 2$, $y = 3$. **E1.25** $x = 5$, $y = 6$, and $z = -3$. **E1.26** $p = 7$ and $q = 3$. **E1.27 a)** $\{2, 4, 6, 7\}$; **b)** $\{1, 2, 3, 4, 5, 6\}$; **c)** $\{1, 3, 5\}$; **d)** $\emptyset$; **e)** $\{1, 2, 3, 4, 5, 6, 7\}$; **f)** $\{7\}$; **g)** $\{2, 4, 6, 7\}$; **h)** $\emptyset$. **E1.28 a)** $(-\infty, \frac{3}{2})$; **b)** $(-\infty, -5]$; **c)** $(-1, 4)$; **d)** $(4, \infty)$; **e)** $[\frac{14}{3}, \infty)$; **f)** $(-\infty, -4] \cup [2, \infty)$.

### Solutions to selected exercises

**E1.12 a)** Rewrite the equation putting all terms on the right-hand side: $0 = x^2 - 2x - 15$. We can factor this quadratic by inspection. Are there numbers $a$ and $b$ such that $a + b = -2$ and $ab = -15$? Yes, $a = -5$ and $b = 3$, so $0 = (x-5)(x+3)$. **b)** Rewrite the equation so all terms are on the left-hand side: $3x^2 - 6x - 6 = 0$. Nice, the cubic terms cancel! We'll use the quadratic formula to solve this equation $x = \frac{6 \pm \sqrt{(-6)^2 - 4(3)(-6)}}{6} = \frac{6 \pm 6\sqrt{3}}{6} = 1 \pm \sqrt{3}$.

**E1.13** The cosine rule tells us $x^2 = 4^2 + 5^2 - 2(4)(5)\cos(60°) = 21$. Therefore $x = \sqrt{21}$.

**P1.34** The first tank contains $\frac{1}{4} \times 4000 = 1000\,\text{L}$. The second tank contains three times more water, so 3000 L. The total is 4000 L.

**P1.35** The amount of wood in a pack of wood is proportional to the area of a circle $A = \pi r^2$. The circumference of this circle is equal to the length of the rope $C = \ell$. Note the circumference is proportional to the radius $C = 2\pi r$. If we want double the area, we need the circle to have radius $\sqrt{2}r$, which means the circumference needs to be $\sqrt{2}$ times larger. If we want a pack with double the wood, we need to use a rope of length $\sqrt{2}\ell$.

**P1.36** In 10 L of a 60% acid solution there are 6 L of acid and 4 L of water. A 20% acid solution will contain four times as much water as it contains acid, so 6 L acid and 24 L water. Since the 10 L we start from already contains 4 L of water, we must add 20 L.

**P1.37** The document must have a 768/1004 aspect ratio, so its height must be $6 \times \frac{1004}{768} = 7.84375$ inches.

**P1.38** If we rewrite $1+2+3+\cdots+98+99+100$ by pairing numbers, we obtain the sum $(1+100)+(2+99)+(3+98)+\cdots$. This list has 50 terms and each term has the value 101. Therefore $1+2+3+\cdots+100 = 50 \times 101 = 5050$.

**P1.49** There exists at least one banker who is not a crook. Another way of saying the same thing is "not all bankers are crooks"—just *most* of them.

**P1.50** Everyone steering the ship at Monsanto ought to burn in hell, forever.

**P1.51** **a)** Investors with money but without connections. **b)** Investors with connections but no money. **c)** Investors with both money and connections.

# Chapter 2 solutions

## Answers to exercises

**E2.1** $A^{-1} = \begin{bmatrix} \frac{1}{7} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}$. **E2.2** **a)** $A\vec{v} = \left[\begin{smallmatrix} 7 \\ 14 \end{smallmatrix}\right]$; **b)** $B\vec{v} = \left[\begin{smallmatrix} -1 \\ 9 \end{smallmatrix}\right]$; **c)** $A(B\vec{v}) = \left[\begin{smallmatrix} 26 \\ 41 \end{smallmatrix}\right]$; **d)** $B(A\vec{v}) = \left[\begin{smallmatrix} -7 \\ 63 \end{smallmatrix}\right]$; **e)** $A\vec{w} = \left[\begin{smallmatrix} -15 \\ -32 \end{smallmatrix}\right]$; **f)** $B\vec{w} = \left[\begin{smallmatrix} 3 \\ -21 \end{smallmatrix}\right]$. **E2.3** $v_1 = -2$, $v_2 = 3$. **E2.4** **a)** $(1,1,3)$; **b)** $(1,1,-3)$; **c)** $(3,3,3)$; **d)** $\sqrt{2}$. **E2.5** **a)** 5; **b)** $(-1,-1,1)$; **c)** $(1,1,-1)$; **d)** $(0,0,0)$. **E2.6** **a)** This part has been omitted for brevity. **b)** $\vec{v}_1 = (9.848, 1.736)$; $\vec{v}_2 = (8.66, 5)$; $\vec{v}_3 = (5, 8.66)$; $\vec{v}_4 = (-5, 8.66)$. **c)** $\Pi_{\hat{\imath}}(\vec{v}_1) = 9.848$; $\Pi_{\hat{\imath}}(\vec{v}_2) = 8.66$; $\Pi_{\hat{\imath}}(\vec{v}_3) = 5$; $\Pi_{\hat{\imath}}(\vec{v}_4) = -5$. **d)** $\Pi_{\hat{\jmath}}(\vec{v}_1) = 1.736$; $\Pi_{\hat{\jmath}}(\vec{v}_2) = 5$; $\Pi_{\hat{\jmath}}(\vec{v}_3) = 8.66$; $\Pi_{\hat{\jmath}}(\vec{v}_4) = 8.66$. **e)** $\Pi_{\vec{d}}(\vec{v}_1) = (5.79, 5.79)$ and $\|\Pi_{\vec{d}}(\vec{v}_1)\| = 8.19$; $\Pi_{\vec{d}}(\vec{v}_2) = (6.83, 6.83)$ and $\|\Pi_{\vec{d}}(\vec{v}_2)\| = 9.66$; $\Pi_{\vec{d}}(\vec{v}_3) = (6.83, 6.83)$ and $\|\Pi_{\vec{d}}(\vec{v}_3)\| = 9.66$; $\Pi_{\vec{d}}(\vec{v}_4) = (1.83, 1.83)$ and $\|\Pi_{\vec{d}}(\vec{v}_4)\| = 2.59$. **E2.7** **a)** $\left[\begin{smallmatrix} 3 & 2 \\ 4 & 1 \end{smallmatrix}\right]$; **b)** $\begin{bmatrix} -2 & 2 \\ 3 & -2 \\ 0 & 1 \end{bmatrix}$; **c)** $\left[\begin{smallmatrix} 17 & 16 \\ 8 & 9 \end{smallmatrix}\right]$; **d)** $\left[\begin{smallmatrix} 13 & 12 & 11 & 10 \\ 2 & 3 & 4 & 5 \end{smallmatrix}\right]$; **e)** $\left[\begin{smallmatrix} 2 & 1 & 4 \\ -2 & 4 & 1 \end{smallmatrix}\right]$; **f)** Doesn't exist; **g)** $\begin{bmatrix} -2 & -6 \\ 5 & 10 \\ 2 & 1 \end{bmatrix}$; **h)** $-5$; **i)** Doesn't exist; **j)** Doesn't exist; **k)** $-5$; **l)** 1; **m)** 4; **n)** 4. **E2.8** **a)** 14; **b)** 5; **c)** 0; **d)** $\left[\begin{smallmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{smallmatrix}\right]$; **e)** $\begin{bmatrix} 4 & -2 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$; **f)** $\begin{bmatrix} 2 & -1 & 0 \\ 4 & -2 & 0 \\ 6 & -3 & 0 \end{bmatrix}$. **E2.9** $\alpha = -\frac{1}{2}$ and $\beta = 12$. **E2.10** **a)** Yes; **b)** No; **c)** Yes.

## Solutions to selected exercises

**E2.1** To find $A^{-1}$ we must consider the action of $A = \left[\begin{smallmatrix} 7 & 0 \\ 0 & 2 \end{smallmatrix}\right]$ on an arbitrary vector $\vec{v} = \left[\begin{smallmatrix} v_1 \\ v_2 \end{smallmatrix}\right]$, and perform the inverse action. Since $A$ multiplies the first component by 7, $A^{-1}$ must divide the first component by 7. Since $A$ multiplies the second component by 2, $A^{-1}$ must divide the second component by 2. Thus $A^{-1} = \begin{bmatrix} \frac{1}{7} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}$.

**E2.10** An expression is linear in the variable $v$ if it contains $v$ raised only to the first power. This is the case for the first and third expressions but not the second, since it contains $\sqrt{x} = x^{\frac{1}{2}}$.

## Answers to problems

**P2.1** **a)** $q(x)$ is nonlinear; **b)** $f(x)$, $g(x)$, and $h(x)$ are all linear; **c)** $i(x)$ is nonlinear; **d)** $j(x)$ is nonlinear. **P2.2** $(1,2,3)$. **P2.3** $|a\rangle + |b\rangle = 5|0\rangle + 2|1\rangle$. **P2.4** **a)** 0; **b)** $(0,0,1)$; **c)** $(0,1,-1)$; **d)** $(0,0,-1)$. **P2.5** **a)** 5; **b)** $(-1,1,1)$; **c)** $(0,0,0)$; **d)** $(0,0,0)$. **P2.6** **a)** 6. **b)** 0. **c)** $-3$. **d)** $(-2,1,1)$. **e)** $(3,-3,0)$. **f)** $(7,-5,1)$. **P2.7** **a)** $(2,3,3,7,8)$. **b)** $(0,-1,-3,-1,-2)$. **c)** 30. **P2.8** $(-\frac{2}{3},\frac{1}{3},\frac{2}{3})$ or $(\frac{2}{3},-\frac{1}{3},-\frac{2}{3})$. **P2.9** $(12,-4,-12)$. **P2.10** The tractor's trajectory is a half-circle. The total distance travelled is 3.14 km. **P2.12** $M\vec{v} = \begin{bmatrix} \alpha z_1+\beta z_2 \\ \gamma z_1+\delta z_2 \end{bmatrix}$. **P2.13** $A = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$; $B = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$; $C = \begin{bmatrix} 3 & 1 \\ 0 & 1 \end{bmatrix}$. $AB = \begin{bmatrix} 3 & 3 \\ 0 & 1 \end{bmatrix}$; $BA = \begin{bmatrix} 3 & 1 \\ 0 & 1 \end{bmatrix} = C$. **P2.14** **a)** $\begin{bmatrix} -5 & -5 \\ 4 & 2 \end{bmatrix}$; **b)** $\begin{bmatrix} -5 & 10 & -5 \\ 20 & 5 & 10 \end{bmatrix}$; **c)** $\begin{bmatrix} 17 & 28 \\ 41 & 64 \end{bmatrix}$; **d)** Doesn't exist; **e)** $\begin{bmatrix} 18 & 21 \\ 9 & 12 \\ 8 & 9 \end{bmatrix}$; **f)** $\begin{bmatrix} 9 & 12 \end{bmatrix}$; **g)** $\begin{bmatrix} -4 \\ -7 \end{bmatrix}$; **h)** 0; **i)** $\begin{bmatrix} 2 & 1 \\ -4 & -2 \end{bmatrix}$; **j)** 8; **k)** 11; **l)** $-3$; **m)** 20. **P2.15** **a)** $\det(B)$; **b)** $\det A$; **c)** 0. **P2.16** **a)** 6; **b)** $\frac{3}{2}$; **c)** 15; **d)** 30.

## Solutions to selected problems

**P2.1** A function is linear in $x$ if it contains $x$ raised only to the first power. Basically, $f(x) = mx$ (for some constant $m$) is the only possible linear function of one variable.

**P2.8** See `bit.ly/1cOa8yo` for calculations.

**P2.9** Any multiple of the vector $\vec{u}_1 \times \vec{u}_2 = (-3,1,3)$ is perpendicular to both $\vec{u}_1$ and $\vec{u}_2$. We must find a multiplier $t \in \mathbb{R}$ such that $t(-3,1,3)\cdot(1,1,0) = 8$. Computing the dot product we find $-3t + t = 8$, so $t = -4$. The vector we're looking for is $(12,-4,-12)$. See `bit.ly/1nmYH8T` for calculations.

**P2.10** The direction of the tractor changes constantly throughout the day, and the overall trajectory has the shape of a half-circle. The total distance travelled by the tractor is equal to half the circumference of a circle of radius $R$. Since it took the tractor six hours of movement at $v = 0.524$ km/h to travel half the circumference of the circle, we have $\frac{1}{2}C = \pi R = v(t_f - t_i) = 0.524(6)$, from which we find $R = 1$ km. The total distance travelled by the tractor is $\pi R = 3.14$ km.

**P2.11** Using algebra we find $\|\vec{u}-\vec{v}\|^2 = \|\vec{u}\|^2 + \|\vec{v}\|^2 - 2\vec{u}\cdot\vec{v}$. Using the cosine rule we find $\|\vec{u}-\vec{v}\|^2 = \|\vec{u}\|^2 + \|\vec{v}\|^2 - 2\|\vec{u}\|\|\vec{v}\|\cos(\varphi)$. Equating these two expressions, we can obtain the geometric formula for the cosine product.

**P2.13** Using the definition of the matrix vector product, we can imitate the action of each linear transformation $T$ by choosing appropriate entries in the matrix. The top row corresponds to the first component of the output; the bottom row corresponds to the second component of the output. Observe that $BA = C$. The composite transformation of applying $T_A$ followed by $T_B$ (denoted $T_B \circ T_A$), is equivalent to the transformation $T_C$. Note $AB \neq BA$: the matrix product $AB$ corresponds to compositing the composition of the linear transformations in the opposite order $T_A \circ T_B$.

# Chapter 3 solutions

## Answers to exercises

**E3.1** $x = 4$, $y = -2$. **E3.3** **a)** No solution; **b)** $(0,2)$; **c)** $\{(2,0) + s(-1,1), \forall s \in \mathbb{R}\}$. **E3.4** **a)** $X = BA^{-1}$; **b)** $X = C^{-1}B^{-1}A^{-1}ED^{-1}$; **c)** $X = AD^{-1}$. **E3.5** $P = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$;

# Appendix B

# Notation

This appendix contains a summary of the notation used in this book.

## Math notation

| Expression | Read as | Used to denote |
|---|---|---|
| $a, b, x, y$ | | variables |
| $=$ | is equal to | expressions that have the same value |
| $\stackrel{\text{def}}{=}$ | is defined as | new variable definitions |
| $a + b$ | $a$ plus $b$ | the combined lengths of $a$ and $b$ |
| $a - b$ | $a$ minus $b$ | the difference in lengths between $a$ and $b$ |
| $a \times b = ab$ | $a$ times $b$ | the area of a rectangle |
| $a^2 \stackrel{\text{def}}{=} aa$ | $a$ squared | the area of a square of side length $a$ |
| $a^3 \stackrel{\text{def}}{=} aaa$ | $a$ cubed | the volume of a cube of side length $a$ |
| $a^n$ | $a$ exponent $n$ | $a$ multiplied by itself $n$ times |
| $\sqrt{a} \stackrel{\text{def}}{=} a^{\frac{1}{2}}$ | square root of $a$ | the side length of a square of area $a$ |
| $\sqrt[3]{a} \stackrel{\text{def}}{=} a^{\frac{1}{3}}$ | cube root of $a$ | the side length of a cube with volume $a$ |
| $a/b = \frac{a}{b}$ | $a$ divided by $b$ | $a$ parts of a whole split into $b$ parts |
| $a^{-1} \stackrel{\text{def}}{=} \frac{1}{a}$ | one over $a$ | division by $a$ |
| $f(x)$ | $f$ of $x$ | the function $f$ applied to input $x$ |
| $f^{-1}$ | $f$ inverse | the inverse function of $f(x)$ |
| $f \circ g$ | $f$ compose $g$ | function composition; $f \circ g(x) \stackrel{\text{def}}{=} f(g(x))$ |
| $e^x$ | $e$ to the $x$ | the exponential function base $e$ |
| $\ln(x)$ | natural log of $x$ | the logarithm base $e$ |
| $a^x$ | $a$ to the $x$ | the exponential function base $a$ |
| $\log_a(x)$ | log base $a$ of $x$ | the logarithm base $a$ |
| $\theta, \phi$ | *theta*, *phi* | angles |
| sin, cos, tan | sin, cos, tan | trigonometric ratios |
| % | percent | proportions of a total; $a\% \stackrel{\text{def}}{=} \frac{a}{100}$ |

# Set notation

You don't need a lot of fancy notation to do math, but it really helps if you know a little bit of set notation.

| Symbol | Read as | Denotes |
|---|---|---|
| $\{\ldots\}$ | the set ... | definition of a set |
| $\mid$ | such that | describe or restrict the elements of a set |
| $\mathbb{N}$ | the naturals | the set $\mathbb{N} \stackrel{\text{def}}{=} \{0, 1, 2, \ldots\}$. Also $\mathbb{N}_+ \stackrel{\text{def}}{=} \mathbb{N} \backslash \{0\}$. |
| $\mathbb{Z}$ | the integers | the set $\mathbb{Z} \stackrel{\text{def}}{=} \{\ldots, -2, -1, 0, 1, 2, 3, \ldots\}$ |
| $\mathbb{Q}$ | the rationals | the set of fractions of integers |
| $\mathbb{R}$ | the reals | the set of real numbers |
| $\mathbb{C}$ | | the set of complex numbers |
| $\mathbb{F}_q$ | finite field | the set $\{0, 1, 2, 3, \ldots, q-1\}$ |
| $\subset$ | subset | one set strictly contained in another |
| $\subseteq$ | subset or equal | containment or equality |
| $\cup$ | union | the combined elements from two sets |
| $\cap$ | intersection | the elements two sets have in common |
| $S \backslash T$ | $S$ set minus $T$ | the elements of $S$ that are not in $T$ |
| $a \in S$ | $a$ in $S$ | $a$ is an element of set $S$ |
| $a \notin S$ | $a$ not in $S$ | $a$ is not an element of set $S$ |
| $\forall x$ | for all $x$ | a statement that holds for all $x$ |
| $\exists x$ | there exists $x$ | an existence statement |
| $\nexists x$ | there doesn't exist $x$ | a non-existence statement |
| $S \times T$ | Cartesian product | all pairs $(s, t)$ where $s \in S$ and $t \in T$ |

An example of a condensed math statement that uses set notation is "$\nexists m, n \in \mathbb{Z}$ such that $\frac{m}{n} = \sqrt{2}$," which reads "there don't exist integers $m$ and $n$ whose fraction equals $\sqrt{2}$." Since we identify the set of fractions of integers with the rationals, this statement is equivalent to the shorter "$\sqrt{2} \notin \mathbb{Q}$," which reads "$\sqrt{2}$ is irrational."

# Vectors notation

| Expression | Denotes |
|---|---|
| $\mathbb{R}^n$ | the set of $n$-dimensional real vectors |
| $\vec{v}$ | a vector |
| $(v_x, v_y)$ | vector in component notation |
| $v_x\hat{\imath} + v_y\hat{\jmath}$ | vector in unit vector notation |
| $\\|\vec{v}\\|\angle\theta$ | vector in length-and-direction notation |
| $\\|\vec{v}\\|$ | length of the vector $\vec{v}$ |
| $\theta$ | angle the vector $\vec{v}$ makes with the $x$-axis |
| $\hat{v} \stackrel{\text{def}}{=} \frac{\vec{v}}{\\|\vec{v}\\|}$ | unit vector in the same direction as $\vec{v}$ |
| $\vec{u} \cdot \vec{v}$ | dot product of the vectors $\vec{u}$ and $\vec{v}$ |
| $\vec{u} \times \vec{v}$ | cross product of the vectors $\vec{u}$ and $\vec{v}$ |

# Complex numbers notation

| Expression | Denotes |
|---:|---|
| $\mathbb{C}$ | the set of complex numbers $\mathbb{C} \stackrel{\text{def}}{=} \{a + bi \mid a, b \in \mathbb{R}\}$ |
| $i$ | the unit imaginary number $i \stackrel{\text{def}}{=} \sqrt{-1}$ and $i^2 = -1$ |
| $\text{Re}\{z\} = a$ | real part of $z = a + bi$ |
| $\text{Im}\{z\} = b$ | imaginary part of $z = a + bi$ |
| $\lvert z\rvert\angle\varphi_z$ | polar representation of $z = \lvert z\rvert\cos\varphi_z + i\lvert z\rvert\sin\varphi_z$ |
| $\lvert z\rvert = \sqrt{a^2 + b^2}$ | magnitude of $z = a + bi$ |
| $\varphi_z = \tan^{-1}(b/a)$ | phase or argument of $z = a + bi$ |
| $\bar{z} = a - bi$ | complex conjugate of $z = a + bi$ |
| $\mathbb{C}^n$ | the set of $n$-dimensional complex vectors |

# Vector space notation

| Expression | Denotes |
|---:|---|
| $U, V, W$ | vector spaces |
| $W \subseteq V$ | vector space $W$ subspace of vector space $V$ |
| $\{\vec{v} \in V \mid \langle\texttt{cond}\rangle\}$ | subspace of vectors in $V$ satisfying condition $\langle\texttt{cond}\rangle$ |
| $\text{span}(\vec{v}_1, \ldots, \vec{v}_n)$ | span of vectors $\vec{v}_1, \ldots, \vec{v}_n$ |
| $\dim(U)$ | dimension of vector space $U$ |
| | |
| $\mathcal{R}(M)$ | row space of $M$ |
| $\mathcal{N}(M)$ | null space of $M$ |
| $\mathcal{C}(M)$ | column space of $M$ |
| $\mathcal{N}(M^{\mathsf{T}})$ | left null space of $M$ |
| $\text{rank}(M)$ | rank of $M$; $\text{rank}(M) \stackrel{\text{def}}{=} \dim(\mathcal{R}(M)) = \dim(\mathcal{C}(M))$ |
| $\text{nullity}(M)$ | nullity of $M$; $\text{nullity}(M) \stackrel{\text{def}}{=} \dim(\mathcal{N}(M))$ |
| | |
| $B_s$ | the standard basis |
| $\{\vec{e}_1, \ldots, \vec{e}_n\}$ | an orthogonal basis |
| $\{\hat{e}_1, \ldots, \hat{e}_n\}$ | an orthonormal basis |
| ${}_{B'}[\mathbb{1}]_B$ | the change-of-basis matrix from basis $B$ to basis $B'$ |
| | |
| $\Pi_S$ | projection onto subspace $S$ |
| $\Pi_{S^\perp}$ | projection onto the orthogonal complement of $S$ |

# Abstract vector spaces notation

| Expression | Denotes |
|---|---|
| $(V, F, +, \cdot)$ | abstract vector space of vectors from the set $V$, whose coefficients are from the field $F$, addition operation "+" and scalar-multiplication operation "$\cdot$ " |
| $\mathbf{u}, \mathbf{v}, \mathbf{w}$ | abstract vectors |
| $\langle \mathbf{u}, \mathbf{v} \rangle$ | inner product of vectors $\mathbf{u}$ and $\mathbf{v}$ |
| $\\|\mathbf{u}\\|$ | norm of $\mathbf{u}$ |
| $d(\mathbf{u}, \mathbf{v})$ | distance between $\mathbf{u}$ and $\mathbf{v}$ |

# Notation for matrices and matrix operations

| Expression | Denotes |
|---|---|
| $\mathbb{R}^{m \times n}$ | the set of $m \times n$ matrices with real entries |
| $A$ | a matrix |
| $a_{ij}$ | entry in the $i^{\text{th}}$ row and $j^{\text{th}}$ column of $A$ |
| $\|A\|$ | determinant of $A$, also denoted $\det(A)$ |
| $A^{-1}$ | matrix inverse |
| $A^{\mathsf{T}}$ | matrix transpose |
| $\mathbb{1}$ | identity matrix; $\mathbb{1}A = A\mathbb{1} = A$ and $\mathbb{1}\vec{v} = \vec{v}$ |
| $AB$ | matrix-matrix product |
| $A\vec{v}$ | matrix-vector product |
| $\vec{w}^{\mathsf{T}}A$ | vector-matrix product |
| $\vec{u}^{\mathsf{T}}\vec{v}$ | vector-vector inner product; $\vec{u}^{\mathsf{T}}\vec{v} \stackrel{\text{def}}{=} \vec{u} \cdot \vec{v}$ |
| $\vec{u}\vec{v}^{\mathsf{T}}$ | vector-vector outer product |
| | |
| $\text{ref}(A)$ | row echelon form of $A$ |
| $\text{rref}(A)$ | reduced row echelon form of $A$ |
| $\text{rank}(A)$ | rank of $A \stackrel{\text{def}}{=}$ number of pivots in $\text{rref}(A)$ |
| $A \sim A'$ | matrix $A'$ obtained from matrix $A$ by row operations |
| $\mathcal{R}_1, \mathcal{R}_2, \ldots$ | row operations, of which there are three types:<br>$\rightarrow$ $R_i \leftarrow R_i + kR_j$: add $k$-times row $j$ to row $i$<br>$\rightarrow$ $R_i \leftrightarrow R_j$: swap rows $i$ and $j$<br>$\rightarrow$ $R_i \leftarrow mR_i$: multiply row $i$ by constant $m$ |
| $E_{\mathcal{R}}$ | elementary matrix for row operation $\mathcal{R}$; $\mathcal{R}(M) \stackrel{\text{def}}{=} E_{\mathcal{R}}M$ |
| $[\, A \mid \vec{b} \,]$ | augmented matrix containing matrix $A$ and vector $\vec{b}$ |
| $[\, A \mid B \,]$ | augmented matrix array containing matrices $A$ and $B$ |
| $M_{ij}$ | minor associated with entry $a_{ij}$. See page 196. |
| $\text{adj}(A)$ | adjugate matrix of $A$. See page 198. |
| $(A^{\mathsf{T}}A)^{-1}A^{\mathsf{T}}$ | Moore–Penrose inverse of $A$. See page 379. |
| $\mathbb{C}^{m \times n}$ | the set of $m \times n$ matrices with complex entries |
| $A^{\dagger}$ | Hermitian transpose; $A^{\dagger} \stackrel{\text{def}}{=} (\overline{A})^{\mathsf{T}}$ |

# Notation for linear transformations

| Expression | Denotes |
|---:|---|
| $T : \mathbb{R}^n \to \mathbb{R}^m$ | linear transformation $T$ from the input space $\mathbb{R}^n$ to the output space $\mathbb{R}^m$ |
| $M_T \in \mathbb{R}^{m \times n}$ | matrix representation of $T$ |
| $\text{Im}(T) = \mathcal{C}(M_T)$ | the image space of $T$ = column space of $M_T$ |
| $\text{Ker}(T) = \mathcal{N}(M_T)$ | the kernel of $T$ = null space of $M_T$ |
| $S \circ T(\vec{x})$ | composition of linear transformations; $S \circ T(\vec{x}) \stackrel{\text{def}}{=} S(T(\vec{x})) = M_S M_T \vec{x}$ |
| $M \in \mathbb{R}^{m \times n}$ | an $m \times n$ matrix |
| $T_M : \mathbb{R}^n \to \mathbb{R}^m$ | the linear transformation defined as $T_M(\vec{v}) \stackrel{\text{def}}{=} M\vec{v}$ |
| $T_{M^{\mathsf{T}}} : \mathbb{R}^m \to \mathbb{R}^n$ | the adjoint linear transformation $T_{M^{\mathsf{T}}}(\vec{a}) \stackrel{\text{def}}{=} \vec{a}^{\mathsf{T}} M$ |

# Matrix decompositions

| Expression | Denotes |
|---:|---|
| $A \in \mathbb{R}^{n \times n}$ | a matrix (assume diagonalizable) |
| $p_A(\lambda) \stackrel{\text{def}}{=} \lvert A - \lambda \mathbb{1} \rvert$ | characteristic polynomial of $A$ |
| $\lambda_1, \ldots, \lambda_n$ | eigenvalues of $A$ = roots of $p_A(\lambda)$ |
| $\Lambda \in \mathbb{R}^{n \times n}$ | diagonal matrix of eigenvalues of $A$ |
| $\vec{e}_{\lambda_1}, \ldots, \vec{e}_{\lambda_n}$ | eigenvectors of $A$ |
| $Q \in \mathbb{R}^{n \times n}$ | matrix whose columns are eigenvectors of $A$ |
| $A = Q \Lambda Q^{-1}$ | eigendecomposition of $A$ |
| $A = O \Lambda O^{\mathsf{T}}$ | eigendecomposition of a normal matrix |
| $B \in \mathbb{R}^{m \times n}$ | a generic matrix |
| $\sigma_1, \sigma_2, \ldots$ | singular values of $B$ |
| $\Sigma \in \mathbb{R}^{m \times n}$ | matrix of singular values of $B$ |
| $\vec{u}_1, \ldots, \vec{u}_m$ | left singular vectors of $B$ |
| $U \in \mathbb{R}^{m \times m}$ | matrix of left singular vectors of $B$ |
| $\vec{v}_1, \ldots, \vec{v}_n$ | right singular vectors of $B$ |
| $V \in \mathbb{R}^{n \times n}$ | matrix of right singular vectors of $B$ |
| $B = U \Sigma V^{\mathsf{T}}$ | singular value decomposition of $B$ |

# Index

The page numbers shown in **bold** point to the concept definitions.

Linear algebra is the foundation of science and engineering. Knowledge of linear algebra is a prerequisite for studying statistics, machine learning, computer graphics, signal processing, chemistry, economics, quantum mechanics, and countless other applications. Indeed, linear algebra offers a powerful toolbox for modelling the real world.

The NO BULLSHIT GUIDE TO LINEAR ALGEBRA shows the connections between the computational techniques of linear algebra, their geometric interpretations, and the theoretical foundations. This university-level textbook contains lessons on linear algebra written in a style that is precise and concise. Each concept is illustrated through definitions, formulas, diagrams, explanations, and examples of real-world applications. Readers build their math superpowers by solving practice problems and learning to use the computer algebra system SymPy to speed up tedious matrix arithmetic tasks.

> *"The book explains the concepts in a way that gives a strong intuitive understanding."* — Joe Nestor, student
>
> *"It's very well written and a fun read!"* — Felix Kwok, professor
>
> *"I used this book in multiple big data courses when I needed a deeper understanding of the material."* — Zane Zakraisek, student

The author, Ivan Savov, combines 15 years of tutoring experience with a B.Eng. in electrical engineering, an M.Sc. in physics, and a Ph.D. in computer science from McGill University.

**Contents:**



The **No Bullshit Guide to Linear Algebra** by Ivan Savov (Minireference Publishing, v2.2 2020, ISBN 0992001021) is available in both print and digital formats.